



Informatica®
10.4

웹 서비스 가이드

Informatica 웹 서비스 가이드

10.4

2019년12월

© 저작권 Informatica LLC 2009, 2020

이 소프트웨어와 설명서는 사용 및 공개에 대한 제한 사항이 포함되어 있는 별도의 사용권 계약에 따라서만 제공됩니다. 본 문서의 어떤 부분도 Informatica LLC의 사전 통지 없이 어떠한 형태나 수단(전자적, 사진 복사, 녹음 등)으로 복제되거나 전송될 수 없습니다.

미국 정부 권한. 미국 정부 고객에게 제공되는 프로그램, 소프트웨어, 데이터베이스, 관련 문서 및 기술 데이터는 해당하는 연방 입수 규정 및 기관별 보안 규정에 따라 "상용 컴퓨터 소프트웨어" 또는 "상용 기술 데이터"입니다. 따라서 사용, 복제, 공개, 수정 및 조정은 해당하는 정부 계약에 규정된 제한 사항 및 라이선스 조건을 따르며, 정부 계약 조건에 의해 적용 가능한 한도 내에서, FAR 52.227-19, 상용 소프트웨어 라이선스에 규정된 추가 권한이 적용됩니다.

Informatica and the Informatica logo are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>. Other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties. Required third party notices are included with the product.

이 설명서의 정보는 예고 없이 변경될 수 있습니다. 이 문서에서 문제가 발견되는 경우 infa_documentation@informatica.com으로 보고해 주십시오.

Informatica 제품은 제품이 제공될 당시의 계약 조건에 따라 보증됩니다. Informatica는 상품성과 특정 목적에의 적합성에 대한 보증 그리고 비침해에 대한 보증 또는 조건을 포함하여 어떠한 종류의 명시적이거나 묵시적인 보증 없이 이 문서의 정보를 "있는 그대로" 제공합니다.

발행 날짜: 2020-02-05

목차

서문	10
Informatica 리소스	10
Informatica 네트워크	10
Informatica 기술 자료	10
Informatica 설명서	10
Informatica Product Availability Matrix	11
Informatica Velocity	11
Informatica Marketplace	11
Informatica 글로벌 고객 지원 센터	11
장 1: 웹 서비스	12
웹 서비스 개요	12
REST 웹 서비스와 SOAP 웹 서비스의 차이점	13
웹 서비스 프로세스	13
웹 서비스 소비자 변환 프로세스	14
장 2: SOAP 웹 서비스	15
SOAP 웹 서비스 구성 요소	15
작업	15
WSDL	16
SOAP	16
SOAP 웹 서비스 개발	16
SOAP 웹 서비스 예제	17
장 3: WSDL 데이터 개체	19
WSDL 데이터 개체 개요	19
WSDL 데이터 개체 개요 보기	20
WSDL 데이터 개체 고급 보기	21
WSDL 데이터 개체 가져오기	21
WSDL 동기화	22
WSDL 데이터 개체 동기화	22
인증서 관리	23
Informatica Developer 인증서 속성	23
Informatica Developer에 인증서 추가	23
장 4: 스키마 개체	24
스키마 개체 개요	24
스키마 개체 개요 보기	24
스키마 파일	25
스키마 개체 스키마 보기	25

네임스페이스 속성.....	26
요소 속성.....	27
단순 유형 속성.....	28
복합 유형 속성.....	29
특성 속성.....	30
스키마 개체 고급 보기.....	31
스키마 개체 작성.....	31
스키마 업데이트.....	32
스키마 동기화.....	32
스키마 파일 편집.....	33
인증서 관리.....	35
Informatica Developer 인증서 속성.....	36
Informatica Developer에 인증서 추가.....	36

장 5: SOAP 웹 서비스를 생성하는 방법..... 37

SOAP 웹 서비스 생성 개요.....	37
유형 및 요소.....	38
웹 서비스 개요 보기.....	38
웹 서비스 WSDL 보기.....	40
WSDL 데이터 개체에서 웹 서비스 작성.....	40
1단계. WSDL 데이터 개체에서 웹 서비스 작성.....	41
2단계. 웹 서비스에 작업 추가.....	41
WSDL 데이터 개체를 웹 서비스와 연결.....	42
SOAP 웹 서비스를 수동으로 생성.....	42
1단계. 수동 웹 서비스 작성.....	42
2단계. 작업 작성.....	43
3단계. 요소 작성.....	46
4단계. 사전 정의된 결합 작성.....	47
5단계. 헤더 작성.....	47

장 6: 작업 매핑..... 48

작업 매핑 개요.....	48
작업 매핑의 일반 탭.....	49
작업 매핑 작업 탭.....	49
작업 매핑 고급 탭.....	49
입력 변환.....	49
입력 변환 포트 탭.....	50
작업 입력을 포트에 매핑하기 위한 규칙 및 지침.....	50
입력 변환 구성.....	50
출력 변환.....	52
출력 변환 포트 탭.....	52
출력 변환 고급 탭.....	52

포트를 작업 출력에 매핑하는 규칙 및 지침.....	52
출력 변환 구성.....	53
결합 변환.....	54
결합 변환 포트 탭.....	55
결합 변환 고급 탭.....	55
포트를 작업 결합에 매핑하는 규칙 및 지침.....	55
결합 변환 작성.....	56
결합 변환 구성.....	56
결합 처리.....	57
시스템 정의된 결합.....	59
사전 정의된 결합.....	59
일반 결합.....	60
작업 매핑 테스트.....	61
작업 매핑 테스트.....	61
사용자 지정된 보기 옵션.....	61

장 7: 웹 서비스 SOAP 메시지 구문 분석..... 62

웹 서비스 SOAP 메시지 구문 분석 개요.....	62
변환 사용자 인터페이스.....	62
다중 발생 출력 구성.....	63
정규화된 관계형 출력.....	64
생성된 키.....	64
비정규화된 관계형 출력.....	65
피벗된 관계형 출력.....	65
anyType 요소 구문 분석.....	66
파생된 유형 구문 분석.....	67
QName 요소 구문 분석.....	68
대체 그룹 구문 분석.....	68
SOAP 메시지의 XML 구성 구문 분석.....	68
선택 요소.....	68
목록 요소.....	69
합집합 요소.....	69

장 8: 웹 서비스 SOAP 메시지 생성..... 70

웹 서비스 SOAP 메시지 생성 개요.....	70
변환 사용자 인터페이스.....	71
입력 포트 영역.....	71
작업 영역.....	72
포트 및 계층 수준 관계.....	72
키.....	73
포트 매핑.....	74
포트 매핑.....	75

그룹 매핑.....	75
여러 포트 매핑.....	75
여러 번 발생하는 포트 피벗.....	76
비정규화된 데이터 매핑.....	77
파생된 유형 및 요소 대체.....	78
파생된 유형 생성.....	78
anyType 요소 및 특성 생성.....	79
대체 그룹 생성.....	79
SOAP 메시지의 XML 구성 생성.....	80
선택 요소.....	80
목록 요소.....	80
합집합 요소.....	81
장 9: 웹 서비스 소비자 변환.....	82
웹 서비스 소비자 변환 개요.....	82
SOAP 메시지.....	82
WSDL 파일.....	83
작업.....	83
웹 서비스 보안.....	84
WSDL 선택.....	84
웹 서비스 소비자 변환 포트.....	85
HTTP 헤더 입력 포트.....	86
기타 입력 포트.....	86
웹 서비스 소비자 변환 입력 매핑.....	86
입력 포트를 노드에 매핑하기 위한 규칙 및 지침.....	87
보기 사용자 지정의 옵션.....	88
작업 입력에 입력 포트 매핑.....	88
웹 서비스 소비자 변환 출력 매핑.....	89
노드를 출력 포트에 매핑하기 위한 규칙 및 지침.....	90
SOAP 메시지를 XML로 매핑.....	90
보기 사용자 지정의 옵션.....	91
출력 포트로 작업 출력 매핑.....	91
웹 서비스 소비자 변환 고급 속성.....	92
웹 서비스 오류 처리.....	94
메시지 압축.....	95
동시 실행.....	95
필터 최적화.....	96
웹 서비스 소비자 변환의 초기 선택 최적화 활성화.....	96
웹 서비스 소비자 변환의 푸시인 최적화.....	96
웹 서비스 소비자 변환 작성.....	98
웹 서비스 소비자 변환 예제.....	100
입력 파일.....	100

논리적 데이터 개체 모델.	100
논리적 데이터 개체 매핑.	100
웹 서비스 소비자 변환.	101

장 10: REST 웹 서비스 103

REST 웹 서비스 개요.	103
REST 웹 서비스 프로세스.	104
웹 서비스 소비자 변환 프로세스.	104
REST 웹 서비스 리소스.	105
REST 웹 서비스 스키마 보기.	107
데이터 개체 동기화.	107
리소스 키.	107
리소스 매핑.	108
기본 리소스 매핑.	108
사용자 지정 리소스 매핑.	109
REST 웹 서비스 출력 변환.	109
REST 출력 변환에서 여러 번 발생하는 데이터.	110
요청 메시지.	111
리소스 매핑의 데이터 필터링.	112
키로 검색.	112
응답 메시지 형식.	113
응답 데이터 미리보기.	114

장 11: REST 웹 서비스를 생성하는 방법 116

REST 웹 서비스 생성.	116
REST 웹 서비스를 수동으로 생성하는 방법.	117
예제 REST 웹 서비스.	117
1단계. REST 웹 서비스 리소스 생성.	117
REST 웹 서비스 리소스 생성.	117
2단계. 리소스 매핑 정의.	120
리소스 매핑 정의.	121
3단계. 출력 매핑 구성.	123
출력 매핑 구성.	124
4단계. 데이터 뷰어 보기에서 매핑 테스트.	125
리소스 ID를 기준으로 출력 필터링.	125
필터 조건을 기준으로 출력 필터링.	126
5단계. 응용 프로그램 배포.	127
응용 프로그램 배포.	127
6단계. 브라우저에서 웹 서비스 쿼리.	128
웹 서비스 쿼리.	129
데이터 개체에서 REST 웹 서비스를 생성하는 방법.	130
데이터 개체를 REST 웹 서비스로 배포하는 방법.	133

장 12: REST 웹 서비스 소비자 변환.....	136
REST 웹 서비스 소비자 변환 개요.....	136
REST 웹 서비스 소비자 변환 프로세스.....	137
REST 웹 서비스 소비자 변환 구성.....	138
메시지 구성.....	138
리소스 식별.....	138
HTTP 메서드.....	139
HTTP Get 메서드.....	139
HTTP Post 메서드.....	140
HTTP Put 메서드.....	141
HTTP Delete 메서드.....	141
REST 웹 서비스 소비자 변환 포트.....	142
입력 포트.....	142
출력 포트.....	142
통과 포트.....	143
인수 포트.....	143
URL 포트.....	143
HTTP 헤더 포트.....	143
쿠키 포트.....	144
출력 XML 포트.....	144
응답 코드 포트.....	144
REST 웹 서비스 소비자 변환 입력 매핑.....	144
입력 포트를 요소에 매핑하기 위한 규칙 및 지침.....	145
메서드 입력에 입력 포트 매핑.....	145
REST 웹 서비스 소비자 변환 출력 매핑.....	146
요소를 출력 포트에 매핑하기 위한 규칙 및 지침.....	147
보기 사용자 지정의 옵션.....	147
출력 포트에 메서드 출력 매핑.....	148
REST 웹 서비스 소비자 변환의 고급 속성.....	148
REST 웹 서비스 소비자 변환 작성.....	149
REST 웹 서비스 소비자 변환 작성.....	149
배열이 포함된 JSON 응답 메시지 구문 분석.....	150
JSON 응답 메시지 예제.....	150
응답 메시지의 명명되지 않은 배열.....	151
 장 13: REST 웹 서비스 소비자 변환 사용 사례.....	 152
REST 웹 서비스 소비자 변환을 구성하여 데이터를 수신하는 방법.....	153
웹 서비스에서 구문 분석되지 않은 데이터를 수신하는 방법.....	153
웹 서비스에서 구문 분석된 데이터를 수신하는 방법.....	155
스키마를 사용하여 REST 웹 서비스 소비자 변환을 생성하는 방법.....	159
XML 형식의 요청 및 응답 메시지를 처리하는 방법.....	159

JSON 형식의 요청 및 응답 메시지를 구문 분석하는 방법.....	165
REST 웹 서비스 소비자 변환에서 사용자 지정 URL을 생성하는 방법.....	168
URL 포트를 사용하여 동적 URL을 작성하는 방법.....	169
인수 포트를 사용하여 URL 매개 변수를 생성하는 방법.....	169
장 14: REST 및 SOAP 웹 서비스 관리.....	172
웹 서비스 관리 개요.....	172
웹 서비스 속성 구성.....	172
웹 서비스 속성.....	173
웹 서비스 작업 및 리소스 속성.....	175
웹 서비스 결과 집합 캐싱.....	175
웹 서비스 보안 관리.....	176
웹 서비스 사용 권한.....	177
SOAP 요청의 UserName 토큰.....	178
그리드의 웹 서비스.....	180
서비스 프로세스에서 작업을 실행하는 그리드 구성.....	180
웹 서비스 로그.....	181
웹 서비스 추적 수준.....	181
웹 서비스 모니터링.....	181
웹 서비스의 속성 보기.....	182
웹 서비스의 보고서 보기.....	182
REST 또는 SOAP 웹 서비스의 작업 보기.....	182
웹 서비스의 요청 보기.....	183
부록 A: 데이터 유형 호환성.....	184
데이터 유형 참조 개요.....	184
XML 및 변환 데이터 유형.....	185
10진수.....	186
인덱스.....	187

서문

Informatica® 웹 서비스 가이드를 읽고 데이터 액세스, 변환 또는 전달을 위해 웹 서비스 클라이언트를 Informatica 웹 서비스에 연결하는 방법을 알아보십시오. SOAP(Simple Object Access Protocol) 또는 REST(Representational State Transfer) 메시징 프로토콜로 웹 서비스를 생성하여 데이터를 액세스 또는 업데이트하거나 다른 태스크를 수행하는 방법을 알아보십시오.

Informatica 리소스

Informatica는 Informatica Network 및 기타 온라인 포털을 통해 다양한 범위의 제품 리소스를 제공합니다. 리소스를 통해 Informatica 제품 및 솔루션을 최대한 활용하고 다른 Informatica 사용자 및 주제별 전문가로부터 배울 수 있습니다.

Informatica 네트워크

Informatica Network는 Informatica 기술 자료, Informatica 글로벌 고객 지원 센터 등 여러 리소스로 연결되는 관문입니다. Informatica Network를 시작하려면 <https://network.informatica.com>을 방문하십시오.

Informatica Network 멤버인 경우 다음 옵션이 가능합니다.

- 기술 자료에서 제품 리소스를 검색할 수 있습니다.
- 제품 사용 가능 여부에 대한 정보를 봅니다.
- 지원 사례를 생성하고 검토할 수 있습니다.
- 거주 지역의 Informatica 사용자 그룹 네트워크를 검색하고 동료와 협업 관계 유지

Informatica 기술 자료

Informatica 기술 자료를 사용하여 사용 방법 문서, 모범 사례, 비디오 자습서, 자주 묻는 질문에 대한 답변 등 제품 리소스를 확인할 수 있습니다.

기술 자료를 검색하려면 <https://search.informatica.com>을 방문하십시오. 기술 자료에 대한 질문, 의견 또는 아이디어가 있는 경우 KB_Feedback@informatica.com을 통해 Informatica 기술 자료 팀에 문의해 주시기 바랍니다.

Informatica 설명서

Informatica 설명서 포털에서 확장된 설명서 라이브러리를 탐색하여 현재 및 최근 제품 릴리스를 확인할 수 있습니다. 설명서 포털을 탐색하려면 <https://docs.informatica.com>을 방문하십시오.

제품 설명서에 대한 질문, 의견 또는 아이디어가 있는 경우 infa_documentation@informatica.com에서 Informatica 설명서 팀에 문의해 주시기 바랍니다.

Informatica Product Availability Matrix

PAM(Product Availability Matrix)은 제품 릴리스에서 지원하는 운영 체제 버전, 데이터베이스 및 데이터 소스 유형과 대상을 나타냅니다.

<https://network.informatica.com/community/informatica-network/product-availability-matrices>에서 Informatica PAM을 찾을 수 있습니다.

Informatica Velocity

Informatica Velocity는 수백 가지 데이터 관리 프로젝트의 실제 경험을 토대로 Informatica 전문 서비스업에서 개발한 팁과 모범 사례 모음입니다. Informatica Velocity는 전 세계의 조직과 협력하여 성공적인 데이터 관리 솔루션을 계획, 개발, 배포 및 유지 관리하는 Informatica 컨설턴트의 포괄적인 지식을 보여줍니다.

Informatica Velocity 리소스는 <http://velocity.informatica.com>에서 확인할 수 있습니다. Informatica Velocity에 대한 질문, 주석 또는 아이디어가 있으시면 Informatica 전문 서비스업(ips@informatica.com)에 문의하십시오.

Informatica Marketplace

Informatica Marketplace는 Informatica 구현을 확대 및 개선하기 위한 솔루션을 찾을 수 있는 포럼입니다. Marketplace에서 Informatica 개발자와 파트너가 제공하는 수백 개의 솔루션을 활용하여 생산성을 향상시키고 프로젝트의 구현에 걸리는 시간을 줄일 수 있습니다. <https://marketplace.informatica.com>에서 Informatica Marketplace를 찾을 수 있습니다.

Informatica 글로벌 고객 지원 센터

전화 또는 Informatica Network를 통해 글로벌 지원 센터에 문의할 수 있습니다.

해당 지역의 Informatica 글로벌 고객 지원 전화 번호는 Informatica 웹 사이트 (<https://www.informatica.com/services-and-training/customer-success-services/contact-us.html>)를 방문하여 찾을 수 있습니다.

Informatica Network에서 온라인 지원 리소스를 찾으려면 <https://network.informatica.com>을 방문하고 eSupport 옵션을 선택하십시오.

제 1 장

웹 서비스

이 장에 포함된 항목:

- [웹 서비스 개요, 12](#)
- [REST 웹 서비스와 SOAP 웹 서비스의 차이점, 13](#)
- [웹 서비스 프로세스, 13](#)
- [웹 서비스 소비자 변환 프로세스, 14](#)

웹 서비스 개요

웹 서비스 클라이언트는 **Informatica** 웹 서비스에 연결하여 데이터를 액세스, 변환 또는 제공할 수 있습니다. 외부 응용 프로그램 또는 웹 서비스 소비자 변환이 웹 서비스 클라이언트로 웹 서비스에 연결할 수 있습니다.

Developer tool에서 **Informatica** 웹 서비스를 생성할 수 있습니다.

웹 서비스는 정보 요청, 데이터 업데이트 요청 또는 태스크 수행 요청을 처리할 수 있습니다. 예를 들어 웹 서비스 작업을 실행하는 요청을 웹 서비스 클라이언트가 보냅니다. 웹 서비스 클라이언트가 요청에서 고객 ID를 전달합니다. 웹 서비스는 고객 및 주문 정보를 검색하여 해당 정보를 클라이언트에 응답으로 반환합니다.

Informatica 웹 서비스는 **SOAP(Simple Object Access Protocol)** 또는 **REST(Representational State Transfer)** 메시징 프로토콜을 사용하여 웹 서비스 클라이언트와 통신합니다.

Developer tool에서는 다음과 같은 유형의 웹 서비스 또는 웹 서비스 클라이언트를 생성할 수 있습니다.

SOAP 웹 서비스

SOAP 프로토콜을 사용하는 웹 서비스입니다. 웹 서비스 클라이언트 요청 및 웹 서비스 응답은 **SOAP** 메시지는입니다. **WSDL(Web Service Description Language)**은 웹 서비스의 기능을 설명하는 **XML** 기반 인터페이스 정의 언어입니다. **WSDL** 파일에는 웹 서비스를 호출하는 방법, 웹 서비스에 필요한 매개 변수 및 웹 서비스가 반환하는 데이터 구조에 대한 설명이 포함됩니다. **WSDL** 파일에서 **Informatica SOAP** 웹 서비스를 생성할 수 있습니다.

SOAP 웹 서비스 소비자 변환

매핑 중에 데이터를 액세스하거나 변환하기 위해 웹 서비스 클라이언트로 웹 서비스에 연결합니다. **WSDL**에서 **SOAP** 웹 서비스 소비자 변환을 생성할 수 있습니다.

REST 웹 서비스

웹 서비스 작업을 수행하는 **HTTP** 요청을 수신하는 웹 서비스입니다. **Informatica REST** 웹 서비스는 **GET** 작업을 수행하는 **HTTP** 요청을 수신할 수 있습니다. **Informatica REST** 웹 서비스는 **JSON** 파일 또는 **XML** 파일로 응답을 반환할 수 있습니다.

REST 소비자 변환

매핑 중에 데이터를 액세스하거나 변환하기 위해 웹 서비스 클라이언트로 REST 웹 서비스에 연결합니다. REST 웹 서비스 소비자 변환은 사용자가 변환, HTTP 연결 또는 HTTPS 연결에 정의하는 URL을 통해 웹 서비스에 연결합니다. 요청 및 응답 메시지는 XML 또는 JSON 데이터가 포함됩니다.

REST 웹 서비스와 SOAP 웹 서비스의 차이점

Informatica Developer tool에서는 REST 웹 서비스 또는 SOAP 웹 서비스를 생성할 수 있습니다.

REST 웹 서비스와 SOAP 웹 서비스의 차이점은 다음과 같습니다.

요청 메시지 형식

SOAP 메시지는 구조화된 XML입니다. SOAP 웹 서비스는 XML을 구문 분석하여 웹 서비스가 수행해야 하는 작업을 확인합니다. REST 요청은 쿼리가 포함된 단순 URI 문자열입니다.

응답 메시지 형식

SOAP 웹 서비스는 WSDL에 의해 정의된 대로 XML 형식으로 응답을 반환합니다.

Informatica REST 웹 서비스는 JSON(JavaScript Object Notation) 또는 XML 응답 메시지를 반환합니다. 응답 메시지 형식은 WSDL 또는 스키마에 의해 정의되지 않습니다. 출력 형식은 Informatica REST 웹 서비스를 정의할 때 정의합니다.

웹 서비스 매핑 형식

Informatica SOAP 웹 서비스에는 작업 매핑이 포함됩니다. SOAP 작업 매핑에는 요청 메시지에서 XML을 구문 분석하는 입력 변환이 포함됩니다. 클라이언트 요청에서 요구하는 대로 데이터를 처리할 수 있도록 웹 서비스 매핑에 변환을 추가해야 합니다.

Informatica REST 웹 서비스에는 리소스 매핑이 포함됩니다. 리소스 매핑은 요청 쿼리를 읽지 않습니다. REST 리소스 매핑에는 입력 변환 대신 읽기 변환이 포함됩니다. 읽기 변환은 모델 리포지토리에 있는 데이터 개체를 읽어서 클라이언트에 반환할 데이터를 검색합니다. 기본적으로 클라이언트 쿼리를 기반으로 데이터를 검색하기 위해 필터 변환이나 조희 변환을 추가하지 않아도 됩니다. REST 웹 서비스는 매핑이 데이터를 반환한 이후에 출력 데이터를 필터링합니다.

웹 서비스 프로세스

웹 서비스는 웹 서비스 클라이언트로부터 요청을 수신합니다.

다음 프로세스에서는 데이터 통합 서비스가 웹 서비스 클라이언트의 웹 서비스 요청을 처리하는 방법을 설명합니다.

1. 데이터 통합 서비스가 웹 서비스 클라이언트로부터 요청을 수신합니다.
2. 데이터 통합 서비스의 웹 서비스 모듈 또는 데이터 통합 서비스의 REST 웹 서비스 모듈이 매핑을 실행하여 요청을 처리합니다.
3. 웹 서비스 모듈 또는 REST 웹 서비스 모듈이 웹 서비스 클라이언트에 응답을 보냅니다.

웹 서비스 소비자 변환 프로세스

외부 응용 프로그램 또는 웹 서비스 소비자 변환이 웹 서비스 클라이언트로 웹 서비스에 연결할 수 있습니다.

다음 프로세스에서는 웹 서비스 소비자 변환이 요청을 전송하고 웹 서비스에서 응답을 수신하는 방법을 설명합니다.

1. 웹 서비스 소비자 변환이 요청을 생성하고, 연결 개체를 사용하여 웹 서비스에 연결합니다.
2. 웹 서비스 소비자 변환이 웹 서비스로부터 응답을 수신합니다.
3. 웹 서비스 소비자 변환이 응답에서 데이터를 추출하고, 변환 출력 포트에서 데이터를 반환합니다.

제 2 장

SOAP 웹 서비스

이 장에 포함된 항목:

- [SOAP 웹 서비스 구성 요소, 15](#)
- [SOAP 웹 서비스 개발, 16](#)
- [SOAP 웹 서비스 예제, 17](#)

SOAP 웹 서비스 구성 요소

SOAP 웹 서비스 구성 요소는 웹 서비스의 용도 및 웹 서비스 클라이언트가 웹 서비스와 통신하는 방법을 정의합니다.

웹 서비스에는 다음과 같은 구성 요소가 있습니다.

작업

웹 서비스에 하나 이상의 작업을 포함할 수 있습니다. 각 작업은 웹 서비스의 작업에 해당합니다.

WSDL(Web Services Description Language)

WSDL은 웹 서비스 작업의 프로토콜, 형식 및 서명을 설명하는 XML 문서입니다.

SOAP(Simple Object Access Protocol)

SOAP는 웹 서비스의 통신 프로토콜입니다.

작업

웹 서비스에는 웹 서비스가 지원하는 각 작업에 대한 작업이 포함됩니다.

예를 들어 웹 서비스에 이름이 `getcustomerid`인 작업이 포함될 수 있습니다. 이 작업은 고객 세부 정보를 사용하여 고객 이름과 응답을 검색합니다. 작업 입력에는 고객 이름에 대한 요소가 포함됩니다. 작업 출력에는 고객 이름에 따라 고객 세부 정보에 대한 요소가 포함됩니다.

Developer tool에서 작업을 정의할 때는 작업 구성 요소를 정의합니다. 작업에는 다음 구성 요소가 포함됩니다.

작업 입력 및 출력

작업 입력은 작업에 대한 SOAP 요청의 요소를 정의합니다. 작업 출력은 작업에 대한 SOAP 응답의 요소를 정의합니다.

작업 입력 및 작업 출력에는 헤더가 포함될 수 있습니다. 헤더는 SOAP 메시지 안에서 데이터를 수신하거나 전송합니다. 헤더는 SOAP 요청 또는 SOAP 응답의 헤더 요소를 정의합니다.

작업 결합

작업 결합은 작업의 결과로 출력될 수 있는 오류 메시지에 대한 메시지 형식을 정의합니다. 단일 작업에 대해 여러 작업 결합을 정의할 수 있습니다.

각 작업에 대해 하나의 작업 매핑을 구성해야 합니다. 작업 입력, 작업 출력 및 각 작업 결합은 작업 매핑에서 하나의 변환에 해당합니다.

WSDL

WSDL은 웹 서비스 작업의 프로토콜, 형식 및 서명을 설명하는 XML 스키마입니다.

서비스 요청을 보내는 사람과 받는 사람이 모두 교환되는 데이터를 이해할 수 있도록 웹 서비스에 전달되는 데이터에 대한 설명이 WSDL에 포함되어 있습니다. WSDL 요소에는 해당 데이터에서 수행되는 작업에 대한 설명이 포함되어 있으므로 메시지를 받는 사람이 처리 방법을 알 수 있습니다. WSDL 요소에는 프로토콜 또는 전송에 대한 바인딩이 포함되어 있으므로 메시지를 보내는 사람이 전송 방법을 알 수 있습니다.

Developer 도구 또는 Administrator 도구에서 웹 서비스의 WSDL을 볼 수 있습니다. 웹 서비스를 데이터 통합 서비스에 배포한 후 WSDL URL을 보거나 WSDL을 파일에 다운로드할 수 있습니다. Administrator 도구에서 표시하는 WSDL URL에 액세스할 경우 WSDL의 콘텐츠를 볼 수 있습니다.

SOAP

SOAP는 웹 서비스의 통신 프로토콜입니다. 웹 서비스, 요청, 응답 및 결합 메시지에 대한 형식을 정의합니다. 데이터 통합 서비스는 문서/리터럴 인코딩이 포함된 SOAP 1.1 및 SOAP 1.2 메시지를 처리할 수 있습니다.

SOAP 메시지에는 다음 섹션이 포함되어 있습니다.

SOAP 엔벨로프

엔벨로프는 메시지의 프레임워크, 메시지의 콘텐츠 및 메시지를 처리하는 대상을 정의합니다.

SOAP 헤더

헤더는 SOAP 메시지를 전송한 항목을 식별합니다. 헤더에는 인증 정보가 포함됩니다. 또한 SOAP 메시지를 처리하는 방법에 대한 정보도 포함됩니다.

SOAP 본문

본문은 클라이언트 및 웹 서비스 공급자가 서로 간에 전달하는 데이터의 컨테이너입니다.

SOAP 메시지는 XML입니다. SOAP 메시지에 다중 발생 요소가 포함된 경우 요소 그룹이 XML 계층의 수준을 구성합니다. 한 수준이 다른 수준 안에 중첩될 경우 그룹이 연결됩니다.

SOAP 요청 메시지에 계층 데이터가 포함될 수 있습니다. 예를 들어, 클라이언트에서 고객 주문을 영업 데이터베이스에 추가하는 요청을 전송합니다. 클라이언트에서 2개의 데이터 그룹을 SOAP 요청 메시지를 통해 전달합니다. 한 그룹에는 고객 ID와 이름이 포함되어 있고, 다른 그룹에는 주문 정보가 포함되어 있습니다. 주문 정보가 여러 번 발생합니다.

SOAP 응답 메시지에 계층 데이터가 포함될 수 있습니다. 예를 들어, 웹 서비스 클라이언트에서 고객 주문에 대한 SOAP 요청을 생성합니다. 웹 서비스에서 주문 헤더 및 다중 발생 주문 세부 정보 요소를 SOAP 응답을 통해 반환합니다.

SOAP 웹 서비스 개발

웹 서비스 클라이언트가 작업을 수행하기 위해 사용할 수 있는 인터페이스를 제공하려면 SOAP 웹 서비스를 개발합니다. 웹 서비스 클라이언트는 외부 웹 서비스 클라이언트 또는 웹 서비스 소비자 변환일 수 있습니다. 예를

들어, 웹 서비스 클라이언트가 웹 서비스에 연결하여 고객 이름 또는 고객 ID를 기반으로 고객 세부 정보를 볼 수 있습니다.

웹 서비스를 개발하려면 다음 단계를 완료하십시오.

1. 웹 서비스를 작성합니다.
 - WSDL 데이터 개체에서 웹 서비스를 작성합니다. WSDL 파일을 가져와서 WSDL 데이터 개체를 작성합니다. WSDL 파일에서 웹 서비스의 작업 입력, 작업 출력 및 작업 결함을 정의합니다.
 - 웹 서비스를 수동으로 작성합니다. 작업 입력, 작업 출력 및 작업 결함을 구성합니다. 스키마 개체의 요소 및 유형을 사용하여 작업 구성 요소를 정의할 수 있습니다. 재사용 가능한 맷, 재사용 가능 변환 및 재사용 가능한 논리적 데이터 개체를 사용하여 작업의 작업 입력 및 작업 출력 요소를 정의할 수 있습니다.
2. 작업 매핑을 구성합니다.

데이터 통합 서비스가 SOAP 메시지와 입력 변환 및 출력 변환 포트 간에 데이터를 추출하는 방법을 구성합니다. 또한 작업 매핑 논리를 구성하고 각 작업 매핑을 테스트합니다.
3. 웹 서비스를 데이터 통합 서비스에 배포합니다.

웹 서비스를 응용 프로그램에 추가하고 응용 프로그램을 데이터 통합 서비스에 배포합니다. 데이터 통합 서비스에서 이미 실행 중인 웹 서비스가 포함된 응용 프로그램을 배포할 경우 데이터 통합 서비스가 웹 서비스의 서비스 이름에 번호를 첨부합니다.
4. 웹 서비스에 대한 관리 태스크를 수행합니다.

Administrator 도구에서 웹 서비스 속성 및 보안을 구성합니다.

SOAP 웹 서비스 클라이언트는 데이터 통합 서비스에서 실행 중인 SOAP 웹 서비스에 연결할 수 있습니다. 웹 서비스 클라이언트는 WSDL 콘텐츠를 사용하여 웹 서비스에 연결합니다. 웹 서비스 연결 개체를 사용하여 웹 서비스에 연결하도록 웹 서비스 소비자 변환을 구성할 수 있습니다.

SOAP 웹 서비스 예제

고객 데이터에 액세스하는 웹 서비스를 생성하거나, 고객 주소 데이터의 유효성을 검사하는 웹 서비스를 생성할 수 있습니다.

고객 데이터 액세스

Hypostores 고객 서비스 담당자가 로스앤젤레스 및 보스턴 사무실의 고객 데이터에 네트워크를 통해 액세스하고자 합니다. 고객 서비스 담당자는 고객 이름 또는 고객 ID를 기반으로 고객 세부 정보를 보고자 합니다. 회사 정책에 따라 네트워크를 통해 액세스하는 데이터는 보안 처리해야 합니다.

개발자 및 관리자는 다음 단계를 수행하여 고객 서비스에 필요한 데이터에 대한 액세스를 제공합니다.

1. Developer tool에서 개발자는 다음 작업이 포함된 웹 서비스를 생성합니다.
 - `getCustomerDetailsByName`

작업 입력에는 고객 이름에 대한 요소가 포함됩니다. 작업 출력에는 고객 이름을 기반으로 한 고객 세부 정보 요소가 포함됩니다.
 - `getCustomerDetailsById`

작업 입력에는 고객 ID에 대한 요소가 포함됩니다. 작업 출력에는 고객 ID를 기반으로 한 고객 세부 정보 요소가 포함됩니다.
2. 개발자는 다음 구성 요소를 사용하여 각 작업에 대한 작업 매핑을 구성합니다.
 - 입력 변환 및 출력 변환.
 - 로스앤젤레스 및 보스턴 사무실의 고객 데이터 단일 보기를 정의하는 논리적 데이터 개체에 대해 조회를 수행하는 조회 변환.

3. 개발자가 웹 서비스를 데이터 통합 서비스에 배포합니다.
4. **Administrator** 도구에서 관리자는 **HTTPS URL**을 사용하여 인증된 요청을 받을 수 있도록 **TLS** 및 메시지 계층 보안을 사용하는 웹 서비스를 구성합니다.
5. 관리자가 **WSDL URL**을 고객 서비스 담당자에게 전송하므로 고객 서비스 담당자가 웹 서비스에 연결할 수 있습니다.

고객 주소 데이터 유효성 검사

Hypostores 이행 부서에서 주문을 확정하기 전에 주소 데이터 유효성을 검사하고자 합니다. 주소 유효성 검사기 변환에서 입력 주소 데이터를 주소 참조 데이터와 비교하여 입력 주소의 정확성을 확인하고 해당 주소의 오류를 수정합니다.

개발자 및 관리자는 다음 단계를 수행하여 주소 유효성 검사 기능을 이행 부서에 제공합니다.

1. **Developer tool**에서 개발자는 주소 데이터를 입력으로 수신하고 유효성 검증된 주소 데이터를 출력으로 반환하는 주소 유효성 검사기 변환을 사용하여 맵렛을 생성합니다.
2. 개발자는 웹 서비스를 생성하고 맵렛을 사용하여 웹 서비스 작업을 생성합니다.
웹 서비스 생성 마법사를 사용하여 재사용 가능 개체에서 작업을 생성할 수 있습니다.
3. 개발자가 웹 서비스를 데이터 통합 서비스에 배포합니다.
4. 관리자가 **WSDL URL**을 이행 부서에 전송하므로 이행 부서가 웹 서비스에 연결할 수 있습니다.

웹 서비스에서 주소를 입력으로 허용하고 유효성 검사된 주소를 출력으로 반환합니다.

제 3 장

WSDL 데이터 개체

이 장에 포함된 항목:

- [WSDL 데이터 개체 개요WSDL 데이터 개체, 19](#)
- [WSDL 데이터 개체 개요 보기, 20](#)
- [WSDL 데이터 개체 고급 보기, 21](#)
- [WSDL 데이터 개체 가져오기, 21](#)
- [WSDL 동기화, 22](#)
- [인증서 관리, 23](#)

WSDL 데이터 개체 개요WSDL 데이터 개체

WSDL 데이터 개체는 WSDL 파일을 소스로 사용하는 실제 데이터 개체입니다. WSDL 데이터 개체를 사용하여 웹 서비스 또는 웹 서비스 소비자 변환을 작성할 수 있습니다. WSDL 파일을 가져와서 WSDL 데이터 개체를 작성합니다. WSDL 데이터 개체는 WSDL 파일을 소스로 사용하는 실제 데이터 개체입니다. WSDL 데이터 개체를 사용하여 웹 서비스 소비자 변환을 작성할 수 있습니다. WSDL 파일을 가져와서 WSDL 데이터 개체를 작성합니다.

WSDL 데이터 개체를 가져온 후 **개요** 및 **고급** 보기에서 일반 및 고급 속성을 편집할 수 있습니다. **WSDL** 보기에는 WSDL 파일 콘텐츠가 표시됩니다.

다음 그림에서는 샘플 WSDL 데이터 개체를 보여 줍니다.

The screenshot shows a software interface titled "Overview" for a WSDL service. It has three tabs: "Overview", "WSDL", and "Advanced". The "Overview" tab is active, displaying the following information:

- General**
 - Name: CURRENCY_CONVERTER
 - Description: (Empty text box)
 - Target Namespace: http://www.webserviceX.NET/
- Operations**

	Operation	Input	Output	Fault
1	Currency Convert...	1 input	1 output	0 fault
2	Currency Convert...	1 input	1 output	0 fault
3	Currency Convert...	1 input	1 output	0 fault
4	Currency Convert...	1 input	1 output	0 fault

WSDL을 가져올 때 다음 지침을 고려하십시오.

- WSDL 파일은 WSDL 1.1과 호환 가능해야 합니다.
- WSDL 파일은 유효해야 합니다.
- 웹 서비스 또는 웹 서비스 소비자 변환에 포함할 작업에서 문서/리터럴 인코딩을 사용해야 합니다. 웹 서비스 소비자 변환에 포함할 작업에서 문서/리터럴 인코딩을 사용해야 합니다. WSDL 파일의 모든 작업에서 문서/리터럴 이외의 인코딩 유형을 사용할 경우 WSDL 가져오기가 실패합니다.
- Developer tool이 WSDL 파일이 참조하는 스키마에 액세스할 수 있어야 합니다.
- WSDL 파일에 스키마가 포함되거나 외부 스키마가 있는 경우 Developer tool이 WSDL 데이터 개체 내에 포함된 스키마를 작성합니다.
- WSDL 파일이 다른 WSDL 파일을 가져올 경우 Developer tool이 두 개의 WSDL을 결합하여 WSDL 데이터 개체를 작성합니다.
- WSDL 파일에서 여러 작업을 정의할 경우 Developer tool이 모든 작업을 WSDL 데이터 개체에 포함합니다. WSDL 데이터 개체에서 웹 서비스를 작성할 경우 하나 이상의 작업을 포함할지 선택할 수 있습니다.

WSDL 데이터 개체 개요 보기

WSDL 데이터 개체 개요 보기에는 WSDL 및 WSDL의 작업에 대한 일반 정보가 표시됩니다.

다음 테이블에는 WSDL 데이터 개체에 대해 구성하는 일반 속성이 설명되어 있습니다.

속성	설명
이름	WSDL 데이터 개체의 이름입니다.
설명	WSDL 데이터 개체에 대한 설명입니다.

다음 테이블에는 WSDL 데이터 개체에 정의된 작업의 열이 설명되어 있습니다.

속성	설명
작업	WSDL이 작업에 대한 메시지 형식 및 프로토콜을 정의하는 위치입니다.
입력	작업 입력과 연관된 WSDL 메시지 이름입니다.
출력	작업 출력과 연관된 WSDL 메시지 이름입니다.
결함	작업 결함과 연관된 WSDL 메시지 이름입니다.

WSDL 데이터 개체 고급 보기

WSDL 데이터 개체 **고급** 보기에 WSDL 데이터 개체의 고급 속성이 표시됩니다.

다음 테이블에는 WSDL 데이터 개체의 고급 속성이 설명되어 있습니다.

속성	설명
연결	웹 서비스 소비자 변환에 대한 기본 웹 서비스 연결입니다.
파일 위치	WSDL 파일이 있는 위치입니다.

WSDL 데이터 개체 가져오기

WSDL에서 웹 서비스를 작성하거나 웹 서비스 소비자 변환을 작성하려면 WSDL 데이터 개체를 가져오십시오. 웹 서비스 소비자 변환을 작성하려면 WSDL 데이터 개체를 가져오십시오. WSDL 파일 또는 WSDL 위치를 가리키는 URI에서 WSDL 데이터 개체를 가져올 수 있습니다. SOAP 1.1 또는 SOAP 1.2 바인딩 작업이나 둘 다를 포함하는 WSDL 파일에서 WSDL 데이터 개체를 가져올 수 있습니다.

1. **파일 > 새로 만들기 > 데이터 개체**를 클릭합니다.
2. **WSDL 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 **WSDL 데이터 개체** 대화 상자가 나타납니다.
3. **WSDL** 옵션 옆에 있는 **찾아보기** 옵션을 클릭하고 WSDL 위치를 입력합니다. 그런 다음 **확인**을 클릭합니다.
WSDL 위치를 입력할 경우 WSDL 파일을 찾아보거나 WSDL에 대한 URI를 입력할 수 있습니다.
참고: URI에 영어 이외 문자가 포함된 경우 가져오기가 실패할 수 있습니다. URI를 브라우저의 주소 표시줄에 복사합니다. 위치를 다시 브라우저에 복사합니다. Developer 도구는 브라우저의 인코딩된 URI를 허용합니다.
4. WSDL 이름을 입력합니다.
5. **위치** 옵션 옆에 있는 **찾아보기**를 클릭하여 WSDL 데이터 개체를 가져올 프로젝트 또는 폴더 위치를 선택합니다.
6. **다음**을 클릭하여 WSDL의 작업을 확인합니다.

7. **마침**을 클릭합니다.

개체 탐색기 보기의 프로젝트 또는 폴더에서 **실제 데이터 개체** 아래에 데이터 개체가 표시됩니다.

WSDL 동기화

WSDL 파일이 변경될 경우 WSDL 데이터 개체를 동기화할 수 있습니다. WSDL 데이터 개체를 동기화할 경우 Developer 도구가 WSDL 파일에서 개체 메타데이터를 다시 가져옵니다.

WSDL 데이터 개체를 사용하여 웹 서비스 또는 웹 서비스 소비자 변환을 작성할 수 있습니다. WSDL 데이터 개체를 사용하여 웹 서비스 소비자 변환을 작성하십시오. WSDL 데이터 개체를 업데이트할 경우 Developer 도구에서 WSDL을 참조하는 개체를 업데이트하고, 사용자가 해당 개체를 열 때 변경됨으로 표시합니다. Developer 도구가 새 WSDL을 이전 WSDL과 비교할 때 이름 특성을 통해 WSDL 구성 요소를 식별합니다.

이름 특성이 변경되지 않은 경우 Developer 도구는 WSDL 구성 요소를 참조하는 개체를 업데이트합니다. 예를 들어, WSDL 파일을 편집하고 단순 요소 "CustID"의 유형을 xs:string에서 xs:integer로 변경하십시오. WSDL 데이터 개체를 동기화할 경우 Developer 도구가 CustID 요소를 참조하는 모든 웹 서비스 및 웹 서비스 소비자 변환의 요소 유형을 업데이트합니다. WSDL 데이터 개체를 동기화할 경우 Developer 도구가 CustID 요소를 참조하는 웹 서비스 소비자 변환의 요소 유형을 업데이트합니다.

이름 특성이 변경된 경우 WSDL 구성 요소를 참조하는 개체를 사용자가 열 때 Developer 도구가 해당 개체를 변경됨으로 표시합니다. 예를 들어, WSDL을 편집하고 요소 이름을 "Resp"에서 "RespMsg"로 변경하십시오. 그런 다음 WSDL을 동기화합니다. 요소를 참조하는 웹 서비스를 열 경우 Developer 도구가 편집기의 웹 서비스 이름에 별표를 표시하여 웹 서비스에 변경 사항이 포함되어 있음을 나타냅니다. Developer 도구에서 웹 서비스의 요소 이름을 업데이트하지만 새 요소가 포트에 매핑되는 방법을 확인할 수 없습니다. Resp 요소가 입력 변환 또는 출력 변환의 포트에 매핑된 경우 RespMsg 요소를 해당 포트에 매핑해야 합니다.

Developer 도구가 WSDL 데이터 개체를 업데이트하기 전에 WSDL 파일의 유효성을 검사합니다. WSDL 파일에 오류가 포함된 경우 Developer 도구가 파일을 가져오지 않습니다.

WSDL 데이터 개체 동기화

WSDL 파일이 변경될 경우 WSDL 데이터 개체를 동기화하십시오.

1. **개체 탐색기** 보기에서 WSDL 데이터 개체를 마우스 오른쪽 단추로 클릭하고, **동기화**를 선택합니다.

WSDL 데이터 개체 동기화 대화 상자가 나타납니다.

2. **WSDL** 필드 옆에 있는 **찾아보기**를 클릭하고 WSDL 위치를 입력합니다. 그런 다음 **확인**을 클릭합니다.

WSDL 위치를 입력할 경우 WSDL 파일을 찾아보거나 WSDL에 대한 URI를 입력할 수 있습니다.

참고: URI에 영어 이외 문자가 포함된 경우 가져오기가 실패할 수 있습니다. URI를 브라우저의 주소 표시줄에 복사합니다. 위치를 다시 브라우저에 복사합니다. Developer 도구는 브라우저의 인코딩된 URI를 허용합니다.

3. WSDL 이름 및 위치를 확인합니다.

4. **다음**을 클릭하여 WSDL의 작업을 확인합니다.

5. **마침**을 클릭합니다.

Developer 도구에서 WSDL을 참조하는 개체를 업데이트하고, 사용자가 해당 개체를 열었을 때 변경된 상태로 표시합니다.

인증서 관리

클라이언트 인증이 필요한 URL에서 WSDL 데이터 개체 및 스키마 개체를 가져오려면 Developer 도구에서 인증서를 사용해야 합니다.

기본적으로 URL을 호스트하는 서버가 트러스트된 인증서를 사용할 경우 Developer 도구는 클라이언트 인증이 필요한 URL에서 개체를 가져옵니다. URL을 호스트하는 서버에서 트러스트되지 않은 인증서를 사용할 경우 Developer 도구에 트러스트되지 않은 인증서를 추가하십시오. Developer 도구에 트러스트되지 않은 인증서를 추가하지 않은 경우 Developer 도구가 개체를 가져올 수 없습니다. 개체를 가져올 URL의 서버 관리자에게 인증서 파일 및 암호를 요청하십시오.

Developer 도구에 추가한 인증서가 Developer 도구 시스템에서 수행하는 가져오기에 적용됩니다. Developer 도구는 모델 리포지토리에 인증서를 저장하지 않습니다.

Informatica Developer 인증서 속성

트러스트되지 않은 인증서를 사용하여 클라이언트를 인증해야 하는 URL에서 개체를 가져오려는 경우 Developer tool에 인증서를 추가해야 합니다.

다음 테이블에는 인증서 속성이 설명되어 있습니다.

속성	설명
호스트 이름	URL을 호스트하는 서버의 이름입니다.
포트 번호	URL의 포트 번호입니다.
인증서 파일 경로	클라이언트 인증서 파일의 위치입니다.
암호	클라이언트 인증서 파일의 암호입니다.

Informatica Developer에 인증서 추가

인증서를 추가할 때는 트러스트되지 않은 인증서를 통한 클라이언트 인증이 필요한 URL에서 개체를 가져올 때 Developer tool에서 사용할 인증서 속성을 구성해야 합니다.

1. 창 > 기본 설정을 클릭합니다.
2. Informatica > 웹 서비스 > 인증서를 선택합니다.
3. 추가를 클릭합니다.
4. 인증서 속성을 구성합니다.
5. 확인을 클릭합니다.

제 4 장

스키마 개체

이 장에 포함된 항목:

- [스키마 개체 개요, 24](#)
- [스키마 개체 개요 보기, 24](#)
- [스키마 개체 스키마 보기, 25](#)
- [스키마 개체 고급 보기, 31](#)
- [스키마 개체 작성, 31](#)
- [스키마 업데이트, 32](#)
- [인증서 관리, 35](#)

스키마 개체 개요

스키마 개체는 사용자가 모델 리포지토리에 가져오는 계층 스키마입니다. 스키마를 가져오면 **Developer tool**에서 스키마 구성 요소를 볼 수 있습니다. **Avro**, **Parquet**, **XML** 또는 **JSON** 스키마를 가져올 수 있습니다.

Developer tool은 모델 리포지토리에서 스키마를 **.xsd** 파일로 변환합니다.

SOAP 웹 서비스를 작성할 때 계층 스키마를 기반으로 웹 서비스의 구조를 정의할 수 있습니다. **WSDL** 없이 웹 서비스를 작성한 경우 스키마에서 정의한 유형 및 요소를 기반으로 작업, 입력, 출력 및 결합 서명을 정의할 수 있습니다.

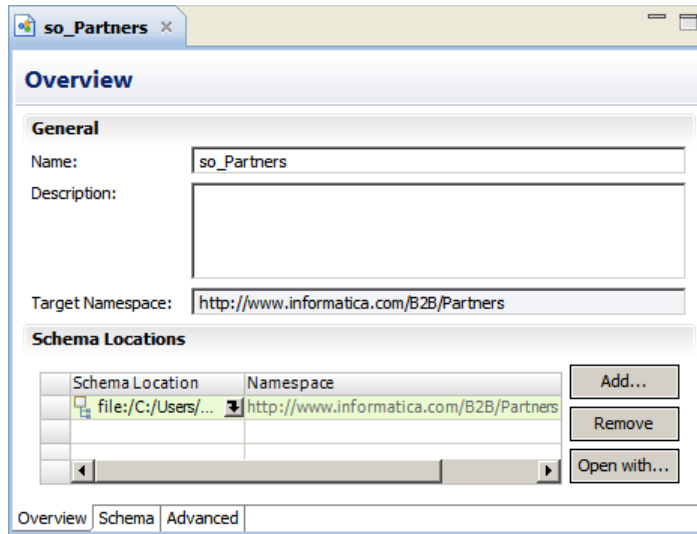
스키마를 가져온 경우 **개요** 보기에서 일반 스키마 속성을 편집할 수 있습니다. **고급** 보기에서 고급 속성을 편집합니다. **스키마** 보기에서 스키마 파일 콘텐츠를 봅니다.

스키마 개체 개요 보기

스키마 이름 또는 스키마 설명을 업데이트하고, 네임스페이스를 보고, 스키마 파일을 관리하려면 **개요** 보기를 선택하십시오.

개요 보기에서 스키마의 이름, 설명 및 대상 네임스페이스를 표시합니다. 스키마 이름과 설명을 편집할 수 있습니다. 대상 네임스페이스에서 스키마 구성 요소가 속하는 네임스페이스를 표시합니다. 대상 네임스페이스가 표시되지 않으면 스키마 구성 요소가 네임스페이스에 속하지 않습니다.

다음 그림은 스키마 개체의 개요 보기를 보여 줍니다.



스키마 위치 영역에 스키마 파일과 네임스페이스가 나열됩니다. 여러 루트 .xsd 파일을 추가할 수 있습니다. 스키마 파일에 다른 스키마 파일이 포함되거나 다른 스키마 파일을 가져올 경우 Developer 도구가 스키마에 하위 .xsd 파일을 포함합니다.

스키마 파일

여러 루트 수준의 .xsd 파일을 스키마 개체에 추가할 수 있습니다. 스키마 개체에서 루트 수준의 .xsd 파일을 제거할 수도 있습니다.

스키마 파일을 추가할 경우 사용자가 추가한 파일에서 가져오거나 그 파일에 포함된 모든 .xsd 파일을 Developer tool이 가져옵니다. Developer tool은 스키마 개체의 일부인 파일에 대해 사용자가 추가한 파일의 유효성을 검사합니다. 파일이 스키마 개체의 일부인 파일과 충돌할 경우 Developer tool에서 해당 파일의 추가를 허용하지 않습니다.

예를 들어, 스키마 개체에 루트 스키마 파일 "BostonCust.xsd"가 포함되어 있습니다. 루트 스키마 파일 "LACust.xsd"를 스키마 개체에 추가하고자 합니다. 두 스키마 파일이 모두 동일한 대상 네임스페이스를 갖고 있고 "Customer"라는 요소를 정의합니다. 스키마 파일 LACust.xsd를 스키마 개체에 추가하려고 하면 Developer tool에서 BostonCust.xsd 파일을 유지할지 또는 LACust.xsd 파일로 덮어쓸지 묻습니다.

루트 수준의 스키마 파일을 제거할 수 있습니다. 스키마 파일을 제거할 경우 스키마 파일에서 정의한 요소의 요소 유형을 Developer tool이 xs:string으로 변경합니다.

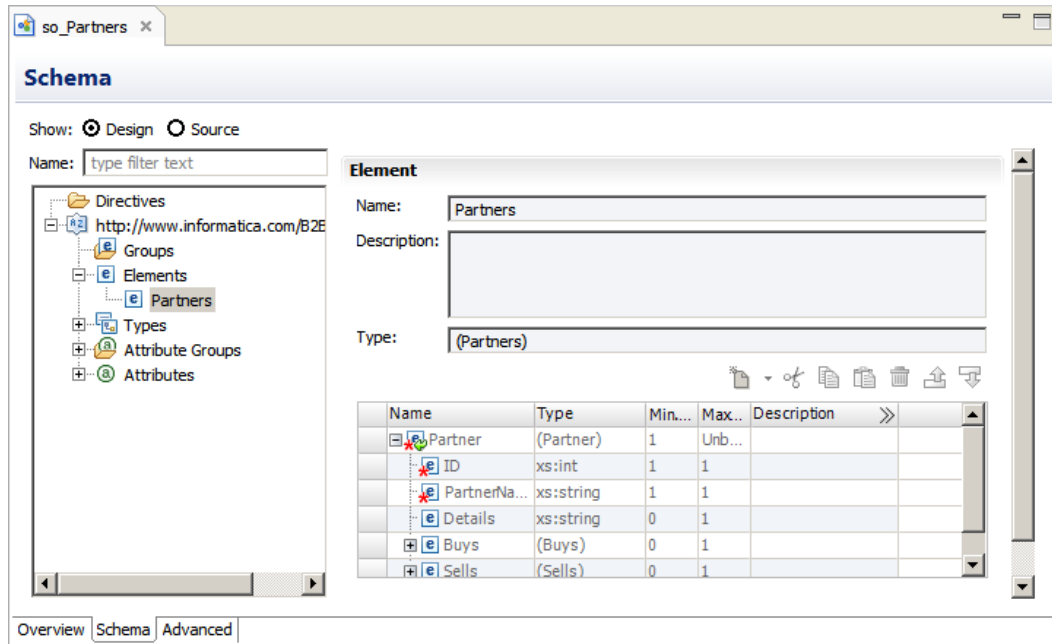
스키마 파일을 추가하려면 개요 보기를 선택하고 스키마 위치 목록 옆에 있는 추가 단추를 클릭합니다. 그런 다음 스키마 파일을 선택합니다. 스키마 파일을 제거하려면 파일을 선택하고 제거 단추를 클릭합니다.

스키마 개체 스키마 보기

스키마 보기에는 스키마의 그룹, 요소, 유형, 특성 그룹 및 특성의 목록이 알파벳 순서로 표시됩니다. 스키마 보기에서 그룹, 요소, 유형, 특성 그룹 또는 특성을 선택할 경우 오른쪽 패널에 속성이 표시됩니다. 스키마 보기에서 모든 .xsd 파일을 볼 수도 있습니다.

스키마 보기에서 스키마 개체의 네임스페이스 및 .xsd 파일의 목록을 제공합니다.

다음 그림은 스키마 개체의 **스키마** 보기를 보여 줍니다.



스키마 보기에서 다음 작업을 수행할 수 있습니다.

- 스키마 구성 목록을 보려면 **Directives** 폴더를 확장합니다. 네임스페이스, 접두사 및 위치를 보려면 목록에서 스키마 구성을 선택합니다.
- 네임스페이스 접두사, 생성된 접두사 및 위치를 보려면 네임스페이스를 선택합니다. 생성된 접두사를 변경할 수 있습니다.
- .xsd 파일 형식의 스키마 개체를 표시하려면 **소스**를 선택합니다. 스키마 개체에 다른 스키마가 포함된 경우 표시할 .xsd 파일을 선택할 수 있습니다.
- 그룹, 요소, 유형, 특성 그룹 및 특성의 알파벳 목록을 스키마의 각 네임스페이스에 표시하려면 **디자인**을 선택합니다. **이름** 필드에 하나 이상의 문자를 입력하여 그룹, 요소, 유형, 특성 그룹 및 특성을 이름별로 필터링할 수 있습니다.
- 요소 속성을 보려면 그룹, 요소, 유형, 특성 그룹 또는 특성을 선택합니다. **Developer tool**은 사용자가 선택한 개체를 기반으로 오른쪽 패널에 다른 필드를 표시합니다.

유형을 볼 때 유형이 다른 유형에서 파생되었는지 확인할 수 있습니다. 인터페이스에서 상위 유형을 표시합니다. 또한 인터페이스에서 하위 요소가 제한 또는 확장으로 값을 상속했는지 표시합니다.

네임스페이스 속성

네임스페이스 보기에는 선택된 네임스페이스의 접두사 및 위치가 표시됩니다.

각 스키마 파일과 연관된 네임스페이스는 다른 소스에서 가져 왔지만 동일한 이름을 가진 요소 간에 구별합니다. **URI(Uniform Resource Identifier)** 참조는 요소 및 특성 이름을 포함하는 파일 위치를 정의합니다.

둘 이상의 네임스페이스가 포함된 스키마를 가져올 경우 **Developer tool**이 네임스페이스를 스키마 개체에 추가합니다. 스키마 파일에 다른 스키마가 포함된 경우 해당 스키마의 네임스페이스도 포함됩니다.

Developer 도구에서 각 네임스페이스에 대해 작성된 접두사를 작성합니다. 스키마에 접두사가 포함되지 않은 경우 **Developer tool**은 네임스페이스 접두사 **tns0**를 생성하고 네임스페이스 접두사가 추가될 때마다 접두사 번호를 늘립니다. **Developer** 도구에서 네임스페이스 접두사 **xs**를 보유합니다. 네임스페이스 접두사 **xs**가 포함된 스키마를 가져오면 **Developer tool**이 생성된 접두사 **xs1**을 작성합니다. 스키마에 생성된 접두사 값이 포함된 경우 **Developer** 도구에서 접두사 번호를 증가시킵니다.

예를 들어, `Customer_Orders.xsd`에 네임스페이스가 있습니다. 스키마에 다른 스키마 `Customers.xsd`가 포함되어 있습니다. 고객 스키마에 다른 네임스페이스가 있습니다. **Developer** 도구는 `Customer_Orders` 네임스페이스에 `tns0` 접두사를 할당하고 고객 네임스페이스에 `tns1` 접두사를 할당합니다.

네임스페이스 위치 및 접두사를 보려면 **스키마** 보기에서 네임스페이스를 선택합니다.

둘 이상의 스키마 개체에서 웹 서비스를 작성한 경우 각 네임스페이스에 고유한 접두사가 있어야 합니다. 각 네임스페이스에 생성된 접두사를 수정할 수 있습니다.

요소 속성

요소는 단순 또는 복합 유형입니다. 복합 유형에는 다른 유형이 포함됩니다. **스키마** 보기에서 요소를 선택하면 **Developer tool**의 화면 오른쪽 패널에 하위 요소 및 속성이 나열됩니다.

다음 테이블에는 요소를 선택할 때 나타나는 요소 속성이 설명되어 있습니다.

속성	설명
이름	요소 이름입니다.
설명	유형에 대한 설명입니다.
유형	요소 유형입니다.

다음 테이블에는 요소를 선택할 때 나타나는 하위 요소 속성이 설명되어 있습니다.

속성	설명
이름	요소 이름입니다.
유형	요소 유형입니다.
최소 발생	인스턴스의 한 지점에서 요소가 발생할 수 있는 최소 횟수입니다.
최대 발생	인스턴스의 한 지점에서 요소가 발생할 수 있는 최대 횟수입니다.
설명	요소에 대한 설명입니다.

추가 하위 요소 속성을 보려면 설명 열에서 이중 화살표를 클릭하여 창을 확대합니다.

다음 테이블에는 설명 열을 확대할 때 표시되는 추가 하위 요소 속성이 설명되어 있습니다.

속성	설명
고정값	변경되지 않는 요소의 특정 값입니다.
Null 가능	Null 값을 가질 수 있는 요소입니다. Null 요소에는 요소 태그가 있지만 값과 콘텐츠는 없습니다.
요약	요소가 추상 유형입니다. 인스턴스에 해당 유형에서 파생된 유형이 포함되어야 합니다. 파생된 요소 유형이 없으면 추상 유형이 유효한 유형이 아닙니다.
최소값	인스턴스의 요소의 최소값입니다.
최대값	인스턴스의 요소의 최대값입니다.

속성	설명
최소 길이	요소의 최소 길이입니다. 요소 유형에 따라 길이는 바이트, 문자 또는 항목입니다.
최대 길이	요소의 최대 길이입니다. 요소 유형에 따라 길이는 바이트, 문자 또는 항목입니다.
열거	요소에 대해 유효한 모든 값 목록입니다.
패턴	유효한 요소 값을 정의하는 식 패턴입니다.

고급 요소 속성

요소에 대한 고급 속성을 보려면 **스키마** 보기에서 요소를 선택합니다. **고급**을 클릭합니다.

다음 테이블에는 요소의 고급 속성이 설명되어 있습니다.

속성	설명
요약	요소가 추상 유형입니다. SOAP 메시지에는 해당 유형에서 파생된 유형이 포함되어야 합니다. 파생된 요소 유형이 없으면 추상 유형이 유효한 유형이 아닙니다.
블록	파생된 요소가 이 요소 대신 계층에 나타나는 것을 방지합니다. 블록 값에는 "#all" 또는 확장, 제한 또는 대체가 포함된 목록이 포함될 수 있습니다.
최종	스키마가 단순 유형을 파생된 유형으로 확장 또는 제한하지 못하도록 지정합니다.
대체 그룹	요소로 대체할 요소의 이름입니다.
Nilable	Null 값을 가질 수 있는 요소입니다. Null 요소에는 요소 태그가 있지만 값과 콘텐츠는 없습니다.

단순 유형 속성

단순 유형 요소는 구조화되지 않은 텍스트를 포함하는 요소입니다. **스키마** 보기에서 단순 유형 요소를 선택한 경우 오른쪽 패널에 단순 유형 요소에 대한 정보가 표시됩니다.

다음 테이블에는 단순 유형에 대해 표시할 수 있는 속성이 설명되어 있습니다.

속성	설명
유형	요소의 이름입니다.
설명	요소에 대한 설명입니다.
다양성	단순 유형이 합집합, 목록, anyType 또는 원자성인지 정의합니다. 원자성 요소에는 다른 요소 또는 특성이 포함되지 않습니다.
멤버 유형	UNION 구성의 유형 목록입니다.
항목 유형	요소 유형입니다.
기본	원자성 요소의 기본 유형입니다(예: 정수 또는 문자열).
최소 길이	요소의 최소 길이입니다. 요소 유형에 따라 길이는 바이트, 문자 또는 항목입니다.

속성	설명
최대 길이	요소의 최대 길이입니다. 요소 유형에 따라 길이는 바이트, 문자 또는 항목입니다.
공백 축소	선행 및 후행 공백을 제거합니다. 여러 공백을 하나의 공백으로 축소합니다.
열거	유형을 유효한 값 목록으로 제한합니다.
패턴	유형을 패턴 식에서 정의한 값으로 제한합니다.

단순 유형 고급 속성

단순 유형에 대한 고급 속성을 보려면 **스키마** 보기에서 단순 유형을 선택합니다. **고급**을 클릭합니다.

단순 유형 속성 아래에 고급 속성이 표시됩니다.

다음 테이블에는 단순 유형의 고급 속성이 설명되어 있습니다.

속성	설명
최종	스키마가 단순 유형을 파생된 유형으로 확장 또는 제한하지 못하도록 지정합니다.

복합 유형 속성

복합 유형은 다른 요소 및 특성을 포함하는 요소입니다. 복합 유형에는 단순 또는 복합 유형의 요소가 포함되어 있습니다. **스키마** 보기에서 복합 유형을 선택한 경우 **Developer** 도구가 화면의 오른쪽 패널에 하위 요소 및 하위 요소 속성을 나열합니다.

다음 테이블에는 복합 유형 속성이 설명되어 있습니다.

속성	설명
이름	유형 이름입니다.
설명	유형에 대한 설명입니다.
상속 위치	상위 유형의 이름입니다.
상속 주체	제한 또는 확장입니다. 복합 유형은 상위 유형에서 파생됩니다. 복합 유형에서 상위의 요소 또는 특성이 줄어들 수 있습니다. 또는 요소 및 특성을 추가할 수도 있습니다.

복합 유형의 각 요소에 대한 속성을 보려면 설명 열의 이중 화살표를 클릭하여 창을 확장합니다.

복합 유형 고급 속성

복합 유형에 대한 고급 속성을 보려면 **스키마** 보기에서 요소를 선택합니다. **고급**을 클릭합니다.

다음 테이블에는 복합 요소 또는 유형에 대한 고급 속성이 설명되어 있습니다.

속성	설명
요약	요소가 추상 유형입니다. SOAP 메시지에는 해당 유형에서 파생된 유형이 포함되어야 합니다. 파생된 요소 유형이 없으면 추상 유형이 유효한 유형이 아닙니다.

속성	설명
블록	파생된 요소가 이 요소 대신 스키마에 나타나는 것을 방지합니다. 블록 값에는 "#all" 또는 확장, 제한 또는 대체가 포함된 목록이 포함될 수 있습니다.
최종	스키마가 단순 유형을 파생된 유형으로 확장 또는 제한하지 못하도록 지정합니다.
대체 그룹	요소로 대체할 요소의 이름입니다.
Nilable	Null 값을 가질 수 있는 요소입니다. Null 요소에는 요소 태그가 있지만 값과 콘텐츠는 없습니다.

특성 속성

특성은 단순 유형입니다. 요소 및 복합 유형에는 특성이 포함됩니다. 글로벌 특성은 스키마의 일부로 나타납니다. **스키마** 보기에서 글로벌 특성을 선택하면 **Developer tool** 화면의 오른쪽 패널에 특성 속성 및 관련된 유형 속성이 나열됩니다.

다음 테이블에는 특성 속성이 설명되어 있습니다.

속성	설명
이름	특성 이름입니다.
설명	특성에 대한 설명입니다.
유형	특성 유형입니다.
값	특성 유형의 값입니다. 특성 유형의 값이 고정된 값인지 기본값이 있는지 여부를 나타냅니다. 정의된 값이 없는 경우 속성에 기본값=0이 표시됩니다.

다음 테이블에는 유형 속성이 설명되어 있습니다.

속성	설명
최소 길이	유형의 최소 길이입니다. 유형에 따라 길이는 바이트, 문자 또는 항목입니다.
최대 길이	유형의 최대 길이입니다. 유형에 따라 길이는 바이트, 문자 또는 항목입니다.
공백 축소	선행 및 후행 공백을 제거합니다. 여러 공백을 하나의 공백으로 축소합니다.
열거	유형을 유효한 값 목록으로 제한합니다.
패턴	유형을 패턴 식에서 정의한 값으로 제한합니다.

스키마 개체 고급 보기

스키마 개체에 대한 고급 속성을 확인하십시오.

다음 테이블에는 스키마 개체에 대한 고급 속성이 설명되어 있습니다.

이름	값	설명
elementFormDefault	정규화 또는 비 정규화	요소에 네임스페이스가 포함되어야 하는지 결정합니다. 스키마에서는 접두사 또는 대상 네임스페이스 선언으로 요소를 한정합니다. 비정규화 값은 요소에 네임스페이스가 필요하지 않음을 의미합니다.
attributeFormDefault	정규화 또는 비 정규화	로컬에서 선언한 특성에 네임스페이스가 있어야 하는지 결정합니다. 스 키마에서는 접두사 또는 대상 네임스페이스 선언으로 특성을 한정합니 다. 비정규화 값은 특성에 네임스페이스가 필요하지 않음을 의미합니다.
파일 위치	.xsd 파일에 대 한 전체 경로	.xsd 파일을 가져올 때 해당 파일의 위치입니다.

스키마 개체 작성

계층 스키마 파일 또는 샘플 파일을 가져와 리포지토리에 스키마 개체를 작성할 수 있습니다.

1. **Object Explorer** 보기에서 프로젝트 또는 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 스키마**를 클릭합니다.
새 스키마 대화 상자가 나타납니다.
3. 스키마 파일을 가져오려면 **스키마에서 작성**을 선택한 다음 계층 스키마 파일을 찾아서 선택합니다.
URI 또는 찾아볼 파일 시스템의 위치를 입력할 수 있습니다. **Developer** 도구에서 선택된 스키마의 유효성
을 검사합니다. 유효성 검사 메시지를 검토합니다. **Avro, Parquet, JSON** 또는 **.xsd** 스키마 파일을 선택할
수 있습니다.
참고: URI에 영어 이외 문자가 포함된 경우 가져오기가 실패할 수 있습니다. URI를 브라우저의 주소 표시줄
에 복사합니다. 위치를 다시 브라우저에 복사합니다. **Developer tool**은 브라우저의 인코딩된 URI를 허용합
니다.
4. 샘플 파일에서 스키마를 작성하려면 **샘플 파일에서 작성**을 선택한 다음 계층 파일을 찾아서 선택합니다.
Avro, Parquet, JSON 또는 **XML** 파일을 선택할 수 있습니다.
참고: **Avro, Parquet, JSON** 또는 **XML** 콘텐츠가 포함된 다른 확장명을 가진 파일을 선택하면 마법사가 해
당 파일 콘텐츠를 인식합니다.
5. 필요에 따라 스키마 이름을 변경합니다.
6. **다음**을 클릭하여 스키마의 요소 및 유형 목록을 표시합니다.
7. 스키마를 가져오려면 **마침**을 클릭합니다.
Object Explorer 보기의 스키마 개체에 스키마가 표시됩니다. **Developer tool**이 스키마를 **.xsd** 파일로 저
장합니다.
8. 스키마 네임스페이스에 대해 생성된 접두사를 변경하려면 **개체 탐색기** 보기에서 네임스페이스를 선택합니
다. **네임스페이스** 보기의 **생성된 접두사** 속성을 변경합니다.

스키마 업데이트

요소, 특성, 유형 또는 기타 스키마 구성 요소가 변경될 때 스키마 개체를 업데이트할 수 있습니다. 스키마 개체를 업데이트할 때 Developer tool이 스키마를 사용하는 개체를 업데이트합니다.

다음 방법을 통해 스키마 개체를 업데이트할 수 있습니다.

스키마를 동기화합니다.

Developer tool 외에서 스키마 파일을 업데이트할 때 스키마 개체를 동기화합니다. 스키마 개체를 동기화할 경우 Developer tool이 변경 사항이 포함된 모든 스키마 .xsd 파일을 다시 가져옵니다.

스키마 파일을 편집합니다.

Developer tool에서 파일을 업데이트하려고 할 때 스키마 파일을 편집합니다. 스키마 파일을 편집하면 Developer tool이 .xsd 파일에 사용된 편집기에서 파일을 엽니다. 다른 편집기에서 파일을 열거나 Developer tool에서 .xsd 파일에 대한 기본 편집기를 설정할 수 있습니다.

스키마를 사용하여 웹 서비스에서 요소 유형을 정의할 수 있습니다. 웹 서비스의 WSDL에 포함된 스키마를 업데이트할 경우 Developer tool이 웹 서비스를 업데이트하고 사용자가 웹 서비스를 열 때 웹 서비스가 변경된 것으로 표시합니다. Developer tool이 새 스키마를 이전 스키마와 비교할 때 이름 특성을 통해 스키마 구성 요소를 식별합니다.

이름 특성이 변경되지 않은 경우 Developer tool이 웹 서비스를 스키마 변경 사항으로 업데이트합니다. 예를 들어, Developer tool 내에서 스키마 파일을 편집하고 "Item" 요소에 대한 maxOccurs 특성을 10에서 무제한으로 변경합니다. 파일을 저장할 경우 Developer tool이 Item 요소를 참조하는 모든 웹 서비스의 maxOccurs 특성을 업데이트합니다.

이름 특성이 변경되면 웹 서비스가 열릴 때 Developer tool에서 웹 서비스가 변경된 것으로 표시합니다. 예를 들어, Developer tool 외부에서 스키마를 편집하고 복합 요소 유형의 이름을 "Order"에서 "CustOrder"로 변경합니다. 그런 다음 스키마를 동기화합니다. 요소를 참조하는 웹 서비스를 열 경우 Developer tool이 편집기의 웹 서비스 이름에 별표를 표시하여 웹 서비스에 변경 사항이 포함되어 있음을 나타냅니다. Developer tool이 CustOrder 요소 유형을 웹 서비스에 추가하지만 Order 요소 유형을 제거하지 않습니다. Developer tool이 Order 요소에 대한 유형을 더 이상 확인할 수 없기 때문에 요소 유형을 xs:string으로 변경합니다.

스키마 동기화

스키마 구성 요소가 변경될 경우 스키마 개체를 동기화할 수 있습니다. 스키마 개체를 동기화할 경우 Developer 도구가 스키마 파일에서 개체 메타데이터를 다시 가져옵니다.

Developer 도구 이외에서 스키마 개체를 복합 변경할 경우 스키마 동기화를 사용하십시오. 예를 들어, 다음 작업을 수행한 후에 스키마를 동기화할 수 있습니다.

- 여러 스키마 파일을 변경합니다.
- 스키마에서 스키마 파일을 추가 또는 제거합니다.
- 가져오기 또는 포함 요소를 변경합니다.

Developer 도구가 스키마 개체를 업데이트하기 전에 스키마 파일의 유효성을 검사합니다. 스키마 파일에 오류가 포함된 경우 Developer 도구가 파일을 가져오지 않습니다.

스키마 개체를 동기화하려면 **개체 탐색기** 보기에서 스키마 개체를 마우스 오른쪽 단추로 클릭하고 **동기화**를 선택합니다.

스키마 파일 편집

Developer 도구 내에서 스키마 파일을 편집하여 스키마 구성 요소를 업데이트할 수 있습니다.

적은 수의 파일의 일부만 업데이트하려면 Developer 도구에서 스키마 파일을 편집하십시오. 예를 들어, 스키마 파일에 다음 내용 중 일부만 업데이트할 수 있습니다.

- 요소의 minOccurs 또는 maxOccurs 특성을 변경합니다.
- 복합 유형에 특성을 추가합니다.
- 단순 개체 유형을 변경합니다.

스키마 파일을 편집할 경우 Developer 도구가 스키마 파일의 임시 사본을 편집기에서 엽니다. .xsd 파일에 사용하는 시스템 편집기를 사용하여 스키마 파일을 편집하거나 다른 편집기를 선택할 수 있습니다. .xsd 파일에 대한 Developer 도구 기본 편집기를 설정할 수도 있습니다. 임시 스키마 파일을 편집한 후 저장하십시오.

Developer 도구가 스키마 개체를 업데이트하기 전에 임시 파일의 유효성을 검사합니다. 스키마 파일에 오류가 포함되어거나 스키마 개체의 다른 스키마 파일과 충돌하는 구성 요소가 포함된 경우 Developer 도구가 파일을 가져오지 않습니다.

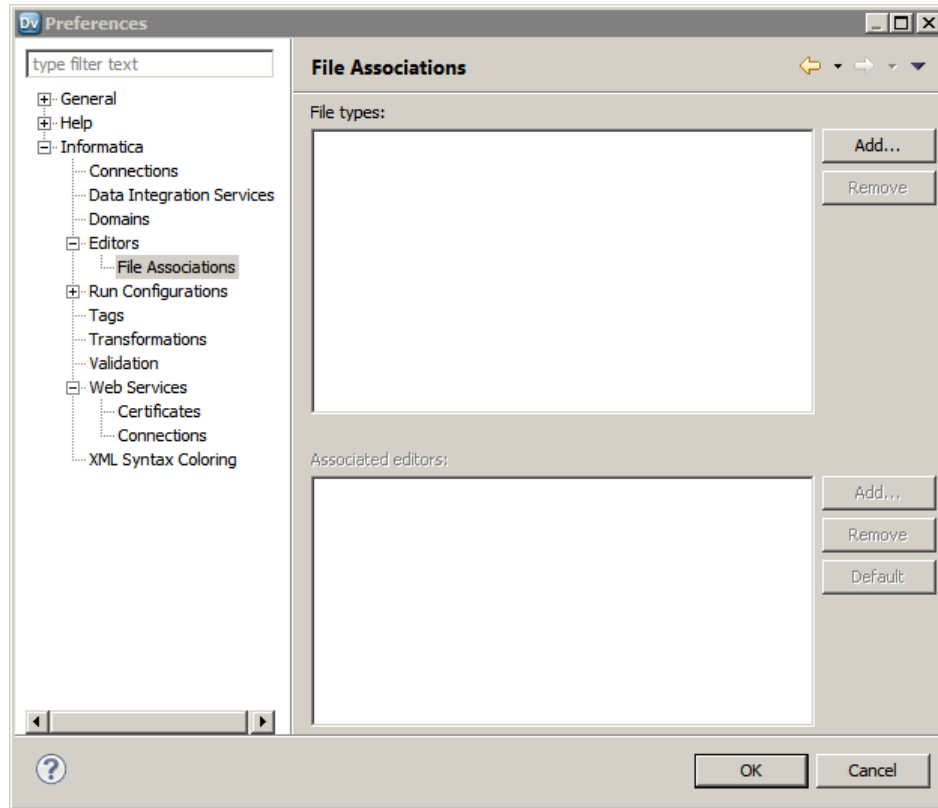
참고: 임시 스키마 파일을 편집 및 저장할 경우 Developer 도구가 **스키마 위치** 목록에 표시되는 스키마 파일을 업데이트하지 않습니다. Developer 도구에서 스키마 파일을 편집한 후 스키마 개체를 동기화할 경우 동기화 작업에서 사용자 편집 내용을 덮어씁니다.

기본 스키마 파일 편집기 설정

스키마 파일을 편집할 때 Developer 도구에서 여는 기본 편집기를 설정할 수 있습니다.

1. **창 > 기본 설정**을 클릭합니다.
기본 설정 대화 상자가 표시됩니다.
2. **편집기 > 파일 연결**을 클릭합니다.

기본 설정 대화 상자의 파일 연결 페이지가 나타납니다.



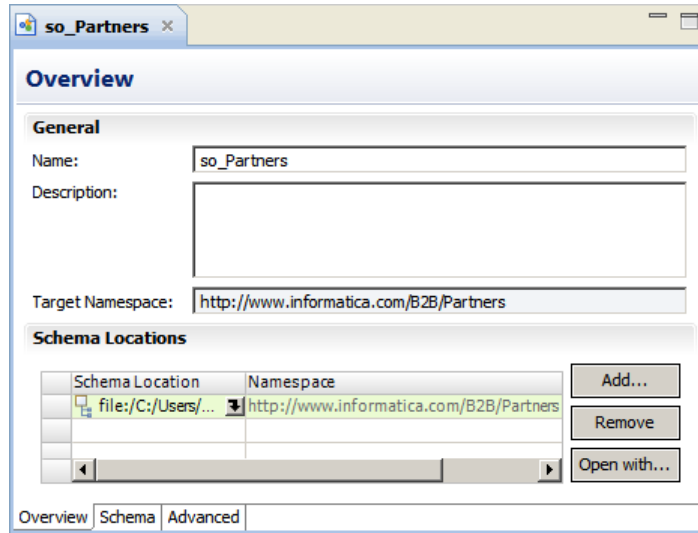
3. 파일 유형 영역 옆에 있는 **추가**를 클릭합니다.
파일 유형 추가 대화 상자가 나타납니다.
4. 파일 유형으로 .xsd를 입력하고 **확인**을 클릭합니다.
5. 연결된 편집기 영역 옆에 있는 **추가**를 클릭합니다.
편집기 선택 대화 상자가 나타납니다.
6. 편집기 목록에서 편집기를 선택하거나 **찾아보기**를 클릭하여 다른 편집기를 선택한 다음, **확인**을 클릭합니다.
선택한 편집기가 **연결된 편집기** 목록에 표시됩니다.
7. 필요에 따라 다른 편집기를 **연결된 편집기** 목록에 추가합니다.
8. 여러 편집기를 추가한 경우 기본 편집기를 변경할 수 있습니다. 편집기를 선택하고 **기본값**을 클릭합니다.
9. **확인**을 클릭합니다.

스키마 파일 편집

스키마 개체의 모든 스키마 파일을 편집할 수 있습니다.

1. 스키마 개체를 엽니다.
2. **개요** 보기를 선택합니다.

스키마 개체의 **개요** 보기가 나타납니다.



3. 스키마 위치 목록에서 스키마 파일을 선택합니다.
4. **연결 프로그램**을 클릭하고 다음 옵션 중 하나를 선택합니다.

옵션	설명
시스템 편집기	.xsd 파일에 대해 운영 체제가 사용하는 편집기에서 스키마 파일이 열립니다.
기본 편집기	Developer tool에 설정된 기본 편집기에서 스키마 파일이 열립니다. 이 옵션은 기본 편집기를 설정한 경우에 나타납니다.
기타	스키마 파일을 열 편집기를 선택합니다.

Developer tool에서 스키마 파일의 임시 사본이 열립니다.

5. 임시 스키마 파일을 업데이트하고 변경 사항을 저장한 다음 편집기를 닫습니다.
Developer tool에서 스키마 개체를 업데이트하라는 메시지를 표시합니다.
6. 스키마 개체를 업데이트하려면 **스키마 개체 업데이트**를 클릭합니다.
Developer tool에서 수행한 변경 사항으로 스키마 파일이 업데이트됩니다.

인증서 관리

클라이언트 인증이 필요한 URL에서 WSDL 데이터 개체 및 스키마 개체를 가져오려면 Developer 도구에서 인증서를 사용해야 합니다.

기본적으로 URL을 호스트하는 서버가 트러스트된 인증서를 사용할 경우 Developer 도구는 클라이언트 인증이 필요한 URL에서 개체를 가져옵니다. URL을 호스트하는 서버에서 트러스트되지 않은 인증서를 사용할 경우 Developer 도구에 트러스트되지 않은 인증서를 추가하십시오. Developer 도구에 트러스트되지 않은 인증서를 추가하지 않은 경우 Developer 도구가 개체를 가져올 수 없습니다. 개체를 가져올 URL의 서버 관리자에게 인증서 파일 및 암호를 요청하십시오.

Developer 도구에 추가한 인증서가 Developer 도구 시스템에서 수행하는 가져오기에 적용됩니다. Developer 도구는 모델 리포지토리에 인증서를 저장하지 않습니다.

Informatica Developer 인증서 속성

트러스트되지 않은 인증서를 사용하여 클라이언트를 인증해야 하는 URL에서 개체를 가져오려는 경우 Developer tool에 인증서를 추가해야 합니다.

다음 테이블에는 인증서 속성이 설명되어 있습니다.

속성	설명
호스트 이름	URL을 호스트하는 서버의 이름입니다.
포트 번호	URL의 포트 번호입니다.
인증서 파일 경로	클라이언트 인증서 파일의 위치입니다.
암호	클라이언트 인증서 파일의 암호입니다.

Informatica Developer에 인증서 추가

인증서를 추가할 때는 트러스트되지 않은 인증서를 통한 클라이언트 인증이 필요한 URL에서 개체를 가져올 때 Developer tool에서 사용할 인증서 속성을 구성해야 합니다.

1. **창 > 기본 설정**을 클릭합니다.
2. **Informatica > 웹 서비스 > 인증서**를 선택합니다.
3. **추가**를 클릭합니다.
4. 인증서 속성을 구성합니다.
5. **확인**을 클릭합니다.

제 5 장

SOAP 웹 서비스를 생성하는 방법

이 장에 포함된 항목:

- [SOAP 웹 서비스 생성 개요, 37](#)
- [유형 및 요소, 38](#)
- [웹 서비스 개요 보기, 38](#)
- [웹 서비스 WSDL 보기, 40](#)
- [WSDL 데이터 개체에서 웹 서비스 작성, 40](#)
- [SOAP 웹 서비스를 수동으로 생성, 42](#)

SOAP 웹 서비스 생성 개요

SOAP 웹 서비스를 생성할 경우 리포지토리에 웹 서비스 개체를 생성합니다. WSDL 데이터 개체에서 SOAP 웹 서비스를 생성하거나, SOAP 웹 서비스를 수동으로 생성할 수 있습니다. 각 SOAP 웹 서비스에는 하나 이상의 작업이 있을 수 있습니다.

WSDL 데이터 개체에서 SOAP 웹 서비스를 생성할 경우 웹 서비스에 포함할 작업을 WSDL 데이터 개체에서 선택합니다. 각 작업에서 SOAP 1.1 바인딩 또는 SOAP 1.2 바인딩을 사용할 수 있지만 둘 다 사용할 수는 없습니다.

수동으로 웹 서비스를 생성할 경우 작업을 생성한 다음 각 작업에 대한 작업 입력, 작업 출력 및 작업 결함의 요소를 정의하십시오. 재사용 가능 개체를 사용하여 작업에 대한 작업 입력 및 작업 출력의 요소를 정의할 수 있습니다.

웹 서비스 개체에는 **개요** 보기와 **WSDL** 보기가 있습니다. **개요** 보기에서 작업을 생성 및 구성할 수 있습니다.

WSDL 보기에서 WSDL 파일 콘텐츠를 볼 수 있습니다.

웹 서비스를 생성한 후 각 작업에 대한 작업 매핑을 구성하십시오. 필요에 따라 웹 서비스에 작업을 추가할 수 있습니다.

유형 및 요소

WSDL 데이터 개체에서 웹 서비스를 작성할 경우 WSDL 데이터 개체가 각 작업의 요소 및 요소 유형을 정의합니다. 웹 서비스를 수동으로 작성한 경우 요소를 정의하거나 스키마 개체의 유형을 사용하여 요소를 정의할 수 있습니다.

작업 구성 요소에 **anyType** 요소, 임의 요소, **anyAttribute** 특성, 파생된 유형 특성 또는 대체 그룹이 포함된 경우 작업 매핑을 구성할 때 하나 이상의 유형, 요소 또는 특성을 선택해야 합니다. 예를 들어, 작업 입력에 **anyType** 요소가 포함된 경우 입력 변환을 구성할 때 하나 이상의 유형을 선택하십시오.

스키마 개체를 사용하여 요소 유형을 정의할 경우 **Developer** 도구가 스키마 개체를 웹 서비스의 WSDL에 포함합니다. 스키마 개체를 삭제하거나 웹 서비스 WSDL에서 스키마 개체에 대한 링크를 삭제한 경우 **Developer** 도구가 스키마 개체에서 정의한 요소의 요소 유형을 **xs:string**으로 변경합니다. 요소 유형을 다른 유형으로 업데이트할 수 있습니다.

수동으로 웹 서비스를 작성한 경우 작업 구성 요소를 정의하기 위해 웹 서비스에서 사용하는 각 네임스페이스마다 고유하게 작성된 접두사 값이 필요합니다. 예를 들어, 웹 서비스에서 **schemaA** 스키마 개체 및 **schemaB** 스키마 개체를 사용하여 유형을 정의할 경우 **schemaA** 및 **schema B**가 임의의 네임스페이스에 대해 동일하게 생성된 접두사 값을 가질 수 없습니다.

웹 서비스 개요 보기

웹 서비스 **개요** 보기에는 웹 서비스에 대한 일반 정보와 웹 서비스 작업 및 작업 구성 요소에 대한 세부 정보가 표시됩니다.

일반 속성

다음 테이블에는 웹 서비스에 구성된 일반 속성이 설명되어 있습니다.

속성	설명
이름	웹 서비스 개체 이름입니다.
설명	웹 서비스에 대한 설명입니다.
네임스페이스	웹 서비스의 targetNamespace 입니다. 웹 서비스가 WSDL 데이터 개체와 연결된 경우 이 필드는 읽기 전용입니다.
접두사	targetNamespace 의 접두사입니다. 웹 서비스가 WSDL 데이터 개체와 연결된 경우 이 필드는 읽기 전용입니다.
WSDL 데이터 개체	웹 서비스와 연결된 WSDL 데이터 개체입니다. WSDL 데이터 개체에서 웹 서비스가 작성된 경우 이 속성이 표시됩니다.
서비스 이름	서비스 이름입니다. 기본값은 웹 서비스의 이름 또는 연결된 WSDL 데이터 개체에 정의된 서비스 이름입니다. 웹 서비스가 WSDL 데이터 개체와 연결된 경우 이 필드는 읽기 전용입니다.

작업 영역

작업 영역의 왼쪽에서 작업을 선택한 경우 오른쪽에 세부 정보가 표시됩니다. 작업 영역의 왼쪽에 관련된 입력, 출력 및 결함이 포함된 작업의 트리 보기가 표시됩니다.

다음 테이블에는 작업 영역 오른쪽에 표시되는 작업 및 입력/출력 속성이 설명되어 있습니다.

속성	설명
작업 이름	작업의 이름입니다.
설명	작업에 대한 설명입니다.
바인딩 유형	작업의 바인딩 유형입니다.
입력	작업 입력과 연관된 WSDL 메시지 이름입니다.
출력	작업 출력과 연관된 WSDL 메시지 이름입니다.
요소 이름	작업 입력 또는 출력에서 참조하는 XSD 요소의 이름입니다.
이름	작업 입력 또는 출력에서 참조하는 요소의 하위 요소입니다.
유형	요소 유형입니다.
최소 발생	XML 인스턴스의 한 지점에서 요소가 발생할 수 있는 최소 횟수입니다.
최대 발생	XML 인스턴스의 한 지점에서 요소가 발생할 수 있는 최대 횟수입니다.
설명	요소에 대한 설명입니다.

추가 하위 요소 속성을 보려면 설명 열에서 이중 화살표를 클릭하여 창을 확대합니다.

다음 테이블에는 설명 열을 확대할 때 표시되는 추가 하위 요소 속성이 설명되어 있습니다.

속성	설명
고정값	변경되지 않는 요소의 특정 값입니다.
Null 가능	Null 값을 가질 수 있는 요소입니다. Null 요소에는 요소 태그가 있지만 값과 콘텐츠는 없습니다.
요약	요소가 추상 유형입니다. XML 인스턴스에 해당 유형에서 파생된 유형이 포함되어야 합니다. 파생된 요소 유형이 없으면 추상 유형이 유효한 유형이 아닙니다.
최소값	XML 인스턴스의 요소에 대한 최소값입니다.
최대값	XML 인스턴스의 요소에 대한 최대값입니다.
최소 길이	요소의 최소 길이입니다. 요소 유형에 따라 길이는 바이트, 문자 또는 항목입니다.
최대 길이	요소의 최대 길이입니다. 요소 유형에 따라 길이는 바이트, 문자 또는 항목입니다.
열거	요소에 대해 유효한 모든 값 목록입니다.
패턴	유효한 요소 값을 정의하는 식 패턴입니다.

작업 매핑 영역

다음 테이블에는 작업 매핑에 대한 열이 설명되어 있습니다.

속성	설명
작업 매핑	작업 매핑 이름입니다.
작업	작업 이름입니다.
입력	작업 입력과 연관된 WSDL 메시지 이름입니다.
출력	작업 출력과 연관된 WSDL 메시지 이름입니다.
결함	작업 결함과 연관된 WSDL 메시지 이름입니다.

웹 서비스 WSDL 보기

웹 서비스의 **WSDL** 보기에서 WSDL 파일의 콘텐츠를 미리 볼 수 있습니다.

WSDL 개체에서 웹 서비스를 작성할 경우 WSDL 파일의 콘텐츠를 볼 수 있습니다.

수동으로 웹 서비스를 작성할 경우 **Developer** 도구가 작업 입력, 작업 출력, 작업 결함 및 헤더의 구성을 기반으로 WSDL 파일의 콘텐츠를 생성합니다. WSDL의 스키마에 요소를 추가하거나 WSDL의 스키마에 있는 요소를 편집할 수 있습니다. 가져온 스키마 개체를 WSDL에서 삭제할 수도 있습니다.

각 작업 입력, 작업 출력, 작업 결함 및 헤더는 WSDL의 메시지에 해당합니다. 작업 결함과 헤더는 WSDL에서 메시지를 공유할 수 있습니다. WSDL 스키마의 요소를 사용하여 결함 또는 헤더를 작성할 경우 메시지 이름이 요소 이름과 동일합니다.

WSDL 디자인 또는 WSDL 소스를 볼 수 있습니다. WSDL 디자인에서 WSDL 콘텐츠의 계층 보기를 표시합니다. WSDL 소스에서 WSDL 콘텐츠를 XML 형식으로 표시합니다.

Developer 도구에서 WSDL을 미리 보려면 웹 서비스의 **WSDL** 보기에서 **표시** 필드 옆에 있는 **소스**를 선택합니다.

WSDL 데이터 개체에서 웹 서비스 작성

WSDL 데이터 개체에서 **SOAP** 웹 서비스를 작성할 경우 WSDL에서 하나 이상의 작업 및 각 작업 입력, 작업 출력 및 작업 결함의 요소를 정의합니다.

WSDL에 선택할 수 있는 여러 작업이 있는 경우 작업을 추가할 수 있습니다. WSDL에 정의되지 않은 작업을 작성할 수 없습니다.

WSDL 데이터 개체에서 웹 서비스를 작성하기 전에 WSDL 데이터 개체를 작성해야 합니다. WSDL은 SOAP 1.1 또는 SOAP 1.2를 기반으로 해야 합니다. WSDL 데이터 개체에서 웹 서비스를 작성하기 전에 **WSDL 데이터 개체에서 웹 서비스 작성** 마법사를 사용하여 WSDL 데이터 개체를 작성할 수 있습니다.

WSDL 데이터 개체에서 웹 서비스를 작성할 경우 웹 서비스는 WSDL 데이터 개체에 종속됩니다. WSDL 데이터 개체를 삭제할 경우 작업 매핑이 유효하지 않고 WSDL 데이터 개체를 웹 서비스와 연결해야 합니다. WSDL 데이

터 개체가 리포지토리에서 삭제될 경우 Developer tool에서 웹 서비스와 WSDL 데이터 개체 간 연결을 제거합니다.

1단계. WSDL 데이터 개체에서 웹 서비스 작성

WSDL 데이터 개체에서 웹 서비스를 작성할 경우 Developer 도구는 WSDL 데이터 개체에 정의된 정보를 사용하여 웹 서비스의 각 작업에 대한 작업 매핑을 작성합니다.

1. **개체 탐색기** 보기에서 프로젝트나 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 데이터 서비스**를 클릭합니다.
새 데이터 서비스 대화 상자가 표시됩니다.
3. **WSDL 데이터 개체의 웹 서비스**를 클릭한 다음, **다음**을 클릭합니다.
4. WSDL 데이터 개체를 작성하려면 새 **WSDL 데이터 개체**를 클릭합니다.
 - a. 위치 옵션 옆에 있는 **찾아보기**를 클릭하고 WSDL 위치를 입력합니다. **확인**을 클릭합니다.
 - b. WSDL 이름을 입력합니다.
 - c. **마침**을 클릭합니다.
5. 웹 서비스에 대한 이름을 입력합니다.
6. **WSDL 데이터 개체** 옵션 옆에 있는 **찾아보기**를 클릭하여 WSDL 데이터 개체를 선택합니다.
7. **작업** 옵션 옆에 있는 **찾아보기**를 클릭하여 웹 서비스에 포함할 하나 이상의 작업을 선택합니다.
WSDL 데이터 개체에 SOAP 1.1 및 SOAP 1.2 형식의 여러 바인딩이 포함될 수 있습니다. 이 바인딩 중 하나에서만 특정 작업을 선택할 수 있습니다.
8. **다음**을 클릭합니다.
9. 기본적으로 선택한 작업의 SOAP 버전이 **작업 유형**으로 표시됩니다.
10. 필요에 따라 각 작업 입력의 **매핑 입력** 탭을 선택하여 작업 입력의 데이터를 출력 포트에 매핑합니다.
입력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
11. 필요에 따라 각 작업 출력의 **매핑 출력** 탭을 선택하여 입력 포트의 데이터를 작업 출력에 매핑합니다.
출력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
12. 필요에 따라 각 작업 결함의 **매핑 결함** 탭을 선택하여 입력 포트의 데이터를 작업 결함에 매핑합니다.
결함 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
13. **마침**을 클릭합니다.

2단계. 웹 서비스에 작업 추가

필요에 따라 웹 서비스에 작업을 추가할 수 있습니다.

1. **개체 탐색기** 보기에서 WSDL 데이터 개체에 종속되는 웹 서비스를 엽니다.
2. **개요** 보기를 선택합니다.
3. **작업** 섹션에서 **작업 매핑** 영역을 선택합니다.
4. **작업 매핑** 영역에서 **선택**을 클릭합니다.
하나 이상의 작업 선택 대화 상자가 표시됩니다.
5. 추가할 작업을 선택하고 **확인**을 클릭합니다.

WSDL 데이터 개체를 웹 서비스와 연결

WSDL 데이터 개체를 웹 서비스와 연결하려면 WSDL 데이터 개체를 선택하고 각 작업 매핑을 WSDL 데이터 개체에 정의된 작업과 연결하십시오.

WSDL 데이터 개체에서 웹 서비스를 작성할 경우 웹 서비스는 WSDL 데이터 개체에 종속됩니다. WSDL 데이터 개체를 삭제할 경우 작업 매핑이 유효하지 않고 WSDL 데이터 개체를 웹 서비스와 연결해야 합니다.

1. **개체 탐색기** 보기에서 WSDL 데이터 개체에 종속되는 웹 서비스를 엽니다.
2. **개요** 보기를 선택합니다.
3. **WSDL 데이터 개체** 옵션 옆에 있는 **찾아보기**를 클릭합니다.
WSDL 데이터 개체 선택 대화 상자가 나타납니다.
4. WSDL 데이터 개체를 선택하고 **확인**을 클릭합니다.
5. **작업 매핑** 영역에서 WSDL 작업을 각 웹 서비스 작업 매핑과 연결합니다.
 - a. 작업 매핑 행의 **작업** 열에서 마우스 오른쪽 단추를 클릭하고 **작업 선택**을 클릭합니다.
작업 선택 대화 상자가 나타납니다.
 - b. 작업을 선택하고 **확인**을 클릭합니다.

SOAP 웹 서비스를 수동으로 생성

WSDL 데이터 개체 없이 SOAP 웹 서비스를 생성할 경우 웹 서비스 속성, 작업 및 작업 구성 요소를 정의합니다. 작업 구성 요소에는 작업 입력, 작업 출력 및 작업 결합이 포함됩니다.

수동으로 웹 서비스를 생성할 경우 하나 이상의 작업을 정의할 수 있습니다. 수동으로 작업을 생성할 경우 SOAP 바인딩 유형을 선택하고 스키마 개체의 요소 및 유형을 사용하여 작업 요소를 정의할 수 있습니다. 또한 맵렛, 재사용 가능 변환, 논리적 데이터 개체, 플랫폼 파일 데이터 개체 또는 관계형 데이터 개체를 사용하여 작업을 생성할 수도 있습니다. 재사용 가능 개체에서 작업을 생성할 경우 작업에 포함할 필드를 개체에서 선택하십시오.

Developer tool은 사용자가 정의한 웹 서비스 속성 및 작업을 사용하여 WSDL 파일의 콘텐츠를 생성합니다. 웹 서비스를 생성한 후 WSDL 파일의 콘텐츠를 미리 볼 수 있습니다.

1단계. 웹 서비스를 수동으로 작성

웹 서비스 작성 마법사를 사용하여 웹 서비스를 작성합니다.

1. **개체 탐색기** 보기에서 프로젝트나 폴더를 선택합니다.
 2. **파일 > 새로 만들기 > 데이터 서비스**
새 데이터 서비스 대화 상자가 표시됩니다.
 3. **웹 서비스**를 클릭합니다.
새 웹 서비스 대화 상자가 나타납니다.
 4. 웹 서비스에 대한 이름을 입력합니다.
 5. 필요에 따라 네임스페이스 및 네임스페이스 접두사를 입력합니다.
 6. **다음**을 클릭합니다.
새 웹 서비스 대화 상자가 나타납니다.
- 웹 서비스를 작성한 후 작업을 작성해야 합니다.

2단계. 작업 작성

수동으로 웹 서비스를 작성할 경우 웹 서비스 작성 마법사 또는 새 작업 마법사에서 작업을 작성할 수 있습니다. 다음 테이블에는 작업 작성 옵션이 나열되어 있습니다.

작업 작성 옵션	경로
웹 서비스 작성 마법사 사용.	새로 만들기 > 데이터 서비스 > 웹 서비스 를 클릭합니다. 재사용 가능 개체 또는 빈 개체에서. 참고: 재사용 가능 개체는 실제 데이터 개체, 맵셋 또는 변환일 수 있습니다.
새 작업 마법사 사용.	개체 탐색기 또는 아웃라인 보기에서 웹 서비스를 마우스 오른쪽 단추로 클릭하고 새로 만들기 > 작업을 클릭합니다.
새 작업 마법사 사용.	웹 서비스를 작성한 다음 웹 서비스 개요 보기를 선택합니다. 작업 영역에서 새로 만들기 단추 옆에 있는 화살표를 클릭하고 새 작업 을 선택합니다.
새 작업 마법사 사용.	재사용 가능 개체를 개요 보기의 작업 영역으로 끕니다.

재사용 가능 개체에서 작업 작성

웹 서비스를 작성하고 재사용 가능 개체에서 작업을 작성하려면 **웹 서비스 작성** 마법사를 사용하십시오.

논리적 데이터 개체, 플랫폼 파일 데이터 개체 및 관계형 데이터 개체의 데이터를 조회하는 작업을 작성할 수 있습니다. 맵셋 또는 재사용 가능 변환에서 작업을 작성할 수도 있습니다. 맵셋 또는 재사용 가능 변환에서 작업 입력 및 출력의 요소를 정의합니다. 웹 서비스 소비자 변환에서 작업을 작성할 수 없습니다.

웹 서비스를 작성하는 단계를 완료한 후 **웹 서비스 작성** 마법사 내에서 데이터 개체, 맵셋 또는 재사용 가능 변환으로부터 작업을 작성할 수 있습니다.

1. 새 **웹 서비스** 대화 상자에서 **새로 만들기** 단추 옆의 화살표를 클릭합니다. 그런 다음 **작업 > 재사용 가능 개체에서 작성**을 선택합니다.
재사용 가능 개체 선택 대화 상자가 나타납니다.
참고: **개체 탐색기**에서 웹 서비스를 열고 **개요 > 작업** 영역을 선택하여 데이터 개체에서 작업을 작성할 수도 있습니다. **새로 만들기** 단추 옆에 있는 화살표를 클릭합니다.
2. 개체를 선택하고 **확인**을 클릭합니다.
3. 필요에 따라 작업 이름과 설명을 입력합니다.
Developer 도구가 사용자가 제공한 작업 이름을 사용하여 작업 입력 및 작업 출력의 이름을 정의합니다.
4. 작업의 바인딩 유형을 선택합니다.
기본값은 **SOAP 1.1**입니다. 작업을 작성 및 저장한 후에는 작업의 바인딩 유형을 변경할 수 없습니다.
5. 작업의 속성을 보고 구성하려면 작업 입력을 클릭합니다.
 - a. **작업 입력** 탭에서 작업 입력 필드를 선택합니다.
참고: 개체에 둘 이상의 입력 그룹이 있는 경우 **작업 입력** 탭에서 입력 그룹 및 작업 입력 필드를 선택합니다.
 - b. 필요에 따라 각 요소의 최소 및 최대 발생 횟수를 구성합니다.
 - c. 필요에 따라 **매핑 입력** 탭을 클릭하여 작업 입력의 데이터를 출력 포트에 매핑하는 방법을 구성합니다.
입력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.

6. 작업 출력을 클릭하여 해당 속성을 표시 및 구성합니다.
 - a. **작업 출력** 탭에서 작업 출력 필드를 선택합니다.
참고: 개체에 둘 이상의 출력 그룹이 있는 경우 **작업 출력** 탭에서 출력 그룹 및 작업 출력 필드를 선택합니다.
 - b. 필요에 따라 각 요소의 최소 발생 횟수를 구성합니다.
 - c. 필요에 따라 **매핑 출력** 탭을 클릭하여 입력 포트의 데이터를 작업 출력에 매핑하는 방법을 구성합니다.
 출력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
7. 작업에서 사용자 정의 결합을 전송할 경우 **새로 만들기** 단추 옆에 있는 화살표를 클릭한 다음 **결합**을 클릭합니다.
 결합에 대한 요소를 작성하도록 선택하거나 재사용 가능한 요소를 선택할 수 있습니다.
8. 작업 결합을 클릭하여 해당 속성을 표시 및 구성합니다.
매핑 결합 탭을 클릭하여 입력 포트의 데이터를 작업 결합에 매핑하는 방법을 구성할 수 있습니다. 결합 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
9. 1에서 8까지 단계를 반복하여 추가 작업을 작성 및 구성합니다.
10. **마침**을 클릭합니다.
 Developer 도구에서 사용자가 작업을 구성한 방법에 따라 각 작업에 대한 작업 매핑을 작성합니다.
 필요에 따라 요소 또는 사전 정의된 결합을 작성할 수 있습니다.

수동으로 작업 작성

작업을 정의하고 각 작업의 작업 입력, 작업 출력 및 작업 결합을 정의하려면 **웹 서비스 작성** 마법사를 사용하십시오.

웹 서비스를 작성하는 단계를 완료한 후 **웹 서비스 작성** 마법사 내에서 작업을 작성 및 구성할 수 있습니다.

1. 새 **웹 서비스** 대화 상자에서 **새로 만들기** 단추 옆의 화살표를 클릭합니다. 그런 다음 **작업 > 빈 항목으로 작성**을 선택합니다.
 - a. 작업 이름을 입력합니다.
 Developer 도구에서 작업 이름을 사용하여 작업 입력 및 작업 출력에 대한 이름을 정의합니다.
 - b. 작업에 대한 SOAP 버전을 **바인딩 유형**으로 선택합니다.
 - c. 결합 메시지를 정의하려면 **새로 만들기** 단추 옆에 있는 화살표를 클릭한 다음 **결합**을 클릭합니다.
 결합 메시지에 대한 요소를 작성하도록 선택하거나 재사용 가능한 요소를 선택할 수 있습니다. 마법사에서 결합 메시지의 세부 정보 요소에 요소를 추가합니다.
2. 작업 입력을 클릭하여 해당 속성을 표시 및 구성합니다.
 - a. **작업 입력** 탭을 클릭합니다.
 - b. 요소를 추가하려면 **새로 만들기** 단추 옆에 있는 화살표를 클릭한 다음 **요소**를 클릭합니다. 하위 요소를 추가하려면 요소를 선택하고, **새로 만들기** 단추 옆의 화살표를 클릭한 후, **하위 요소**를 클릭합니다.
 - c. 각 요소에 대한 유형을 지정하려면 **유형** 필드에서 선택 단추를 클릭합니다. XSD 유형 또는 스키마 개체 유형을 선택하고 **확인**을 클릭합니다.
팁: **유형** 필드를 클릭하고 선택할 유형의 처음 몇 자를 입력합니다. 입력한 문자로 시작하는 이름을 가진 XSD 유형 목록이 표시됩니다.
 - d. 각 요소의 최소 및 최대 발생 횟수를 구성합니다.
 - e. 필요에 따라 각 요소에 대한 설명을 입력합니다.

- f. 필요에 따라 **매핑 입력** 탭을 클릭하여 작업 입력의 데이터를 출력 포트에 매핑합니다.
입력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
3. 작업 결함을 클릭하여 해당 속성을 표시 및 구성합니다.
매핑 결함 탭을 클릭하여 입력 포트의 데이터를 작업 결함에 매핑하는 방법을 구성할 수 있습니다. 결함 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
4. 작업 출력을 클릭하여 해당 속성을 표시 및 구성합니다.
 - a. **작업 출력** 탭을 클릭합니다.
 - b. 요소를 추가하려면 **새로 만들기** 단추 옆에 있는 화살표를 클릭한 다음 **요소**를 클릭합니다. 하위 요소를 추가하려면 요소를 선택하고, **새로 만들기** 단추 옆의 화살표를 클릭한 다음, **하위 요소**를 클릭합니다.
 - c. 각 요소에 대한 유형을 지정하려면 **유형** 필드에서 선택 단추를 클릭합니다. 그런 다음 XSD 유형 또는 스키마 개체 유형을 선택하고 **확인**을 클릭합니다.
 - d. 각 요소의 최소 및 최대 발생 횟수를 구성합니다.
 - e. 필요에 따라 각 요소에 대한 설명을 입력합니다.
 - f. 필요에 따라 **매핑 출력** 탭을 클릭하여 입력 포트의 데이터를 작업 출력에 매핑합니다.
출력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
5. 1에서 4까지 단계를 반복하여 다른 작업을 추가 및 구성합니다.
6. **마침**을 클릭합니다.
Developer 도구에서 사용자가 작업을 구성한 방법에 따라 각 작업에 대한 작업 매핑을 작성합니다.
필요에 따라 요소 또는 사전 정의된 결함을 작성할 수 있습니다.

새 작업 마법사에서 작업 작성

웹 서비스를 수동으로 작성한 후 작업을 작성하려면 새 작업 마법사를 사용하십시오.

1. 작업을 작성하는 방법을 선택합니다.
 - 개체 탐색기에서 웹 서비스를 마우스 오른쪽 단추로 클릭하고 **새로 만들기 > 작업**을 선택합니다.
 - 아웃라인 보기에서 웹 서비스를 마우스 오른쪽 단추로 클릭하고 **새로 만들기 > 작업**을 선택합니다.
 - **개체 탐색기**에서 웹 서비스를 열고 **개요 > 작업** 영역을 선택합니다. **새로 만들기** 단추 옆에 있는 화살표를 클릭합니다.

새 작업 대화 상자가 나타납니다.
2. 작업 이름과 선택적 설명을 입력합니다.
Developer 도구에서 작업 이름을 사용하여 작업 입력 및 작업 출력에 대한 이름을 정의합니다.
3. **다음**을 클릭합니다.
4. 작업의 바인딩 유형을 선택합니다.
기본값은 **SOAP 1.1**입니다. 작업을 작성 및 저장한 후에는 작업의 바인딩 유형을 변경할 수 없습니다.
5. 필요에 따라 각 작업 입력의 **매핑 입력** 탭을 선택하여 작업 입력의 데이터를 출력 포트에 매핑합니다.
입력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
6. 필요에 따라 각 작업 출력의 **매핑 출력** 탭을 선택하여 입력 포트의 데이터를 작업 출력에 매핑합니다.
출력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.
7. 필요에 따라 **새로 만들기** 단추 옆의 화살표를 클릭하여 작업에 대한 결함을 작성합니다.
결함 메시지에 대한 요소를 작성하도록 선택하거나 재사용 가능한 요소를 선택할 수 있습니다. 마법사에서 결함 메시지의 세부 정보 요소에 요소를 추가합니다. 결함 변환을 구성할 때 데이터를 매핑할 수도 있습니다.

8. **마침**을 클릭합니다.

필요에 따라 요소 또는 사전 정의된 결함을 작성할 수 있습니다.

재사용 가능 개체 끌기

재사용 가능 개체에서 작업을 작성하려면 재사용 가능 개체를 **개요** 보기의 **작업** 영역으로 끌어서 놓습니다.

1. **개체 탐색기** 보기에서 웹 서비스를 엽니다.

2. **개요** 보기에서 **작업** 영역을 선택합니다.

3. **개체 탐색기** 보기의 재사용 가능 개체를 **개요** 보기의 **작업** 영역으로 끌어서 놓습니다.

새 작업 대화 상자가 나타납니다.

4. 작업의 바인딩 유형을 선택합니다.

기본값은 **SOAP 1.1**입니다. 작업을 작성 및 저장한 후에는 작업의 바인딩 유형을 변경할 수 없습니다.

5. 필요에 따라 각 작업 입력의 **매핑 입력** 탭을 선택하여 작업 입력의 데이터를 출력 포트에 매핑합니다.

입력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.

6. 필요에 따라 각 작업 출력의 **매핑 출력** 탭을 선택하여 입력 포트의 데이터를 작업 출력에 매핑합니다.

출력 변환을 구성할 때 데이터를 매핑할 수도 있습니다.

7. 필요에 따라 **새로 만들기** 단추 옆의 화살표를 클릭하여 작업에 대한 결함을 작성합니다.

결함 변환을 구성할 때 데이터를 매핑할 수도 있습니다.

8. **마침**을 클릭합니다.

필요에 따라 요소 또는 사전 정의된 결함을 작성할 수 있습니다.

3단계. 요소 작성

필요에 따라 웹 서비스의 **WSDL** 보기에서 요소를 작성할 수 있습니다. 웹 서비스의 **WSDL** 보기에서 작성한 요소를 사용하여 작업 결함 또는 헤더를 작성할 수 있습니다. 하나의 요소에 여러 요소와 하위 요소가 포함될 수 있습니다.

요소를 작성하기 전에 **웹 서비스 작성** 마법사에서 웹 서비스를 작성하고 하나 이상의 작업을 정의해야 합니다.

1. 웹 서비스를 엽니다.

2. **WSDL** 보기를 선택합니다.

3. WSDL의 **디자인**을 표시합니다. 그런 다음 스키마 섹션에서 요소 또는 요소 위의 네임스페이스 항목을 선택합니다.

4. **새 요소** 단추를 클릭합니다.

5. 요소를 요소에 추가하려면 **새로 만들기** 단추 옆의 화살표를 클릭한 다음, **요소**를 클릭합니다.

6. 하위 요소를 요소에 추가하려면 요소를 선택하고, **새로 만들기** 단추 옆의 화살표를 클릭한 다음, **하위 요소**를 클릭합니다.

7. 필요에 따라 각 요소의 유형을 구성합니다.

a. **유형** 필드에서 **선택** 단추를 클릭합니다.

b. XSD 유형 또는 스키마 개체 유형을 선택하고 **확인**을 클릭합니다.

8. 필요에 따라 각 요소에 대한 최소 및 최대 발생 횟수를 구성합니다.

9. 필요에 따라 각 요소에 대한 설명을 입력합니다.

필요에 따라 사전 정의된 결함을 작성할 수 있습니다.

요소의 계층 수준 변경

요소를 작성한 후에 계층 수준을 변경할 수 있습니다. 작업을 수동으로 작성하는 경우 **작업 입력** 탭 및 **작업 출력** 탭을 마우스 오른쪽 단추로 클릭하여 요소 또는 하위 요소의 제거, 이동 또는 복사와 같은 작업을 수행합니다.

요소 또는 하위 요소를 끌어 요소의 계층을 변경할 수도 있습니다.

4단계. 사전 정의된 결합 작성

필요에 따라 결합을 작성하여 SOAP 응답 내에서 사용자 정의 오류를 전송할 수 있습니다. 사전 정의된 결합을 작성할 경우 **Developer** 도구가 결합 변환을 작업 매핑에 추가합니다.

웹 서비스를 작성하고 하나 이상의 작업을 정의하십시오. 작업을 작성할 때 결합을 작성하거나 작업에 결합을 추가할 수 있습니다. 하나의 작업에 여러 결합이 있을 수 있습니다.

1. 웹 서비스를 엽니다.
2. **아웃라인** 보기에서 작업 매핑을 선택합니다.
편집기에 작업 매핑이 표시됩니다.
3. **속성** 보기에서 **작업** 탭을 클릭합니다.
4. **새로 만들기** 단추 옆에 있는 화살표를 클릭합니다.
5. **결합**을 클릭합니다.
6. 요소를 작성하거나 요소를 재사용할지 선택합니다.
 - 결합에 대한 요소를 작성하려면 **새 요소 작성**을 선택합니다. 하위 요소 및 다중 발생 요소를 작성할 수 있습니다.
 - 결합에 대한 요소를 재사용하려면 **기존 요소 재사용**을 선택합니다.

Developer 도구가 결합 메시지 계층의 세부 정보 요소에 요소를 추가합니다.

5단계. 헤더 작성

필요에 따라 SOAP 메시지의 헤더 내에 데이터를 수신 또는 전송할 헤더를 작성할 수 있습니다. 작업 입력 또는 작업 출력에 대한 헤더를 작성할 수 있습니다. HTTP POST 메서드를 사용할 수 있습니다.

1. 웹 서비스를 엽니다.
2. **아웃라인** 보기에서 작업 매핑을 선택합니다.
편집기에 작업 매핑이 표시됩니다.
3. **속성** 보기에서 **작업** 탭을 클릭합니다.
4. 헤더를 작업 입력 또는 작업 출력에 추가할지 선택합니다.
 - 헤더를 작업 입력에 추가하려면 입력을 선택합니다.
 - 헤더를 작업 출력에 추가하려면 출력을 선택합니다.
5. **새로 만들기** 단추 옆에 있는 화살표를 클릭합니다.
6. **헤더**를 클릭한 다음 요소를 작성 또는 재사용할지 선택합니다.
 - 헤더에 대한 요소를 작성하려면 **새 요소 작성**을 선택합니다.
 - 헤더에 대한 요소를 재사용하려면 **기존 요소 재사용**을 선택합니다.

제 6 장

작업 매핑

이 장에 포함된 항목:

- [작업 매핑 개요, 48](#)
- [작업 매핑의 일반 탭, 49](#)
- [작업 매핑 작업 탭, 49](#)
- [작업 매핑 고급 탭, 49](#)
- [입력 변환, 49](#)
- [출력 변환, 52](#)
- [결함 변환, 54](#)
- [결함 처리, 57](#)
- [작업 매핑 테스트, 61](#)
- [사용자 지정된 보기 옵션, 61](#)

작업 매핑 개요

Informatica SOAP 웹 서비스에는 작업 매핑이 있습니다. 작업 매핑은 웹 서비스 클라이언트를 위해 웹 서비스 작업을 수행합니다.

작업 매핑은 입력 변환, 출력 변환 및 여러 개의 결함 변환을 포함할 수 있습니다. 입력, 출력 및 결함 변환은 SOAP 메시지를 처리합니다. 매핑에는 클라이언트가 요청한 웹 서비스 작업에 따라 데이터를 검색, 변환 또는 업데이트하는 다른 변환도 포함될 수 있습니다.

Developer tool에서 웹 서비스를 생성한 후에 웹 서비스의 각 작업에 대한 작업 매핑을 구성합니다. 작업 매핑은 작업의 논리를 나타냅니다.

작업 매핑을 구성할 때는 SOAP 요청에서 수신한 데이터에 대한 데이터 통합 서비스의 처리 방법을 정의합니다. SOAP 요청은 작업 매핑에 연결된 바인딩 작업에서 사용하는 바인딩 유형에 따라 SOAP 1.1 또는 SOAP 1.2 형식일 수 있습니다.

입력 변환은 웹 서비스 클라이언트의 SOAP 요청을 수신한 다음 매핑의 변환 다운스트림으로 데이터를 반환합니다. 입력 변환은 클라이언트가 요청한 작업을 수행합니다.

출력 변환은 클라이언트에 반환할 데이터를 수신합니다. 출력 변환은 SOAP 메시지를 생성하여 클라이언트에 전송합니다.

오류가 발생할 경우 데이터 통합 서비스가 결함을 생성합니다. 데이터 통합 서비스에서는 결함 변환의 사용자 정의 결함을 반환합니다. 사용자 정의 결함은 두 가지 유형의 결함인 미리 정의된 결함과 일반 결함으로 구성됩니다.

작업 매핑을 구성하려면 다음 단계를 완료합니다.

1. 입력, 출력 및 결합 변환을 구성합니다.
2. 작업 논리를 구현하는 데 필요한 추가 변환을 생성하고 구성합니다.
3. 포트를 연결합니다.
4. 매핑의 유효성을 검사하고 저장합니다.

작업 매핑 **속성** 보기의 **일반** 탭, **작업** 탭 및 **고급** 탭에서 작업 매핑 속성을 보고 구성할 수 있습니다.

작업 매핑의 일반 탭

작업 매핑 **속성** 보기의 **일반** 탭에서 작업 매핑 이름과 설명을 구성합니다.

일반 탭에는 작업 매핑에 연결된 작업의 이름도 표시됩니다.

작업 매핑 작업 탭

작업 매핑 **속성** 보기의 **작업** 탭에서 작업 속성을 조회하거나 구성하십시오.

웹 서비스를 수동으로 작성한 경우 **작업** 탭을 사용하여 결합을 정의하거나 작업 입력 또는 출력을 업데이트할 수 있습니다. 또한 작업 입력이나 작업 출력에 헤더를 추가할 수도 있습니다.

작업 매핑 고급 탭

속성 보기의 **고급** 탭에서 작업 매핑 고급 속성을 구성하십시오. 고급 속성을 구성할 때 데이터 통합 서비스가 SOAP 요청의 XML 유효성을 검사할지 여부를 지정할 수 있습니다.

다음 테이블에는 작업 매핑에 대한 고급 속성이 설명되어 있습니다.

속성	설명
XML 스키마 유효성 검사	런타임 시 SOAP 요청 메시지의 유효성을 검사합니다. 올바르지 않은 XML에 대한 오류 또는 유효성 검사 없음 을 선택합니다. XML이 올바르지 않을 경우 데이터 통합 서비스가 SOAP 응답에서 결합을 반환하고 웹 서비스 런타임 로그에 오류를 기록합니다.

입력 변환

입력 변환은 웹 서비스 WSDL의 입력 요소 및 헤더 요소를 나타냅니다. 입력 변환은 클라이언트의 SOAP 요청을 수신합니다. XML 메시지를 관계형 데이터 그룹으로 구문 분석하여 작업 매핑의 다른 변환에 전달합니다.

웹 서비스에 대한 작업 입력을 정의하면 **Developer tool**이 입력 변환을 작성합니다.

입력 변환의 **포트** 탭을 사용하여 작업 입력 계층을 보고 출력 포트를 정의하고 작업 입력의 데이터를 출력 포트에 매핑할 수 있습니다. 작업 입력 계층은 **SOAP** 요청 메시지 계층을 정의합니다.

관계형 데이터 그룹을 개별 출력 포트로 반환하지 않고 전체 **SOAP** 요청을 **XML**로 매핑할 수 있습니다. **SOAP** 요청을 **XML**로 매핑하면 데이터 통합 서비스에서 전체 **SOAP** 메시지를 하나의 출력 포트로 반환합니다.

입력 변환 포트 탭

포트 탭에서 출력 그룹을 정의하고, 출력 포트를 정의하며, 작업 입력의 노드를 출력 포트에 매핑하십시오.

작업 입력 계층을 조회할 필요가 없을 경우 포트를 표시하도록 선택합니다. 포트를 표시할 경우 그룹을 정의하고, 포트를 정의하며, 작업 입력에서 출력 포트에 노드를 매핑할 수 있습니다. 작업 입력에서 출력 포트에 노드를 매핑하려면 **위치** 열의 필드를 클릭하고 **위치 선택** 대화 상자에서 계층을 확장합니다. 그런 다음, 계층에서 노드를 선택하십시오.

작업 입력 계층을 조회하려면 입력 매핑을 표시하도록 선택하십시오. 탭의 왼쪽은 **작업 입력** 영역이고, 탭의 오른쪽은 **포트** 영역입니다. **작업 입력** 영역에는 **SOAP** 요청 메시지 계층이 표시됩니다. **포트** 영역에서 출력 포트를 정의할 수 있습니다. 작업 입력에서 출력 포트에 노드를 매핑할 경우 노드 위치가 **작업 입력** 영역의 **위치** 열에 표시됩니다.

입력 매핑을 표시할 경우 출력 포트를 계층으로 표시하도록 선택할 수 있습니다. 입력 포트를 작업 입력의 노드에 연결하는 행을 표시하도록 선택할 수도 있습니다.

계층의 첫 번째 수준을 매핑하기로 선택한 경우 **Developer** 도구에서 작업 입력의 첫 번째 수준에 있는 노드를 출력 포트에 매핑합니다. **Developer** 도구에서 데이터를 매핑해야 하는 출력 포트도 작성합니다. 계층의 첫 번째 수준에 하나 이상의 다중 발생 하위 노드가 있는 다중 발생 상위 노드가 포함된 경우 **Developer** 도구는 포트를 작성하지 않거나 계층의 첫 번째 수준을 매핑하지 않습니다.

작업 입력을 포트에 매핑하기 위한 규칙 및 지침

입력 변환을 구성할 경우 작업 입력 계층의 노드를 출력 포트에 매핑하십시오.

작업 입력 계층의 노드를 출력 포트에 매핑할 경우 다음 규칙 및 지침을 고려하십시오.

- 노드 및 출력 포트에는 호환 가능한 데이터 유형이 있어야 합니다.
- 그룹의 둘 이상의 출력 포트에 노드를 매핑할 수 없습니다.

입력 변환 구성

웹 서비스 작업이 **SOAP** 요청 메시지를 수신하는 경우 해당 요청을 처리하는 입력 변환을 구성합니다. 변환에 출력 포트를 정의합니다. 작업 입력의 노드를 출력 포트에 매핑합니다.

1. 편집기에서 입력 변환을 선택합니다.
2. **속성** 보기의 **포트** 탭을 클릭합니다.
3. **입력 매핑**을 클릭합니다.

작업 입력 영역에 요청 메시지 계층이 표시됩니다. **포트** 영역에서 출력 포트를 정의합니다.

4. 필요에 따라 **행 표시**를 클릭하여 출력 포트를 작업 입력의 노드에 연결하는 행을 표시합니다.

모든 행을 표시하거나 선택된 포트의 행을 표시하도록 선택할 수 있습니다.

5. 필요에 따라 **계층으로 표시**를 클릭하여 출력 포트를 계층으로 표시합니다.

각 하위 그룹은 상위 그룹 아래에 나타납니다.

- 작업 입력에 anyType 요소, 임의 요소, anyAttribute 특성, 파생된 유형 요소 또는 대체 그룹이 포함된 경우 **작업 입력** 영역의 개체를 선택합니다. 노드의 **유형** 열에서 **선택**을 클릭한 다음, 목록에서 하나 이상의 유형, 요소 또는 특성을 선택합니다.
- 출력 그룹을 추가하려면 다음 방법 중 하나를 사용합니다.

옵션	설명
노드 끌기	작업 입력 영역의 그룹 노드 또는 하위 노드에서 포트 영역의 빈 열로 포인터를 끌어서 놓습니다. 노드가 그룹 노드일 경우 Developer tool에서 포트 없이 그룹을 추가합니다. 다른 출력 그룹이 있는 경우 새 그룹에 매핑 대화 상자에서 그룹을 다른 그룹에 연결합니다. Developer tool에서 관련 그룹에 대한 키를 작성합니다.
그룹을 수동으로 추가	새로 만들기 > 그룹 을 클릭하여 그룹을 추가합니다.
첫 번째 수준 계층 매핑 선택	첫 번째 수준 계층 매핑 을 선택합니다. Developer tool에서 작업 입력의 첫 번째 수준에 있는 노드를 출력 포트 및 그룹에 매핑합니다. 또한 Developer tool에서 데이터를 매핑하는 데 필요한 출력 포트 및 그룹을 작성합니다. 계층의 첫 번째 수준에 하나 이상의 다중 발생 하위 노드가 있는 다중 발생 상위 노드가 포함된 경우 Developer tool은 포트를 작성하지 않거나 계층의 첫 번째 수준을 매핑하지 않습니다.

- 출력 포트를 추가하고 노드를 출력 포트에 매핑하려면 다음 방법 중 하나를 사용합니다.

옵션	설명
노드 끌기	작업 입력의 노드에서 포트 영역의 그룹 이름 또는 포트로 포인터를 끌어서 놓습니다.
매핑 단추 클릭	작업 입력 영역에서 하나 이상의 노드를 선택합니다. 포트 영역에서 대상을 선택합니다. 매핑 을 클릭합니다.
포트 복사	다른 변환에서 포트를 선택하여 작업 입력 영역에 복사합니다. 포트를 복사하기 위해 키보드 바로 가기를 사용하거나 Developer tool의 복사 및 붙여넣기 단추를 사용할 수 있습니다.
포트를 수동으로 추가	새로 만들기 > 필드 를 클릭하여 포트를 추가합니다.
첫 번째 수준 계층 매핑 선택	첫 번째 수준 계층 매핑 을 선택합니다. Developer tool에서 작업 입력의 첫 번째 수준에 있는 노드를 출력 포트 및 그룹에 매핑합니다. 또한 Developer tool에서 매핑을 수행하는 데 필요한 출력 포트 및 그룹을 작성합니다. 계층의 첫 번째 수준에 하나 이상의 다중 발생 하위 노드가 있는 다중 발생 상위 노드가 포함된 경우 Developer tool은 포트를 작성하지 않거나 계층의 첫 번째 수준을 매핑하지 않습니다.

- 포트의 위치를 지우려면 다음 방법 중 하나를 사용합니다.

옵션	설명
지우기 단추 클릭	포트 영역에서 하나 이상의 포트를 선택하고 지우기 를 클릭합니다.
노드를 포트에 연결하는 행 삭제	작업 입력의 노드를 출력 포트에 연결하는 하나 이상의 행을 선택하고 삭제 를 누릅니다.

- 전체 SOAP 요청을 XML로 매핑하려면 **작업 입력** 영역의 **요청** 노드를 마우스 오른쪽 단추로 클릭하고 **XML로 매핑**을 선택합니다.

출력 변환

출력 변환은 웹 서비스 WSDL의 출력 요소 및 헤더 요소를 나타냅니다. 출력 변환은 작업 매핑의 관계형 데이터 그룹에서 SOAP 응답 메시지를 작성합니다. WSDL이 규모가 큰 SOAP 메시지 계층을 설명하는 경우에도 웹 서비스 작업은 SOAP 응답 메시지의 일부에 대한 데이터를 반환할 수 있습니다.

웹 서비스 작업 출력을 정의하면 Developer tool이 출력 변환을 작성합니다.

출력 변환의 **포트** 탭을 사용하여 작업 출력 계층을 보고 변환 입력 포트를 추가하고 입력 포트를 작업 출력에 매핑할 수 있습니다. 작업 출력 계층은 SOAP 응답 메시지 계층을 정의합니다.

한 문자열 또는 텍스트 입력 포트의 XML 데이터를 전체 SOAP 응답에 매핑할 수 있습니다. XML 데이터를 전체 SOAP 응답에 매핑할 경우 포트를 작업 출력의 노드에 매핑할 수 없습니다.

출력 변환의 **고급** 탭에서 고급 속성을 구성합니다.

출력 변환 포트 탭

포트 탭에서 입력 그룹 및 입력 포트를 정의하고 입력 포트를 작업 출력 노드에 매핑합니다.

포트를 표시할 때 수동으로 그룹 및 포트를 추가하거나 다른 변환의 포트를 출력 변환에 복사할 수 있습니다. 키보드 바로 가기를 사용하거나 Developer tool에서 복사 및 붙여넣기 단추를 사용할 수 있습니다.

출력 매핑을 표시할 경우 입력 그룹 및 입력 포트를 정의하고 입력 포트를 작업 출력 계층에 매핑할 수 있습니다. 탭의 왼쪽은 **포트** 영역이고, 탭의 오른쪽은 **작업 출력** 영역입니다. **작업 출력** 영역에는 SOAP 응답 메시지 계층이 표시됩니다. **포트** 영역에서 입력 그룹 및 입력 포트를 정의할 수 있습니다. **포트** 영역의 입력 포트를 **작업 출력** 영역의 노드에 매핑하면 작업 출력 영역의 **위치** 열에 입력 포트 위치가 나타납니다.

계층의 첫 번째 수준을 매핑하기로 선택한 경우 Developer tool에서 입력 포트를 작업 출력의 첫 번째 수준에 있는 노드에 매핑합니다. 또한 데이터를 매핑할 때 필요한 입력 포트도 작성합니다. 계층의 첫 번째 수준에 하나 이상의 다중 발생 하위 노드가 있는 다중 발생 상위 노드가 포함된 경우 Developer tool은 포트를 작성하지 않거나 계층의 첫 번째 수준을 매핑하지 않습니다.

출력 매핑을 표시할 때 입력 포트를 작업 입력의 노드에 연결하는 행을 표시하도록 선택할 수 있습니다.

출력 변환 고급 탭

고급 탭에서 출력 변환 고급 속성을 구성하십시오.

다음 테이블에는 출력 변환 **고급** 탭에서 구성할 수 있는 속성이 설명되어 있습니다.

속성	설명
정렬된 입력	데이터 통합 서비스에서 모든 입력 데이터를 처리하지 않고도 출력을 생성할 수 있습니다. 입력 데이터를 작업 입력 계층의 키별로 정렬한 경우 정렬된 입력을 활성화합니다. 기본값이 비활성화됩니다.

포트를 작업 출력에 매핑하는 규칙 및 지침

출력 변환을 구성할 때 포트를 작업 출력 계층에 매핑하십시오.

입력 포트를 작업 출력 계층에 매핑할 때 다음 규칙 및 지침을 고려하십시오.

- 입력 포트 및 노드는 호환되는 데이터 유형을 갖고 있어야 합니다.
- 입력 포트를 계층의 노드 하나에 매핑할 수 있습니다.
- 하나의 입력 그룹에서 작업 출력의 동일한 계층 수준에 있는 노드로 포트를 매핑할 수 있습니다.

- 하나의 입력 그룹에서 작업 출력의 다른 계층 수준에 있는 노드로 다른 포트를 매핑할 수 있습니다.
- 입력 포트를 작업 출력의 키에 매핑합니다. 키에 매핑하는 포트는 문자열, 정수 또는 **bigint** 데이터 유형이어야 합니다. SOAP 메시지에 포함하는 계층 수준 이상의 작업 출력에 있는 모든 수준의 키에 데이터를 매핑하십시오. 매핑하는 수준을 포함하여 그 이상의 모든 수준에 대한 외래 키를 포함하십시오.

참고: 작업 출력 계층의 가장 낮은 수준만 매핑할 경우 입력 포트를 키에 매핑할 필요가 없습니다.

- 다른 데이터 유형의 여러 입력 포트를 키에 매핑할 수 있습니다. 키의 **위치** 필드를 클릭할 때 입력 포트의 순서를 다시 정렬하거나 포트 중 하나를 제거할 수 있습니다.

출력 변환 구성

웹 서비스 작업에서 응답 메시지를 반환할 경우 출력 변환을 구성하십시오. 각 변환에 입력 포트를 정의하고 입력 포트의 데이터를 작업 출력 계층의 노드에 매핑하십시오.

1. 편집기에서 출력 변환을 선택합니다.

2. **속성** 보기의 **포트** 탭을 클릭합니다.

3. **출력 매핑**을 클릭합니다.

작업 출력 영역에 작업 출력 계층이 표시됩니다. **포트** 영역에서 입력 포트를 정의하십시오.

4. 필요에 따라 **행 표시**를 클릭하여 입력 포트를 작업 결합의 노드에 연결하는 행을 표시합니다. 모든 행을 표시하거나 선택된 포트의 행을 표시하도록 선택할 수 있습니다.

5. 작업 출력에 **anyType** 요소, 임의 요소, **anyAttribute** 특성, 파생된 유형 요소 또는 대체 그룹이 포함된 경우 **작업 출력** 영역의 개체를 선택합니다. 노드의 **유형** 열에서 **선택**을 클릭한 다음, 목록에서 하나 이상의 유형, 요소 또는 특성을 선택합니다.

6. 입력 그룹을 추가하려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
노드 끌기	작업 출력 영역의 그룹 노드 또는 하위 노드에서 포트 영역의 빈 열로 포인터를 끕니다. 노드가 그룹 노드일 경우 Developer 도구에서 포트 없이 그룹을 추가합니다.
그룹을 수동으로 추가	새로 만들기 단추 옆에 있는 화살표를 클릭한 다음 새 그룹 을 클릭합니다.
첫 번째 수준 계층 매핑 선택	첫 번째 수준 계층 매핑 을 선택합니다. Developer 도구에서 작업 입력의 첫 번째 수준에 있는 노드를 출력 포트 및 그룹에 매핑합니다. 또한 Developer 도구에서 데이터를 매핑하는 데 필요한 입력 포트 및 그룹을 작성합니다.

7. 입력 포트를 추가하려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
포트를 수동으로 추가	새로 만들기 단추 옆에 있는 화살표를 클릭한 다음 새 포트 를 클릭합니다.
다른 변환에서 포트 끌기	편집기에서 다른 변환의 포트를 출력 변환으로 끄십시오.
포트 복사	다른 변환에서 포트를 선택하여 작업 출력 영역에 복사합니다. 포트를 복사하기 위해 키보드 바로 가기를 사용하거나 Developer 도구의 복사 및 붙여넣기 단추를 사용할 수 있습니다.

옵션	설명
첫 번째 수준 계층 매핑 선택	첫 번째 수준 계층 매핑을 선택합니다. Developer 도구에서 작업 출력의 첫 번째 수준에 있는 노드를 입력 포트 및 그룹에 매핑합니다. 또한 Developer 도구에서 매핑을 수행하는 데 필요한 입력 포트 및 그룹을 작성합니다.

- 입력 포트의 데이터를 작업 출력 계층의 노드에 매핑하려면 각 입력 포트 또는 그룹의 포인터를 작업 출력의 연관된 노드로 끄십시오. **작업 출력** 영역의 노드 옆에 입력 필드 위치가 표시됩니다.
- 입력 포트의 XML 데이터를 전체 SOAP 응답에 매핑하려면 포트를 마우스 오른쪽 단추로 클릭하고 **XML로 매핑**을 선택합니다.
- 입력 포트를 복합 키로 매핑하려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
입력 포트 끌기	두 개 이상의 입력 포트를 선택하고 작업 출력 계층의 키로 끄십시오.
위치 선택 대화 상자에서 입력 포트 선택	작업 출력 계층에서 키의 위치 열을 클릭한 다음 입력 포트를 선택합니다.

- 노드의 위치를 지우려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
지우기 단추 클릭	작업 출력 영역에서 하나 이상의 노드를 선택하고 지우기 를 클릭합니다.
포트를 노드에 연결하는 행 삭제	작업 출력에서 입력 포트를 노드에 연결하는 하나 이상의 행을 선택하고 삭제 를 누릅니다.

결함 변환

결함 변환은 웹 서비스 WSDL의 결함 요소를 나타냅니다. 결함 변환은 웹 서비스 작업에서 사용자 정의 결함을 생성합니다.

결함 변환은 다음 사용자 정의 결함 중 하나를 나타냅니다.

미리 정의된 결함

미리 정의된 결함은 결함 변환이 웹 서비스 WSDL의 결함 요소를 나타내는 결함입니다. 미리 정의된 결함에 서 결함 변환은 웹 서비스 작업 매핑의 관계형 데이터로부터 오류 메시지를 작성합니다.

일반 결함

일반 결함은 결함 변환이 웹 서비스 작업에 대해 웹 서비스 WSDL에 정의된 결함 요소를 나타내지 않는 결함입니다. 일반 결함의 결함 변환은 변환에서 오류가 발생할 경우 일반 오류 메시지를 반환합니다. 일반 결함의 결함 변환은 Developer tool에서 작성하고 구성할 수 있습니다.

작업 매핑에는 여러 결함 변환이 포함될 수 있습니다. 동일한 결함 변환의 여러 인스턴스를 매핑에 추가하여 동일한 메시지를 매핑의 서로 다른 부분에 생성할 수 있습니다.

작업 서명을 변경하지 않고 결함 변환을 매핑에 추가하거나 매핑에서 제거할 수 있습니다. 매핑이 실행될 때 결함 오류가 발생할 수 없다는 것을 아는 경우 매핑에서 결함 변환을 제거할 수 있습니다. 결함 변환은 업스트림 변환에 연결해야 합니다. 그렇지 않으면 매핑이 올바르게 작동하지 않습니다.

SOAP 1.1 바인딩이 있는 작업에서 결함을 작성할 경우 마법사에서 **faultcode**, **faultstring** 및 **faultactor** 요소가 작성됩니다. SOAP 1.2 바인딩이 있는 작업에서 결함을 작성할 경우 마법사에서 **code**, **reason**, **node** 및 **role** 요소가 작성됩니다. 결함에 요소를 추가하면 마법사에서 요소가 결함의 세부 정보 그룹에 추가됩니다. 결함 변환의 **포트** 탭을 사용하여 작업 결함 계층을 보고 변환 입력 포트를 추가하고 입력 포트를 작업 결함에 매핑할 수 있습니다. 작업 결함 계층은 사용자 정의 결함으로부터 생성되는 오류 메시지에 대한 SOAP 응답 메시지 계층을 정의합니다.

한 문자열 또는 텍스트 입력 포트의 XML 데이터를 전체 SOAP 응답에 매핑할 수 있습니다. XML 데이터를 전체 SOAP 응답으로 매핑하는 경우 포트를 작업 결함의 노드로 매핑할 수 없습니다.

결함 변환의 **고급** 탭에서 고급 속성을 구성할 수 있습니다.

결함 변환 포트 탭

포트 탭에서 입력 그룹 및 입력 포트를 정의하고 입력 포트를 작업 결함 노드에 매핑하십시오.

포트를 표시할 때 수동으로 그룹 및 포트를 정의할 수 있습니다. 또는 다른 변환에서 결함 변환으로 포트를 복사할 수 있습니다. 키보드 바로 가기를 사용하거나 **Developer** 도구에서 복사 및 붙여넣기 단추를 사용할 수 있습니다.

결함 매핑을 표시할 경우 입력 그룹 및 입력 포트를 정의하고 입력 포트를 작업 결함 계층에 매핑할 수 있습니다. 탭의 왼쪽은 **포트** 영역이고, 탭의 오른쪽은 **작업 결함** 영역입니다. **작업 결함** 영역에는 SOAP 응답 메시지 계층이 표시됩니다. 포트 영역의 입력 포트를 **작업 결함** 영역의 노드에 매핑할 경우 입력 포트 위치가 **작업 결함** 영역의 **위치** 열에 표시됩니다.

계층의 첫 번째 수준을 매핑하기로 선택한 경우 **Developer** 도구에서 입력 포트를 작업 결함의 첫 번째 수준에 있는 노드를 매핑합니다. **Developer** 도구에서 데이터를 매핑해야 하는 포트도 작성합니다.

결함 변환 고급 탭

고급 탭에서 결함 변환 고급 속성을 구성하십시오.

다음 테이블에는 **고급** 탭에서 구성할 수 있는 속성이 설명되어 있습니다.

속성	설명
정렬된 입력	데이터 통합 서비스에서 모든 입력 데이터를 처리할 필요 없이 출력을 생성할 수 있습니다. 입력 데이터를 작업 결함 계층의 키별로 정렬한 경우 정렬된 입력을 활성화합니다. 기본값이 비활성화됩니다.

포트를 작업 결함에 매핑하는 규칙 및 지침

결함 변환을 구성할 때 입력 포트를 작업 결함 계층에 매핑하십시오.

입력 포트를 작업 결함 계층에 매핑할 때 다음 규칙 및 지침을 고려하십시오.

- 입력 포트를 작업 결함 계층의 노드 하나에 매핑할 수 있습니다. 입력 포트 및 노드는 호환되는 데이터 유형을 갖고 있어야 합니다.
- 하나의 입력 그룹에서 작업 결함의 동일한 계층 수준에 있는 노드로 포트를 매핑할 수 있습니다.
- 하나의 입력 그룹에서 작업 결함의 다른 계층 수준에 있는 노드로 다른 포트를 매핑할 수 있습니다.
- 입력 포트 데이터를 작업 결함 계층의 키에 매핑해야 합니다. 키에 매핑하는 포트는 문자열, 정수 또는 **bigint** 데이터 유형이어야 합니다. SOAP 메시지에 포함하는 계층 수준 이상의 작업 결함에 있는 모든 수준의 키에 데이터를 매핑하십시오. 매핑하는 수준을 포함하여 그 이상의 모든 수준에 대한 외래 키를 포함하십시오.
- 다른 데이터 유형의 여러 입력 포트를 키에 매핑할 수 있습니다. 키의 **위치** 필드를 클릭할 때 입력 포트의 순서를 다시 정렬하거나 포트 중 하나를 제거할 수 있습니다.

결함 변환 작성

일반 결함 또는 미리 정의된 결함 변환을 작성할 수 있습니다. 일반 결함의 경우 웹 서비스 WSDL에서 결함 요소를 정의하지 않습니다. 사전 정의된 결함의 경우 웹 서비스에서 결함 요소를 사용하여 결함을 정의합니다.

1. 웹 서비스를 엽니다.
2. **아웃라인** 보기에서 작업 매핑을 선택합니다.
편집기에 작업 매핑이 표시됩니다.
3. 편집기에서 마우스 오른쪽 단추를 클릭하고 **변환 추가**를 선택합니다.
변환 추가 대화 상자가 나타납니다.
4. **결함**을 선택하고 **확인**을 클릭합니다.
결함 추가 대화 상자가 나타납니다.
5. 결함 변환을 작성하려면 다음 단계 중 하나를 수행하십시오.
 - **일반 결함으로 작성**을 선택합니다.
 - **결함 요소로부터 미리 정의된 결함으로 작성**을 선택합니다.
참고: 웹 서비스에 결함을 정의하는 요소가 포함되어 있습니다.
6. **확인**을 클릭합니다.
결함 변환이 일반 결함 또는 미리 정의된 결함으로 표시됩니다.

결함 변환 구성

웹 서비스 작업에서 결함을 반환하거나 일반 결함에 대한 결함 변환을 작성한 경우 각 결함 변환을 구성하십시오. 입력 포트를 정의하고 입력 포트의 데이터를 작업 결함의 노드에 매핑하십시오.

1. 편집기에서 결함 변환을 선택합니다.
2. **속성** 보기의 **포트** 탭을 클릭합니다.
3. **결함 매핑**을 클릭합니다.
작업 결함 영역에는 응답 또는 결함 메시지 계층이 표시됩니다. **포트** 영역에서 입력 포트를 정의하십시오.
4. 필요에 따라 **행 표시**를 클릭하여 입력 포트를 작업 결함의 노드에 연결하는 행을 표시합니다. 모든 행을 표시하거나 선택된 포트의 행을 표시하도록 선택할 수 있습니다.
5. 작업 결함에 anyType 요소, 임의 요소, anyAttribute 특성, 파생된 유형 요소 또는 대체 그룹이 포함된 경우 **작업 결함** 영역의 개체를 선택합니다. 노드의 **유형** 열에서 **선택**을 클릭한 다음, 목록에서 하나 이상의 유형, 요소 또는 특성을 선택합니다.
6. 입력 그룹을 추가하려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
노드 끌기	작업 결함 영역의 그룹 노드 또는 하위 노드에서 포트 영역의 빈 열로 포인터를 끕니다. 노드가 그룹 노드일 경우 Developer 도구에서 포트 없이 그룹을 추가합니다.
그룹을 수동으로 추가	새로 만들기 단추 옆에 있는 화살표를 클릭한 다음 새 그룹 을 클릭합니다.
첫 번째 수준 계층 매핑 선택	첫 번째 수준 계층 매핑 을 선택합니다. Developer 도구에서 작업 결함의 첫 번째 수준에 있는 노드를 입력 포트 및 그룹에 매핑합니다. 또한 Developer 도구에서 데이터를 매핑하는 데 필요한 입력 포트 및 그룹을 작성합니다.

7. 입력 포트를 추가하려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
포트를 수동으로 추가	새로 만들기 단추 옆에 있는 화살표를 클릭하고 새 포트 를 클릭합니다.
다른 변환의 포트를 끕니다.	편집기에서 다른 변환의 포트를 결합 변환으로 끄십시오.
포트 복사	다른 변환에서 포트를 선택하여 작업 결합 영역에 복사합니다. 포트를 복사하기 위해 키보드 바로 가기를 사용하거나 Developer 도구의 복사 및 붙여넣기 단추를 사용할 수 있습니다.
첫 번째 수준 계층 매핑 선택	첫 번째 수준 계층 매핑 을 선택합니다. Developer 도구에서 작업 결합의 첫 번째 수준에 있는 노드를 입력 포트 및 그룹에 매핑합니다. 또한 Developer 도구에서 데이터를 매핑하는데 필요한 입력 포트 및 그룹을 작성합니다.

8. 입력 포트의 데이터를 작업 결합 계층의 노드에 매핑하려면 각 입력 포트 또는 그룹의 포인터를 작업 결합의 연관된 노드로 끄십시오.

작업 결합 영역의 노드 옆에 입력 필드 위치가 표시됩니다.

9. 입력 포트의 XML 데이터를 전체 SOAP 응답에 매핑하려면 포트를 마우스 오른쪽 단추로 클릭하고 **XML로 매핑**을 선택합니다.
10. 입력 포트를 복합 키로 매핑하려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
입력 포트 끌기	두 개 이상의 입력 포트를 선택하고 작업 결합 계층의 키로 끄십시오.
위치 선택 대화 상자에서 입력 포트 선택	작업 결합 계층에서 키의 위치 열을 클릭한 다음 입력 포트를 선택합니다.

11. 노드의 위치를 지우려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
지우기 단추 클릭	작업 결합 영역에서 하나 이상의 노드를 선택하고 지우기 를 클릭합니다.
포트를 노드에 연결하는 행 삭제	작업 결합에서 입력 포트를 노드에 연결하는 하나 이상의 행을 선택하고 삭제 를 누릅니다.

결함 처리

웹 서비스에서 오류가 발생할 경우 데이터 통합 서비스가 오류 메시지를 생성하고 결함의 메시지를 웹 서비스 클라이언트에 반환합니다.

SOAP 1.1 요청이 SOAP 1.2 바인딩이 있는 작업에 전송될 경우 웹 서비스에서 SOAP 1.1을 사용하여 결함을 생성합니다. SOAP 1.2 요청이 SOAP 1.1 바인딩이 있는 작업에 전송될 경우 웹 서비스에서 SOAP 1.2를 사용하여 결함을 생성합니다. 웹 서비스에서는 시스템 정의 결함 및 사용자 정의 결함이 생성될 수 있습니다.

사용자 정의 오류가 발생하면 데이터 통합 서비스에서 웹 서비스 클라이언트에 오류 메시지를 반환합니다. 사용자 정의 결함은 다음 유형 중 하나일 수 있습니다.

- 미리 정의됨
- 일반

작업 매핑에 출력 변환이 포함될 경우 웹 서비스에서 출력 변환의 데이터를 반환하거나 결함을 반환합니다. 작업 매핑이 데이터를 대상 변환 또는 외부 응용 프로그램에 커밋한 후에 결함이 발생할 경우 데이터 통합 서비스에서 데이터를 롤백할 수 없습니다. 따라서 매핑이 중지되고 웹 서비스 출력 변환이 수신한 데이터가 무시됩니다.

SOAP 1.1 결함

SOAP 1.1의 경우 결함은 다음 구조의 SOAP 메시지입니다.

```
Fault (FaultName)
Key_Fault (FaultName)
faultcode          xs:QName
faultstring        xs:string
faultactor        xs:anyURI
detail
```

결함에는 다음 요소가 포함됩니다.

결함 코드

오류 메시지 번호와 같은 결함 ID 코드입니다.

결함 문자열

오류에 대한 설명입니다.

결함 작업자

결함을 야기한 개체에 대한 선택적 정보입니다.

세부 정보

결함에 따라 달라지는 선택적 정보입니다.

SOAP 1.2 결함

SOAP 1.2의 경우 결함은 다음 구조의 SOAP 메시지입니다.

```
Fault (FaultName)
Key_Fault (FaultName)
Code          tns:faultcodeEnum
Reason        tns:reasonText
Node          xs:anyURI
Role          xs:anyURI
detail
```

결함에는 다음 요소가 포함됩니다.

코드

결함 ID입니다.

코드의 값 요소는 다음 값 중 하나여야 합니다.

- infasapns:DataEncodingUnknown
- infasapns:MustUnderstand
- infasapns:Receiver

- infasoapns:Sender
- infasoapns:VersionMismatch

참고: 코드 결함 요소를 확장하여 한 수준 위의 **SubCode** 결함 요소를 추출할 수 있습니다. **SubCode**의 스 키마 유형은 **xsd:QName**입니다. **SubCode** 결함 요소를 사용하여 오류 메시지 번호를 정의할 수 있습니다.

이유

오류에 대한 설명입니다.

노드

결함을 생성한 **SOAP** 노드의 **URI**를 포함합니다.

역할

결함을 야기한 개체에 대한 선택적 정보입니다.

세부 정보

결함에 따라 달라지는 선택적 정보입니다.

시스템 정의된 결함

데이터 통합 서비스는 시스템 오류가 발생할 경우 시스템 정의된 결함을 생성합니다. 시스템 오류가 발생할 경우 데이터 통합 서비스가 시스템 정의된 결함에서 오류 메시지를 반환합니다.

예를 들어, 숫자 입력 포트에서 숫자가 아닌 데이터를 받을 경우 데이터 통합 서비스가 다음 결함을 반환할 수 있습니다.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <soapenv:Fault>
      <faultcode>WSCMN_10022</faultcode>
      <faultstring>[WSCMN_10022] Mapping execution failed:
[com.informatica.platform.ldtm.common.ExecutionException: [MPSVCCMN_10009] The Mapping Service Module
[MappingService] encountered an exception with the following details: [LDTM_0072] [ERROR] XML parsing
component [Input_S2R] message code: [66022], message body: A data conversion error occurred in field
[customerID] of the transformation group [Operation]. The field contains the following data:
[1100AA].].</faultstring>
      <detail>
        </detail>
      </soapenv:Fault>
    </soapenv:Body>
  </soapenv:Envelope>
```

참고: 예제에서 **SOAP 1.1**을 사용합니다.

결함 변환 없이 오류 메시지를 웹 서비스 클라이언트에 반환할 수 있습니다. 식 변환에서 **ABORT(msg)** 함수를 호출하여 결함을 발생시킬 수 있습니다. **ABORT** 메시지를 호출할 경우 데이터 통합 서비스에서 시스템 정의된 결함 메시지를 생성합니다. **ABORT** 함수에 전달하는 메시지가 시스템 정의된 결함의 결함 문자열이 됩니다.

사전 정의된 결함

사전 정의된 결함의 경우 웹 서비스에서 결함 요소를 사용하여 결함을 정의합니다. 사용자 지정 오류 메시지를 반환하려면 결함 변환을 구성하십시오.

웹 서비스에서 결함 변환을 구성한 경우 오류 조건을 반환하는 작업 매핑 논리를 정의해야 합니다. 결함 변환에 전달할 오류 메시지를 생성하려면 변환을 정의해야 합니다.

결함 변환을 정의한 경우 결함 코드, 결함 문자열 및 결함 작업자에서 반환할 데이터를 정의하십시오. 웹 서비스 클라이언트에 추가 정보를 반환하기 위해 결함 메시지에 다른 요소를 추가할 수도 있습니다. 결함에 요소를 정의

한 경우 마법사가 세부 정보 그룹에 요소를 추가합니다. 다중 발생 요소를 정의하고 세부 정보 그룹의 요소 그룹 간에 계층 관계를 작성할 수 있습니다.

오류마다 오류 메시지 구조가 동일한 경우 매핑에서 동일한 결함 변환을 여러 번 사용할 수 있습니다. 그렇지 않으면 웹 서비스 클라이언트에 반환할 오류 메시지마다 다른 결함 변환을 구성할 수 있습니다.

예를 들어, 조회에서 직원을 찾지 못한 경우 웹 서비스 클라이언트에 메시지를 반환하도록 결함 변환을 구성하십시오. **faultcode** 요소에 오류 번호를 전달하고 **faultstring** 요소에 오류 메시지를 전달하십시오. 또한 부서 ID 및 직원 ID도 웹 서비스 클라이언트에 반환해야 합니다. 결함을 정의한 경우 세부 정보 그룹에 **DeptID** 및 **EmployeeID** 요소를 추가합니다.

데이터 통합 서비스에서 다음 결함을 반환할 수 있습니다.

```
<infasoapns:Envelope xmlns:infasoapns="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:infawSDLns="http://schemas.xmlsoap.org/wsdl/" xmlns:tns="http://www.informatica.com/dis/ws/
  Get_Employee_Info_Web_Service">
  <infasoapns:Body>
    <infasoapns:Fault>
      <faultcode>ERR_12345</faultcode>
      <faultstring>Web service failed to retrieve employee information.</faultstring>
      <detail>
        <tns:Employee_Not_Found>
          <tns:DeptID>100</tns:DeptID>
          <tns:EmployeeID>2428</tns:EmployeeID>
        </tns:Employee_Not_Found>
      </detail>
    </infasoapns:Fault>
  </infasoapns:Body>
</infasoapns:Envelope>
```

참고: 예제에서 SOAP 1.1을 사용합니다.

일반 결함

WSDL의 결함 요소에서 오류를 정의하지 않았을 때 오류 메시지를 웹 서비스 클라이언트에 반환하도록 일반 결함을 정의할 수 있습니다. 변환에서 오류가 발생할 때 일반 오류 메시지를 반환하려면 결함 변환을 작성합니다.

웹 서비스에서 일반 결함에 대한 결함 변환을 작성할 때 오류 조건을 반환하는 작업 매핑 논리를 정의해야 합니다.

일반 결함에 대한 결함 변환을 정의할 때 작업 바인딩이 **SOAP 1.1** 형식이면 결함 코드, 결함 문자열 및 결함 작업자 요소로 반환할 데이터를 정의합니다. 작업 바인딩이 **SOAP 1.2** 형식일 경우 코드, 이유, 노드 및 역할 요소로 반환할 데이터를 정의합니다. 또한 필요에 따라 세부 정보 문자열로 반환할 데이터를 정의할 수도 있습니다. 세부 정보 문자열은 선택 사항이며 **xsd:any** 요소가 포함됩니다. 세부 정보 문자열로 데이터를 전송하려는 경우 세부 정보 요소를 웹 서비스 작업에 정의된 포트 중 하나로 매핑합니다. 웹 서비스 클라이언트에 추가 정보를 반환하기 위해 결함 메시지에 다른 요소를 추가할 수도 있습니다.

세부 정보 그룹에 있는 요소 그룹 간의 계층 관계를 작성할 수 있습니다. 오류마다 오류 메시지 구조가 동일한 경우 매핑에서 동일한 결함 변환을 여러 번 사용할 수 있습니다. 웹 서비스 클라이언트에 반환할 각 오류 메시지마다 다른 결함 변환을 작성할 수도 있습니다.

참고: 결함 변환을 작성한 후에는 일반 결함 또는 미리 정의된 결함의 상태를 변경할 수 없습니다.

작업 매핑 테스트

작업 매핑의 출력을 미리 보고 원하는 결과를 생성하는지 확인하십시오. 매핑에서 변환 결과를 미리 볼 수도 있습니다.

작업 매핑의 출력 또는 작업 매핑의 변환 출력을 미리 볼 때 **Developer** 도구가 적용하는 설정을 제어하려면 웹 서비스 구성을 작성하십시오. **기본 설정** 대화 상자를 사용하여 기본 웹 서비스 실행 구성을 구성하십시오. 또한 **실행** 대화 상자를 사용하여 **데이터 뷰어** 보기에서 지정할 수 있는 웹 서비스 구성을 작성할 수도 있습니다.

작업 매핑 테스트

작업에 대한 **SOAP** 응답을 미리 보려면 작업 매핑을 테스트하십시오. 작업 매핑의 출력을 미리 보거나 매핑의 변환에서 출력을 미리 볼 수 있습니다.

도메인에 둘 이상의 데이터 통합 서비스가 포함되는 경우 데이터를 미리 보려면 기본 데이터 통합 서비스를 선택해야 합니다.

1. 웹 서비스를 엽니다.
2. **아웃라인** 보기에서 작업 매핑을 선택합니다.
편집기에 작업 매핑이 표시됩니다.
3. **데이터 뷰어** 보기를 선택합니다.
4. **입력** 창에서 요청을 입력합니다.
5. 특정 변환의 출력 데이터를 볼 경우 편집기에서 변환을 선택합니다. 그렇지 않으면 데이터 뷰어를 실행할 때 작업 매핑의 출력이 표시됩니다.
6. **실행**을 클릭합니다.
요청 결과가 **출력** 창에 표시됩니다.
7. 다른 요청을 사용하여 작업 매핑을 테스트하려면 **재설정**을 클릭한 다음 **4**에서 **6**까지의 단계를 반복합니다.

사용자 지정된 보기 옵션

입력, 출력 및 결합 변환을 구성할 경우 **작업** 영역에 키를 표시하도록 **SOAP** 메시지 계층을 변경할 수 있습니다. 노드 정렬 방법을 정의하는 그룹화 구성을 표시할 수도 있습니다.

사용자 지정된 보기 옵션을 표시하려면 **작업 입력** 영역, **작업 출력** 영역 또는 **작업 결합** 영역에서 **보기 사용자 지정**을 클릭합니다.

다음 옵션을 활성화할 수 있습니다.

모두, 시퀀스 및 선택

요소 정의가 모두, 시퀀스 또는 선택인지 보여주는 행을 표시합니다.

모든 그룹의 노드가 **SOAP** 메시지에 모두 포함되어야 합니다.

시퀀스 그룹의 노드는 **WSDL**에 지정된 순서여야 합니다.

선택 그룹의 노드 하나 이상이 **SOAP** 메시지에 표시되어야 합니다.

키

각 계층 수준에 대한 키를 표시합니다.

제 7 장

웹 서비스 SOAP 메시지 구문 분석

이 장에 포함된 항목:

- [웹 서비스 SOAP 메시지 구문 분석 개요, 62](#)
- [변환 사용자 인터페이스, 62](#)
- [다중 발생 출력 구성, 63](#)
- [anyType 요소 구문 분석, 66](#)
- [파생된 유형 구문 분석, 67](#)
- [QName 요소 구문 분석, 68](#)
- [대체 그룹 구문 분석, 68](#)
- [SOAP 메시지의 XML 구성 구문 분석, 68](#)

웹 서비스 SOAP 메시지 구문 분석 개요

데이터 통합 서비스는 웹 서비스 변환에서 SOAP 메시지를 구문 분석할 때 행 데이터를 생성합니다.

웹 서비스 입력 변환 및 웹 서비스 소비자 변환은 SOAP 메시지를 구문 분석하는 웹 서비스 변환입니다.

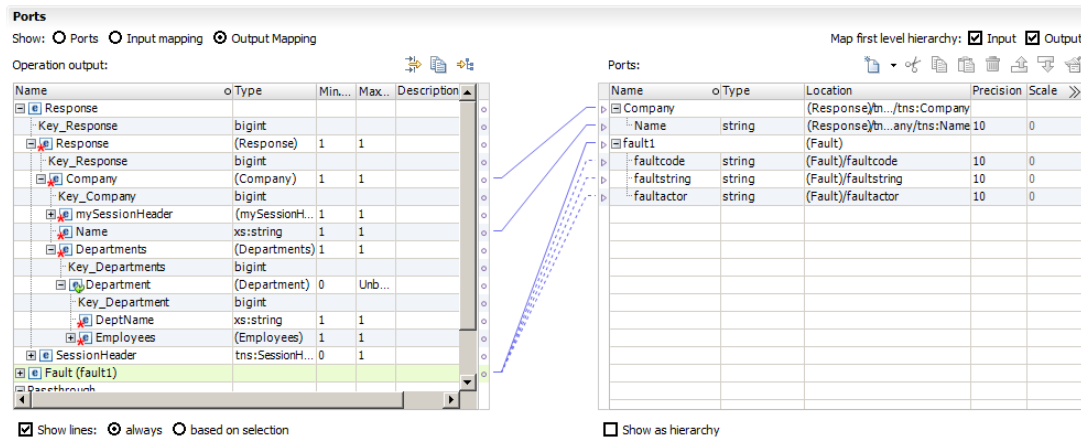
SOAP 메시지를 구문 분석하는 변환을 구성하려면 SOAP 메시지 계층과 유사한 구조의 출력 포트를 작성합니다. SOAP 메시지 계층의 노드를 포트에 매핑합니다.

출력 포트의 정규화된 그룹, 비정규화된 그룹 및 포트의 피벗된 그룹을 구성할 수 있습니다. SOAP 메시지에 파생된 유형, anyType 요소 또는 대체 그룹이 포함되는 경우 SOAP 메시지 인스턴스에서 발생할 수 있는 유형에 따라 서로 다른 출력 그룹을 구성할 수 있습니다.

변환 사용자 인터페이스

웹 서비스 소비자 변환 및 웹 서비스 입력 변환에서 SOAP 메시지의 데이터를 변환 출력 포트에 매핑하는 데 사용할 수 있는 사용자 인터페이스를 제공합니다.

다음 그림에서는 웹 서비스 소비자 변환에서 SOAP 1.1 메시지 노드 및 출력 포트 간 매핑을 보여 줍니다.



작업 영역

작업 영역에는 SOAP 메시지 계층이 포함되어 있습니다. 복합 노드 또는 다중 발생 노드는 구조의 계층 수준을 정의합니다. Developer tool에서 수준 간 상위-하위 관계를 정의하는 키를 수준에 추가합니다.

이전 그림에서 SOAP 메시지 계층에는 다음 수준이 있습니다.

응답 또는 요청

응답 또는 요청 메시지의 루트를 나타내는 수준입니다.

회사

요청 데이터의 최상위 수준입니다.

부서

회사 내 다중 발생 부서입니다.

직원

직원은 부서 내 복합 요소입니다.

결함 그룹

오류 메시지를 수신하는 결함 메시지 그룹입니다.

포트 영역

각 SOAP 메시지 수준의 데이터를 출력 포트에 매핑할 수 있습니다. 출력 포트의 각 그룹이 기본 외래 키 관계를 가진 다른 출력 그룹에 연결될 수 있습니다.

이전 그림에서 변환에는 SOAP 메시지의 노드 그룹에 해당하는 출력 포트 그룹이 있습니다.

다중 발생 출력 구성

입력 변환 또는 웹 서비스 소비자 변환에서 다중 발생 데이터를 반환할 경우 출력 포트를 서로 다른 구성으로 구성할 수 있습니다.

정규화된 출력 데이터, 피벗된 출력 데이터 또는 비정규화된 출력 데이터를 구성할 수 있습니다.

예를 들어 SOAP 메시지에 Department 및 Employee 복합 요소가 포함되어 있습니다. 각 Department에는 다수의 Employee가 포함됩니다. Department는 Employee의 상위 요소입니다.

SOAP 메시지에는 다음 요소 계층이 포함됩니다.

```
Departments
  Department_ID
  Department_Name
Employees
  Employee_ID
  Employee_Name
```

정규화된 관계형 출력

정규화된 출력 데이터를 생성할 경우 데이터 값이 출력 그룹에서 반복되지 않습니다. SOAP 메시지의 계층 수준과 포트의 출력 그룹 사이에 일대일 관계를 작성합니다.

SOAP 메시지에 **Department** 상위 계층 수준과 **Employee** 하위 계층 수준이 포함될 경우 다음 포트 그룹을 작성할 수 있습니다.

```
Departments
  Department_Key
  Department_ID
  Department_Name

Employees
  Department_Key
  Employee_ID
  Employee_Name
```

Department_Key는 Employee 출력 그룹을 Department 그룹에 연결하는 생성된 키입니다.

생성된 키

출력 그룹을 추가하면 Developer tool에서 생성된 키가 있는 다른 출력 그룹에 해당 출력 그룹을 연결합니다. Developer tool에서는 상위 그룹과 하위 그룹에 **bigint** 키를 추가합니다. 런타임 시 데이터 통합 서비스에서는 생성된 키에 대한 키 값을 작성합니다.

예

SOAP 계층에는 다음 노드가 포함됩니다.

```
Departments
  Dept_Key
  Dept_Num
  Dept_Name

Employees
  Dept_FK
  Employee_Num
  Employee_Name
```

Departments에 대한 포트의 출력 그룹을 작성하는 경우 Departments 노드를 포트 영역의 빈 필드에 매핑해야 합니다. Developer tool은 다음 출력 그룹을 작성합니다.

```
Departments
  Dept_Num
  Dept_Name
```

Employees 노드를 Ports 영역의 빈 필드에 매핑하면 Developer tool에서 Employees 그룹을 Departments 그룹에 연결하라는 메시지를 표시합니다. Employees 그룹을 둘 이상의 그룹에 연결할 수 있습니다. Developer tool이 각 그룹에 키를 추가합니다.

Developer tool은 다음 그룹 및 생성된 키를 작성합니다.

```
Departments
  Key_Departments
```



```

Dept_Num
Dept_Name

Employees
  Key_Departments
  Employee_Num
  Employee_Name

```

참고: 노드를 생성된 키에 매핑할 필요는 없습니다. 데이터 통합 서비스에서 런타임 시 키 값을 작성합니다.

Developer tool은 하나의 출력 그룹에 여러 수준으로 생성된 키를 작성할 수 있습니다. Employees 그룹에는 다음 포트가 포함될 수 있습니다.

```

Employees
  Key_Employees
  Key_Departments
  Key_Managers
  Employee_Num
  Employee_Name

```

Key_Departments 및 Key_Managers는 상위 그룹을 가리키는 생성된 키입니다. Key_Employees는 Employees 그룹에 대한 생성된 키입니다. Key_Employees는 하위 그룹을 Employees 그룹에 연결할 때 나타냅니다.

비정규화된 관계형 출력

관계형 출력을 비정규화할 수 있습니다. 출력 데이터를 비정규화하는 경우 상위 그룹의 요소 값이 각 하위 요소에 대해 반복됩니다.

출력 데이터를 비정규화하려면 상위 계층 수준의 노드를 출력 포트의 하위 그룹에 매핑합니다.

다음 예제는 Employees 출력 그룹의 Department_ID 및 Department_Name을 보여 줍니다.

```

Employees
  Department_ID
  Department_Name
  Employee_ID
  Employee_Name

```

Department_ID 및 Department_Name은 부서의 각 직원에 대해 반복됩니다.

Department_ID	Department_Name	Employee_ID	Employee_Name
100	Accounting	56500	Kathy Jones
100	Accounting	56501	Tom Lyons
100	Accounting	56509	Bob Smith

피벗된 관계형 출력

여러 번 발생하는 요소를 특정 개수만큼 출력 그룹에 포함할 수 있습니다.

여러 번 발생하는 요소를 피벗하려면 여러 번 발생하는 하위 요소를 출력 포트의 상위 그룹에 매핑합니다.

Developer tool에서 상위에 포함할 하위 요소 수를 정의하라는 메시지가 표시됩니다.

다음 예제는 Departments 상위 그룹의 두 Employee_ID 인스턴스를 보여 줍니다.

```

Departments
  Department_ID
  Department_Name
  Employee_ID1
  Employee_ID2

```

anyType 요소 구문 분석

anyType 요소는 WSDL 또는 스키마에서 모든 글로벌 유형의 선택을 나타냅니다. **Developer tool**에서 노드를 포트에 매핑할 때 **anyType** 요소에 대한 SOAP 메시지에 표시할 유형을 선택해야 합니다. SOAP 메시지의 **anyType** 요소를 복합 유형 또는 **xs:string**으로 바꿔야 합니다. 선택한 각 유형에 대한 포트 그룹을 작성합니다.

데이터를 출력 포트에 매핑하려면 유형을 선택해야 합니다. WSDL 또는 스키마에 글로벌 유형이 포함되지 않는 경우 **Developer tool**이 **anyType** 요소를 **xs:string**으로 바꿉니다.

작업 영역에서 요소 유형을 선택하려면 **anyType** 요소의 **유형** 열에서 **선택**을 클릭합니다. 사용 가능한 복합 유형 및 **xs:string**의 목록이 나타납니다.

anyType 요소를 파생된 유형으로 바꿀 경우 데이터 통합 서비스에서는 한 번에 한 유형에 대한 요소를 채웁니다. SOAP 메시지에는 기본 유형 및 파생된 유형의 데이터가 동시에 포함되지 않습니다.

파생된 유형 예제

WSDL에 **anyType** 요소가 포함되어 있습니다. 이 요소를 **AddressType**과 파생된 유형 **USAddressType**으로 바꿉니다. SOAP 메시지 계층에는 다음 그룹이 포함됩니다.

```
Address:AddressType (base type)
  Address: AddressType
    Street
    City

Address:USAddressType (derived type)
  Street
  City
  State
  ZipCode
```

SOAP 메시지에는 다음 데이터가 포함됩니다.

```
<address xsi: type ="AddressType">
<street>1002 Mission St.</street>
<city>san jose</city>
</address>

<address xsi:type="USAddressType">
<street>234 Fremont Blvd</street>
<city>Fremont</city>
<zip>94556</zip>
<state>CA</state>
</address>
```

데이터 통합 서비스는 **xsi: AddressType**에 대해 하나의 행을 반환합니다.

Street	City
1002 Mission St.	San Jose

데이터 통합 서비스는 파생된 유형 **xsi: USAddressType**에 대해 하나의 행을 반환합니다.

Street	City	State	Zip
234 Fremont Blvd.	Sunnyvale	CA	94556

유형이 **xsi: USAddressType**일 경우 데이터 통합 서비스가 **AddressType**을 채우지 않습니다.

파생된 유형 구문 분석

파생된 유형이 포함된 SOAP 메시지를 구문 분석할 수 있습니다. SOAP 메시지의 데이터를 수신하는 포트를 정의할 때는 SOAP 메시지에 표시될 수 있는 유형을 선택합니다. 선택한 유형의 요소에 따라 작성해야 하는 포트가 결정됩니다.

예를 들어 WSDL에는 AddressType과 이름이 USAddressType인 파생된 유형이 포함될 수 있습니다. Developer tool의 작업 영역에서 다음 그룹을 작성할 수 있습니다.

```
Address
  Address: AddressType
    Street
    City

  Address:USAddressType
    Street
    City
    State
    ZipCode
```

SOAP 메시지에는 다음 데이터가 포함될 수 있습니다.

```
<address>
<street>1002 Mission St.</street>
<city>san jose</city>
</address>

<address xsi:type="USAddressType">
<street>234 Fremont Blvd</street>
<city>Fremont</city>
<zip>94556</zip>
<state>CA</state>
</address>

<address xsi:type="USAddressType">
<street>100 Cardinal Way</street>
<city>Redwood City</city>
<zip>94536</zip>
<state>CA</state>
</address>

<address>
<street>100 El Camino Real</street>
<city>Sunnyvale</city>
</address>
```

데이터 통합 서비스는 기본 유형인 Address에 대해 다음 행을 반환합니다.

Street	City
1002 Mission St.	San Jose
234 Fremont Blvd	Sunnyvale
100 Cardinal Way	Redwood City
100 El Camino Real	Sunnyvale

데이터 통합 서비스는 파생된 유형인 USAddress에 대해 다음 행을 반환합니다.

Street	City	State	Zip
234 Fremont Blvd.	Sunnyvale	CA	94556
100 Cardinal Way	Redwood City	CA	94536

데이터 통합 서비스는 기본 유형의 모든 주소를 반환합니다. 데이터 통합 서비스는 파생된 유형의 모든 미국 주소를 반환합니다. 파생된 유형에는 USAddressType이 기본 유형으로부터 상속한 Street 및 City 요소가 포함됩니다.

QName 요소 구문 분석

데이터 통합 서비스는 SOAP 메시지의 QName 요소를 구문 분석할 때 스키마에 정의된 네임스페이스 접두사를 사용하기 위해 스키마의 네임스페이스에 속하는 QName 값을 업데이트합니다. 이 외에는 요소의 값을 업데이트하지 않습니다.

예를 들어 네임스페이스 "http://user/test"에 대해 정의된 네임스페이스 접두사 tns가 스키마에 있습니다. SOAP 메시지에는 동일한 네임스페이스에 대해 정의된 네임스페이스 접두사 mytns가 있습니다. 데이터 통합 서비스는 QName 값 mytns:myelement를 구문 분석할 때 값을 tns:myElement로 변경합니다.

데이터 통합 서비스는 SOAP 메시지에 QName 요소를 생성할 때 요소의 값을 업데이트하지 않습니다.

대체 그룹 구문 분석

대체 그룹은 한 요소를 동일한 그룹의 다른 요소로 바꿉니다. 대체 그룹은 각 요소 정의에 대체 그룹 이름이 포함 된다는 점을 제외하고 파생된 유형과 유사합니다.

대체 그룹의 특정 유형으로부터 요소를 수신하는 포트의 출력 그룹을 구성할 수 있습니다. 대체 그룹의 다른 유형으로부터 요소를 수신하는 포트의 출력 그룹을 여러 개 작성할 수 있습니다.

SOAP 메시지의 XML 구성 구문 분석

SOAP 메시지에 선택, 목록 및 합집합 요소 같은 XML 구성이 포함될 수 있습니다.

웹 서비스 변환은 일부 제한이 있지만 이 구성을 가진 SOAP 메시지를 구문 분석할 수 있습니다.

선택 요소

선택 요소는 하위 요소를 <선택> 선언의 요소 중 하나로 제한합니다.

다음 텍스트는 직원 또는 계약자인 사람 요소를 나타냅니다.

```
<xs:element name="person">
  <xs:complexType>
    <xs:choice>
      <xs:element name="employee" type="employee"/>
      <xs:element name="contractor" type="contractor"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

다음 방법을 사용하여 선택 요소를 매핑할 수 있습니다.

- 각 선택 요소에 대한 출력 포트를 출력 그룹에 작성합니다. 일부 요소의 경우 출력 행에 null 값이 포함됩니다.
- 각 선택에 대한 출력 그룹을 작성합니다. 위의 예의 경우 직원 그룹과 계약자 그룹을 작성합니다. 데이터 통합 서비스에서는 SOAP 메시지에 나타나는 요소를 바탕으로 행을 생성합니다.

목록 요소

목록은 "Monday Tuesday Wednesday"와 같이 여러 단순 유형 값을 포함할 수 있는 XML 요소입니다.

데이터 통합 서비스에서는 목록을 문자열 값으로 반환할 수 있습니다. SOAP 메시지에 목록이 포함될 경우 목록의 항목을 개별 출력 행으로 매핑할 수 없습니다. 매핑에서 분리해야 하는 경우 목록의 요소를 분리하는 식 변환을 구성할 수 있습니다.

합집합 요소

합집합 요소는 둘 이상의 유형이 결합된 단순 유형입니다.

다음 텍스트는 단순 유형, `size_no` 및 `size_string`의 합집합인 크기 요소를 보여 줍니다.

```
<xs:element name="Size">
  <xs:simpleType>
    <xs:union memberTypes="size_no size_string" />
  </xs:simpleType>
</xs:element>
```

크기를 출력 포트에 매핑하려면 크기에 대한 포트 하나를 작성합니다. 출력 포트를 문자열로 구성합니다. 데이터를 다른 유형으로 변환하도록 매핑에서 다른 변환을 구성할 수 있습니다.

제 8 장

웹 서비스 SOAP 메시지 생성

이 장에 포함된 항목:

- [웹 서비스 SOAP 메시지 생성 개요, 70](#)
- [변환 사용자 인터페이스, 71](#)
- [포트 및 계층 수준 관계, 72](#)
- [키, 73](#)
- [포트 매핑, 74](#)
- [여러 번 발생하는 포트 피벗, 76](#)
- [비정규화된 데이터 매핑, 77](#)
- [파생된 유형 및 요소 대체, 78](#)
- [SOAP 메시지의 XML 구성 생성, 80](#)

웹 서비스 SOAP 메시지 생성 개요

데이터 통합 서비스에서는 SOAP 메시지를 생성할 때 입력 데이터 그룹으로부터 XML 데이터를 생성합니다. 웹 서비스 소비자 변환, 웹 서비스 출력 변환 또는 결합 변환을 작성할 때는 SOAP 메시지 계층에 매핑할 입력 포트를 구성해야 합니다.

SOAP 메시지를 생성하도록 변환을 구성하려면 입력 포트의 그룹을 작성하고 각 그룹을 SOAP 메시지 계층의 그룹에 매핑합니다. SOAP 메시지의 구조는 WSDL 또는 스키마를 통해 정의됩니다.

비정규화된 입력 데이터를 바탕으로 SOAP 메시지의 데이터 그룹을 구성할 수 있습니다. 또한 다중 발생 입력 데이터를 SOAP 메시지의 다중 발생 노드에 피벗할 수도 있습니다.

데이터를 SOAP 메시지의 파생된 유형, anyType 요소 또는 대체 그룹에 매핑할 수 있습니다. 변환을 정의할 때 SOAP 메시지에서 발생할 수 있는 유형을 선택해야 합니다. 선택한 유형에 따라 작성해야 하는 입력 포트가 결정됩니다.

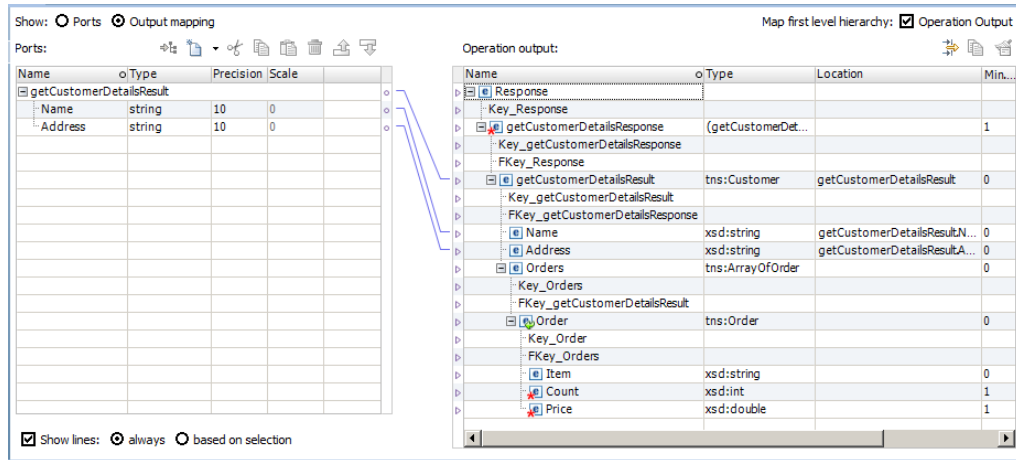
Developer tool에서 SOAP 메시지 계층을 보면 계층에 키가 포함됩니다. 키는 SOAP 메시지에 표시되지 않습니다. 키는 데이터 통합 서비스에서 SOAP 메시지에 있는 그룹 간의 상위-하위 관계를 정의할 때 사용됩니다. 키 값을 구성하려면 입력 데이터를 SOAP 메시지의 키에 매핑합니다.

변환 사용자 인터페이스

웹 서비스 출력 변환, 결합 변환 및 웹 서비스 소비자 변환에는 SOAP 메시지를 구성하는 데 사용하는 사용자 인터페이스가 포함되어 있습니다.

SOAP 메시지를 생성하도록 변환을 구성하려면 SOAP 메시지 계층과 유사한 구조로 입력 포트를 작성하십시오. WSDL 또는 스키마에서 계층의 구조를 정의합니다. 각 입력 포트를 SOAP 메시지의 노드에 매핑합니다.

다음 그림에서는 웹 서비스 출력 변환에서 입력 포트 및 SOAP 메시지 노드 간의 매핑의 보여 줍니다.



입력 포트 영역

입력 포트 영역에서 입력 포트의 그룹을 작성합니다. 매핑해야 하는 SOAP 메시지 계층의 각 수준에 대한 입력 포트를 포함합니다.

응답 또는 요청 입력 그룹과 데이터를 수신하는 하위 그룹을 작성해야 합니다.

입력 포트 그룹을 작성할 때 각 상위 그룹의 기본 키를 정의합니다. 각 하위 그룹의 외래 키를 정의합니다. 외래 키는 그룹을 상위 그룹에 연결합니다.

응답 수준 또는 WSDL 루트 수준에 대한 키는 WSDL 루트 수준에서 데이터를 전달하는 경우를 제외하고 정의할 필요가 없습니다. 예를 들어 루트 수준에는 HTTP 헤더가 포함될 수 있습니다.

고객 및 주문에 대한 다음 그룹과 유사한 포트 그룹을 작성할 수 있습니다.

```

Response
  Response_Key

Customer_Details_Root
  Key_Cust_Det
  FK_Response_Key

Customer
  Customer_ID
  FK_Cust_Det
  Name
  Address

Orders
  Order_Num
  FK_Cust_ID

Order_Items
  Order_Num
    
```

Item
Count
Price

작업 영역

작업 영역에는 WSDL 또는 스키마를 통해 정의된 SOAP 메시지 계층의 요소가 표시됩니다. WSDL 또는 스키마의 모든 요소가 SOAP 메시지에 포함될 필요는 없습니다. 메시지에는 사용자가 입력 포트에서 매핑한 데이터가 포함됩니다.

다중 발생 노드 및 복합 노드는 SOAP 메시지 구조의 계층 수준을 정의합니다. Developer tool은 키를 수준에 추가하여 수준 간의 상위-하위 관계를 작성합니다. 리프 수준을 제외한 계층의 모든 수준에는 기본 키가 있습니다. 각 하위 수준에는 상위 수준에 대한 외래 키가 있습니다. SOAP 메시지 계층에 나타나는 키는 SOAP 메시지 인스턴스에는 나타나지 않습니다. 데이터 통합 서비스에서는 SOAP 메시지를 생성할 때 키의 값을 사용하여 데이터의 수준을 연결합니다.

위치 열에는 SOAP 메시지의 요소에 대한 데이터가 포함된 그룹 이름 및 입력 포트가 포함됩니다. **위치** 열은 입력 포트를 노드에 매핑하기 전까지 비어 있습니다.

이전 그림의 SOAP 메시지에는 단일 인스턴스의 고객 세부 정보 및 주문이 포함되어 있습니다. **Orders** 그룹에는 이름이 **Order**인 다중 발생 요소가 포함되어 있습니다. SOAP 메시지 계층에는 키로 연결된 다음 수준이 포함됩니다.

```
Response
  GetCustomerDetailsResponse
    GetCustomerDetailsResult
      Orders
        Order
```

응답 수준은 응답 메시지의 루트를 나타냅니다. 데이터 통합 서비스는 이 수준을 사용하여 헤더를 SOAP 메시지에 연결합니다.

GetCustomerDetailsResponse 수준은 메시지의 루트입니다.

포트 및 계층 수준 관계

입력 포트를 SOAP 메시지 계층에 매핑하는 경우 입력 그룹과 SOAP 메시지 계층 수준 간의 관계가 유지 관리됩니다. 예를 들어 **Department**와 **Employee**라는 입력 그룹 2개가 있습니다.

Department 입력 그룹은 다음 행을 수신합니다.

Dept_num	Name	Location
101	HR	New York
102	Product	California

Employee 입력 그룹은 다음 행을 수신합니다.

Dept_num	Employee
101	Alice
101	Bob
102	Carol
102	Dave

Employee 그룹의 부서 번호를 **Department** 그룹과 **Employee** 그룹 간의 관계를 설정하는 외래 키로 매핑합니다. 부서 번호는 부서 계층 수준에서 발생하지만 직원 수준에서는 발생하지 않습니다.

SOAP 메시지에는 다음 XML 구조가 포함됩니다.

```
<department>
  <dept_num>101</dept_num>
  <name>HR</name>
  <location>New York</location>

  <employee>
    <name>Alice</name>
  </employee>

  <employee>
    <name>Bob</name>
  </employee>
</department>

<department>
  <dept_num>102</dept_num>
  <name>Product</name>
  <location>California</location>

  <employee>
    <name>Carol</name>
  </employee>

  <employee>
    <name>Dave</name>
  </employee>
</department>
```

키

SOAP 메시지 계층에는 키가 포함됩니다. 데이터 통합 서비스에서 SOAP 메시지에 XML 계층을 구성하려면 키 값이 필요합니다.

입력 포트 데이터를 SOAP 메시지 계층의 키에 매핑해야 합니다. 데이터를 제공하는 각 수준의 키에 데이터를 매핑합니다. 다중 발생 노드가 있는 경우 노드를 상위 그룹에 연결해야 합니다.

키는 유형 없이 SOAP 메시지에 나타납니다. 키에 매핑하는 포트는 문자열, 정수 또는 **bigint** 데이터 유형이어야 합니다. 상위 그룹의 기본 키와 각 하위 그룹의 외래 키는 데이터 유형, 전체 자릿수 및 소수 자릿수가 같아야 합니다. 생성된 키를 SOAP 메시지 키에 매핑할 수 있습니다.

포트를 노드에 매핑하고 동일한 계층 수준의 키에 매핑할 수 있습니다. 예를 들어 **Employee_ID**를 SOAP 메시지의 노드에 매핑하고 **Employee** 수준의 키에 매핑할 수 있습니다.

계층의 그룹 노드 2개가 상위-하위 관계에 있는 경우 다음 작업을 완료하십시오.

- 포트를 상위 노드 그룹의 기본 키에 매핑합니다.
- 포트를 하위 노드 그룹의 외래 키에 매핑합니다.

기본 키를 입력 포트에 매핑하여 기본 키가 null이거나 기본 키가 중복된 행을 제거할 수도 있습니다.

여러 포트를 동일한 키에 매핑하여 SOAP 메시지에 복합 키를 작성할 수 있습니다. 복합 키는 데이터를 비정규화하고 일부 다중 발생 값 조합에 대해 고유한 키를 유지 관리해야 하는 경우에 사용됩니다. 문자열, **bigint** 또는 정수 값을 포함하는 복합 키를 작성할 수 있습니다.

참고: 식 변환을 작업 매핑에 포함하여 키 값을 생성할 수 있습니다.

복합 키 예

다음 포트 그룹에서 고유한 Division-Department 키를 구성합니다.

```
Company
  Company_Num
  Company_Name

Division
  Company_Num
  Division_Num
  Division_Name

Department
  Division_Num
  Dept_Num
  Dept_Name
  Location
```

Dept_Num은 Division 내에서 고유하지만 Company의 모든 Division에 대해 고유하지는 않습니다.

Division 및 Department 정보를 포함하는 Department 그룹을 구성할 수 있습니다. 그러면 Division 번호와 Department 번호를 복합 키의 일부로 구성합니다.

```
Department
  Division_Num + Dept_Num (key)
  Dept_Name
  Location
```

키 값은 포트를 매핑하는 순서에 따라 결정됩니다.

포트 매핑

입력 포트를 작성한 후에 각 입력 포트를 SOAP 메시지 계층에 매핑합니다. 포트의 위치는 **작업** 영역에서 노드 옆에 표시됩니다.

다음 유형의 노드에 포트를 매핑할 수 있습니다.

원자성 노드

하위 항목이 없고 나눌 수 없는 단순 요소 또는 특성입니다.

다중 발생 원자성 노드

계층의 동일한 위치에서 여러 번 발생하는 단순 요소 또는 특성입니다.

복합 노드

다른 요소가 포함된 요소입니다.

상위 노드에 위치가 없는 경우 상위 노드가 입력 그룹 이름을 위치로 수신합니다. 상위 노드에 위치가 있는 경우 계층 수준의 각 노드에는 동일한 위치의 출력 위치가 있어야 합니다.

입력 그룹 이름을 계층 수준의 상위 노드에 매핑할 수 있습니다. Developer tool이 계층의 상위 노드에 대한 위치 필드를 업데이트합니다. Developer tool은 계층의 그룹에 속하는 하위 노드를 업데이트하지 않습니다. 입력 포트를 하위 노드에 매핑할 경우 각 입력 포트 위치는 위치 상위 노드와 동일한 위치여야 합니다.

입력 그룹을 계층 수준에 매핑한 후 나중에 변경할 수 있습니다. **지우기**를 클릭하거나 포트와 작업 영역 간의 행을 삭제할 수 있습니다. 행을 삭제하려면 행의 포인터를 끌어 선택합니다. **삭제**를 클릭합니다.

포트 매핑

포트를 SOAP 메시지의 노드에 매핑할 경우 포트를 매핑한 노드의 유형에 따라 Developer tool에서 제공하는 결과가 달라집니다.

다음 테이블에는 **작업** 영역에서 단일 포트를 서로 다른 대상 노드에 매핑할 때의 결과가 설명되어 있습니다.

대상 노드	결과
원자성 노드	단일 포트를 노드에 매핑하고 상위 노드에 위치가 없는 경우 노드가 포트의 위치를 수신합니다. 상위 노드 위치는 단일 포트에 대한 입력 그룹의 위치를 수신합니다. 단일 포트를 노드에 매핑하고 상위 노드에 이미 위치가 있는 경우 상위 노드의 위치를 변경하고 동일한 수준의 다른 하위 노드에 대한 위치를 지울 수 있습니다. 계층 수준 위치는 포트의 그룹 이름으로 변경됩니다.
다중 발생 원자성 노드 또는 다중 발생 원자성 노드의 기본 키	단일 포트를 다중 발생 원자성 노드에 매핑할 경우 Developer tool이 원자성 노드의 위치를 선택한 포트의 그룹으로 설정합니다.
복합 노드	단일 포트를 복합 노드에 매핑할 경우 Developer tool이 복합 노드의 위치를 포트가 포함된 그룹의 위치로 설정합니다. 단일 발생 원자성 노드에 포트를 할당하라는 메시지가 표시됩니다. 모든 단일 발생 원자성 노드에 위치가 있는 경우 복합 노드를 매핑할 수 없습니다.

그룹 매핑

입력 그룹을 SOAP 메시지의 노드에 매핑할 경우 포트를 매핑한 노드의 유형에 따라 Developer tool에서 제공하는 결과가 달라집니다.

다음 테이블에는 **작업** 영역에서 그룹을 노드에 매핑할 때의 결과가 설명되어 있습니다.

대상 노드	결과
원자성 노드	원자성 노드에는 그룹을 매핑할 수 없습니다.
다중 발생 원자성 노드	입력 그룹의 포트를 선택하여 노드 및 기본 키의 위치를 업데이트해야 합니다.
다중 발생 복합 노드	Developer tool이 복합 노드의 위치를 그룹의 위치로 설정합니다.

여러 포트 매핑

여러 포트를 SOAP 메시지의 노드에 매핑할 경우 포트를 매핑한 노드의 유형에 따라 Developer tool에서 제공하는 결과가 달라집니다. 동일한 그룹에서 매핑하는 경우 여러 포트를 동시에 매핑할 수 있습니다.

다음 테이블에는 여러 포트를 노드에 매핑할 때의 결과가 설명되어 있습니다.

대상 노드	결과
단일 원자성 노드	여러 포트를 단일 포트에 매핑할 때는 작업 영역에서 단일 원자성 노드 2개 이상에 대한 위치를 업데이트해야 합니다. 계층의 수준에 업데이트할 충분한 노드가 없는 경우 Developer tool이 사용 가능한 노드에 대한 포트만 매핑합니다.

대상 노드	결과
다중 발생 원자성 노드	여러 포트를 다중 발생 원자성 노드에 매핑할 경우 포트를 노드의 다중 발생에 피벗해야 합니다. Developer tool은 매핑한 포트 수에 따라 노드의 인스턴스를 작성합니다. 연결한 포트의 수를 설명하는 메시지가 표시됩니다.
다중 발생 복합 노드	여러 포트를 복합 노드에 매핑할 경우 업데이트할 단일 발생 노드 원자성 노드를 선택해야 합니다. 또한 포트를 노드의 다중 발생으로 피벗해야 합니다. Developer tool은 매핑한 포트 수에 따라 노드의 인스턴스를 작성합니다.

여러 번 발생하는 포트 피벗

여러 입력 포트를 SOAP 메시지의 여러 번 발생하는 노드에 매핑할 수 있습니다. Developer tool이 입력 데이터를 SOAP 메시지의 여러 노드에 피벗합니다.

피벗할 요소의 수를 변경하려면 **매핑 옵션** 대화 상자에서 **기존 피벗 재정의**를 선택합니다.

포트 영역에서 피벗된 포트 인스턴스 중 하나를 제거할 경우 Developer tool이 **작업** 영역에서 모든 인스턴스를 제거합니다.

피벗 예제

입력 그룹에 다음 행이 포함될 수 있습니다.

Num	Name	Location	emp_name1	emp_name2	emp_name3
101	HR	New York	Alice	Tom	Bob
102	Product	California	Carol	Tim	Dave

각 행에는 부서 번호와 세 명의 직원 이름이 포함되어 있습니다.

Employee는 SOAP 메시지 계층에서 여러 번 발생하는 노드입니다. Employee의 모든 인스턴스를 입력 행에서 SOAP 메시지 계층에 매핑할 수 있습니다. Employee의 모든 발생을 선택합니다. **매핑**을 클릭합니다. **매핑 옵션** 대화 상자에 목록에서 노드를 선택하라는 메시지가 표시됩니다.

Developer tool이 SOAP 메시지 계층의 여러 이름 노드를 포함하도록 Employee 노드를 변경합니다.

```

Department
  num
  name
  location
  Employee (unbounded)
    emp_name1
    emp_name2
    emp_name3

```

SOAP 메시지는 다음 계층을 반환합니다.

```

<department>
  <num>101</num>
  <name>HR</name>
  <location>New York</location>
  <employee>
    <emp_name>Alice</name>
  </employee>
  <employee>
    <emp_name>Tom</name>
  </employee>
  <employee>
    <emp_name>Bob</name>
  </employee>
</department>

```

```

</department>

<department>
  <num>102</num>
  <name>Product</name>
  <location>California</location>
  <employee>
    <emp_name>Carol</name>
  </employee>
  <employee>
    <emp_name>Tim</name>
  </employee>
  <employee>
    <emp_name>Dave</name>
  </employee>
</department>

```

비정규화된 데이터 매핑

비정규화된 데이터를 매핑하고 SOAP 메시지의 비정규화된 노드에 전달할 수 있습니다.

비정규화된 데이터를 매핑할 때는 하나의 입력 그룹의 데이터를 SOAP 메시지 계층의 여러 노드에 전달합니다. 다음 관계와 유사한 유형의 그룹 관계를 SOAP 메시지에 작성할 수 있습니다.

선형 노드 관계

노드 A는 노드 B의 상위 노드입니다. 노드 B는 노드 C의 상위 노드입니다. 노드 C는 노드 D의 상위 노드입니다.

계층 노드 관계

노드 A는 노드 B의 상위 노드입니다. 노드 A는 또한 노드 C의 상위 노드입니다. 노드 B와 노드 C는 관계가 없습니다.

다음 테이블에는 비정규화된 Division 및 Department 데이터가 포함된 입력 행이 표시되어 있습니다.

Division	Dept_Num	Dept_Name	Phone	Employee_Num	Employee_Name
01	100	Accounting	3580	2110	Amir
01	100	Accounting	3580	2113	Robert
01	101	Engineering	3582	2114	Stan
01	101	Engineering	3582	2115	Jim
02	102	Facilities	3583	2116	Jose

입력 데이터에는 고유한 직원 번호와 이름이 포함됩니다. Division 및 Department 데이터는 동일한 Division 및 Department의 각 직원에 대해 반복됩니다.

선형 그룹 관계

포트를 구성할 때 Division, Department 및 Employee에 대한 개별 그룹을 구성할 수 있습니다. Division은 Department의 상위 그룹이고 Department는 Employee의 상위 그룹입니다. 다음과 같은 선형 구조로 그룹을 구성할 수 있습니다.

```

Division
  Division_Key
  Division_Num
  Division Name

  Department
    Department_Key
    Division_FKey
    Dept_Num

```

```

Dept_Name
Phone

Employee
  Department_Fkey
  Employee_Num
  Employee_Name

```

Division_Num 및 Dept_Num이 입력 데이터에서 반복되지만 SOAP 메시지에는 Division 및 Department의 고유한 인스턴스가 포함됩니다. Division_Num을 Division 그룹의 기본 키로 정의합니다. Dept_Num을 Department 그룹의 기본 키로 정의합니다.

계층 그룹 관계

Division 상위 그룹과 Department 및 Employee 하위 그룹이 포함된 그룹 계층을 작성할 수 있습니다. Department와 Employee에는 기본 키-외래 키 관계가 없습니다. Department 및 Employee는 Division의 하위 그룹입니다. 다음과 같은 구조로 그룹을 구성할 수 있습니다.

```

Division
  Division_Key
  Division_Num
  Division_Name

Department
  Division_FKey
  Dept_Num
  Dept_Name

Employee
  Division_FKey
  Employee_Num
  Employee_Name

```

파생된 유형 및 요소 대체

입력 포트를 SOAP 메시지의 파생된 복합 유형, anyType 요소 및 대체 그룹에 매핑할 수 있습니다. SOAP 메시지에는 기본 유형 및 파생된 유형의 요소가 포함될 수 있습니다.

유형 관계에서 기본 유형은 다른 유형을 파생하는 유형입니다. 파생된 유형은 기본 유형의 요소를 상속받습니다. 확장된 복합 유형은 기본 유형에서 요소를 상속받는 파생된 유형이며 추가 요소를 포함합니다. 제한된 복합 유형은 기본 유형에서 일부 요소를 제한하는 파생된 유형입니다.

파생된 유형 생성

WSDL 또는 스키마에 파생된 유형이 포함될 경우 SOAP 메시지에 포함할 유형을 선택해야 합니다.

예를 들어 WSDL에서 기본 유형인 AddressType을 정의합니다. WSDL에는 파생된 AddressType인 USAddressType 및 UKAddressType도 포함됩니다.

각 유형에는 다음 요소가 포함됩니다.

- AddressType: street, city
- USAddressType(AddressType 확장): state, zipCode
- UKAddressType(AddressType 확장): postalCode, country

작업 영역에서 **USAddressType**을 선택하면 **Developer tool**이 **USAddressType** 요소에 대한 그룹을 **SOAP** 메시지에 작성합니다. 이 그룹에는 기본 주소의 **street** 및 **city**와 **USAddress**의 **state** 및 **zipCode**가 포함됩니다. 기본 유형을 확장하는 파생된 유형에는 항상 기본 유형의 요소가 포함됩니다.

SOAP 메시지의 사용 가능한 모든 파생된 유형을 선택할 경우 **Developer tool**은 다음과 유사한 그룹을 **SOAP** 계층에 작성합니다.

```
Address
  Address: Address
    Street
    City

Address:USAddressType
  Street
  City
  State
  ZipCode

Address: UKAddressType
  Street
  City
  PostalCode
  Country
```

사용자는 **Address**, **USAddress** 및 **UKAddress**에 대한 입력 포트 그룹을 정의해야 합니다.

anyType 요소 및 특성 생성

일부 스키마 요소 및 특성의 경우 **SOAP** 메시지에서 모든 데이터 유형을 사용할 수 있습니다.

anyType 요소는 전역적으로 알려진 모든 유형의 선택을 나타냅니다. **SOAP** 메시지의 **anyType** 요소에 포트를 매핑하기 전에 사용 가능한 복합 유형 또는 **xs:string**을 선택하십시오. **WSDL** 또는 스키마에 복합 유형이 포함되지 않을 경우 **Developer tool**이 **anyType** 요소 유형을 **xs:string**으로 바꿉니다.

작업 영역에서 요소 유형을 선택하려면 **anyType** 요소의 **유형** 열에서 **선택**을 클릭합니다. 사용 가능한 복합 유형 및 **xs:string**의 목록이 나타납니다.

다음 요소 및 특성의 경우 데이터의 모든 유형을 사용할 수 있습니다.

anyType 요소

요소가 연결된 XML 파일의 모든 데이터 유형이 될 수 있습니다.

anySimpleType 요소

요소가 연결된 XML 파일의 모든 **simpleType**이 될 수 있습니다.

ANY 콘텐츠 요소

요소가 스키마에서 정의된 모든 글로벌 요소가 될 수 있습니다.

anyAttribute 특성

요소가 스키마에서 이미 정의된 모든 특성이 될 수 있습니다.

대체 그룹 생성

대체 그룹은 **SOAP** 메시지에서 한 요소를 다른 요소로 바꿀 때 사용합니다. 대체 그룹은 요소 정의에 대체 그룹 이름이 포함된다는 점을 제외하고 파생된 유형과 유사한 방식으로 작동합니다.

예를 들어 기본 유형인 **Address**와 파생된 유형인 **USAddress** 및 **UKAddress**가 있습니다.

```
xs:element name="Address" type="xs:string"/>
<xs:element name="USAddress" substitutionGroup="Address"/>
<xs:element name="UKAddress" substitutionGroup="Address"/>
```

SOAP 메시지 계층을 구성할 때 **SOAP** 메시지에서 **Address** 유형을 대체할 요소를 선택할 수 있습니다.

SOAP 메시지의 XML 구성 생성

WSDL 또는 스키마에는 선택, 목록 또는 합집합 요소가 포함될 수 있습니다. 웹 서비스 변환을 사용하면 이러한 요소가 포함되는 SOAP 메시지를 생성할 수 있습니다.

선택 요소

선택 요소는 하위 요소를 <선택> 선언의 요소 중 하나로 제한합니다.

선택 요소가 포함된 SOAP 메시지에 포트를 매핑하려면 선택 구성의 모든 요소를 포함하는 하나의 입력 그룹을 작성합니다. 예를 들어 항목 설명은 차원 또는 가중치일 수 있습니다.

item: description, choice {dimension, weight}

차원일 경우 설명은 길이, 너비 및 높이를 포함하는 복합 유형입니다.

가중치일 경우 설명은 단순 문자열 유형입니다.

입력 데이터에는 다음과 같은 열 및 행이 포함됩니다.

설명	길이	너비	높이	가중치
Box	20cm	18cm	15cm	Null
Coffee	Null	Null	Null	500g

SOAP 메시지에는 차원 또는 가중치 설명이 포함된 Item 그룹이 포함됩니다.

```
Item
  Description
    Dimension
      Length
      Width
      Height
    Weight
```

입력 데이터의 NULL 값은 XML 출력에서 누락된 요소가 됩니다.

SOAP 메시지에는 다음 데이터가 포함됩니다.

```
<item>
  <desc>box</desc>
  <dimension>
    <length>20cm</length>
    <width>18cm</width>
    <height>15cm</height>
  </dimension>
</item>

<item>
  <desc>coffee</desc>
  <weight>500g</weight>
</item>
```

목록 요소

목록은 동일한 요소 또는 특성의 여러 단순 유형 값이 포함될 수 있는 XML 요소입니다. 데이터 통합 서비스에서는 입력 데이터의 목록이 통합된 문자열 데이터로 표시될 경우 목록을 처리할 수 있습니다.

목록의 각 항목이 ClassDates1, ClassDates2 및 ClassDates3과 같은 개별 요소일 경우 데이터 통합 서비스에서 항목을 목록으로 처리할 수 없습니다. SOAP 메시지의 목록을 반환해야 하는 경우 식 변환을 사용하여 항목을 문자열로 결합할 수 있습니다.

다음 입력 행에는 요일이 포함된 **ClassDates**라는 목록 요소가 포함됩니다.

CourseID	Name	ClassDates
Math 1	Beginning Algebra	Mon Wed Fri
History 1	World History	Tue Thu

데이터 통합 서비스에서는 다음 XML 구조의 SOAP 메시지를 반환할 수 있습니다.

```
<class>
  <courseId>Math 1</courseId>
  <name>Beginning Algebra</name>
  <classDates>Mon Wed Fri</classDates>
</class>
<class>
  <courseId>History 1</courseId>
  <name>World History</name>
  <classDates>Tue Thu</classDates>
</class>
```

합집합 요소

합집합 요소는 둘 이상의 유형이 결합된 단순 유형입니다. SOAP 메시지에 합집합 요소가 포함된 경우 문자열의 데이터를 포함하는 단일 입력 포트를 매핑해야 합니다.

예를 들어, SOAP 메시지에 크기라는 요소가 포함되어 있습니다. 크기는 정수 및 문자열의 합집합입니다.

```
<xs:element name="size">
  <xs:simpleType>
    <xs:union memberTypes="size_no size_string" />
  </xs:simpleType>
</xs:element>
```

입력 행에 설명 및 크기가 포함된 항목이 들어 있습니다. 항목에는 숫자 크기(예: 42)가 있을 수 있습니다. 또는 항목에 문자열 값(라지, 미디엄, 스몰)의 크기가 있을 수 있습니다.

다음 테이블에는 숫자 크기와 문자열 크기를 가진 입력 행이 표시되어 있습니다.

설명	크기
신발	42
셔츠	라지

항목 크기에 대해 하나의 포트를 작성합니다. 포트를 문자열로 매핑합니다. SOAP 메시지에 다음 요소가 포함되어 있습니다.

```
<item>
  <desc>shoes</desc>
  <size>42</size>
</item>

<item>
  <desc>shirt</desc>
  <size>large</size>
</item>
```

제 9 장

웹 서비스 소비자 변환

이 장에 포함된 항목:

- [웹 서비스 소비자 변환 개요, 82](#)
- [WSDL 선택, 84](#)
- [웹 서비스 소비자 변환 포트, 85](#)
- [웹 서비스 소비자 변환 입력 매핑, 86](#)
- [웹 서비스 소비자 변환 출력 매핑, 89](#)
- [웹 서비스 소비자 변환 고급 속성, 92](#)
- [필터 최적화, 96](#)
- [웹 서비스 소비자 변환 작성, 98](#)
- [웹 서비스 소비자 변환 예제, 100](#)

웹 서비스 소비자 변환 개요

웹 서비스 소비자 변환은 웹 서비스 클라이언트로 웹 서비스에 연결하여 데이터를 액세스하거나 변환합니다. 웹 서비스 소비자 변환은 다중 그룹 변환입니다.

웹 서비스는 **SOAP**, **WSDL** 및 **XML** 같은 공개된 표준을 사용합니다. **SOAP**는 웹 서비스의 통신 프로토콜입니다. 웹 서비스 클라이언트 요청 및 웹 서비스 응답은 **SOAP** 메시지입니다. **WSDL**은 웹 서비스 작업의 프로토콜, 형식 및 서명을 설명하는 **XML** 스키마입니다.

웹 서비스 작업에는 정보 요청, 데이터 업데이트 요청 또는 태스크 수행 요청이 포함됩니다. 예를 들어, 웹 서비스 소비자 변환에서 **getCustomerOrders**라는 웹 서비스 작업을 실행하는 **SOAP** 요청을 전송합니다. 변환에서 요청을 통해 고객 **ID**를 전달합니다. 웹 서비스 클라이언트가 고객 및 주문 정보를 검색합니다. 웹 서비스가 **SOAP** 응답으로 변환에 정보를 반환합니다.

웹 서비스 소비자 변환에서 **WSDL**, 웹 서비스 연결 또는 끝점 **URL** 입력 포트에 정의된 끝점 **URL**을 사용하여 웹 서비스에 연결합니다. 웹 서비스 연결에서 웹 서비스에 대한 보안을 활성화합니다.

SOAP 메시지

웹 서비스 소비자 변환에서는 **SOAP(Simple Object Access Protocol)**를 사용하여 웹 서비스 공급자와 정보를 교환하고 웹 서비스를 요청합니다. **SOAP**는 웹 서비스 요청 및 응답 메시지에 대한 형식을 정의합니다.

웹 서비스 소비자 변환을 사용하여 데이터를 변환할 경우 변환 시 **SOAP** 요청이 생성되고 변환이 웹 서비스에 연결됩니다. **WSDL** 개체, 웹 서비스 연결 또는 엔드포인트 **URL** 입력 포트에 정의된 엔드포인트 **URL**을 사용하여 변환이 웹 서비스에 연결합니다. **SOAP** 요청에는 요청된 작업을 실행하기 위해 웹 서비스에 필요한 정보가 포함되

어 있습니다. 웹 서비스 작업에서 SOAP 응답의 변환에 데이터를 반환합니다. 변환에서 SOAP 응답의 데이터를 매핑하고 출력 포트에 데이터를 반환합니다.

웹 서비스 소비자 변환은 ISO-8859-1 형식으로 SOAP 메시지 헤더를 인코딩합니다.

변환에서 문서/리터럴 인코딩이 포함된 SOAP 메시지를 처리할 수 있습니다. 문서/리터럴 스타일에는 SOAP 메시지를 설명하기 위한 XML 스키마가 필요합니다. SOAP 메시지는 XML로 구성됩니다. SOAP 메시지에 다중 발생 요소가 포함된 경우 요소 그룹이 XML 계층의 수준을 구성합니다. 한 수준이 다른 수준 안에 중첩될 경우 그룹이 연결됩니다.

SOAP 요청 메시지에 계층 데이터가 포함될 수 있습니다. 예를 들어, 웹 서비스 소비자 변환에서 고객 주문을 영업 데이터베이스에 추가하는 요청을 전송합니다. 변환에서 2개의 데이터 그룹을 SOAP 요청 메시지를 통해 전달합니다. 한 그룹에는 고객 ID와 이름이 포함되어 있고, 다른 그룹에는 주문 정보가 포함되어 있습니다. 주문 정보가 여러 번 발생합니다.

SOAP 응답 메시지에 계층 데이터가 포함될 수 있습니다. 예를 들어, 웹 서비스 소비자 변환에서 고객 주문에 대한 SOAP 요청을 생성합니다. 웹 서비스에서 주문 헤더 및 다중 발생 주문 세부 정보 요소를 SOAP 응답을 통해 반환합니다.

WSDL 파일

WSDL 파일에는 웹 서비스에 전달되는 데이터에 대한 설명이 포함되므로 보내는 사람과 받는 사람이 교환되는 데이터를 알 수 있습니다. 웹 서비스 소비자 변환을 작성하려면 먼저 WSDL 파일을 리포지토리에 가져와야 합니다.

WSDL에서 데이터에서 수행할 작업과 프로토콜 바인딩 또는 전송을 설명하므로 웹 서비스 소비자가 요청 메시지를 올바른 형식으로 전송할 수 있습니다. WSDL에서 웹 서비스에 연결하는 네트워크 주소를 설명합니다.

WSDL에는 SOAP 요청 및 응답 메시지를 인코딩하는 방법에 대한 정보가 포함되어 있습니다. SOAP 인코딩에서 SOAP 메시지 본문의 형식을 결정합니다. 이는 웹 서비스 소비자와 통신하기 위해 웹 서비스에서 사용하는 요청 및 응답 메시지의 형식을 설명합니다. 웹 서비스 개발자는 다양한 토크를 사용하여 웹 서비스를 작성할 수 있습니다. 토크에서 SOAP 메시지를 인코딩하는 다양한 방법을 지원합니다.

웹 서비스 소비자 변환에서는 문서/리터럴 SOAP 인코딩 스타일을 지원합니다. 웹 서비스 소비자 변환과 함께 WSDL 1.1을 사용할 수 있습니다. MIME, DIME 및 MTOM 메시지 같은 WSDL 첨부 파일은 사용할 수 없습니다.

작업

웹 서비스에는 웹 서비스가 지원하는 각 작업에 대한 작업이 포함됩니다.

예를 들어 웹 서비스에 이름이 `getcustomerid`인 작업이 포함될 수 있습니다. 이 작업은 고객 세부 정보를 사용하여 고객 이름과 응답을 검색합니다. 작업 입력에는 고객 이름에 대한 요소가 포함됩니다. 작업 출력에는 고객 이름에 따라 고객 세부 정보에 대한 요소가 포함됩니다.

웹 서비스 소비자 변환을 구성할 때는 변환이 데이터를 작업 입력에 매핑하는 방법과 변환이 작업 출력의 데이터를 매핑하는 방법을 정의합니다. 변환에서 다음 정보를 구성해야 합니다.

입력 매핑

변환 입력 포트를 웹 서비스 작업 입력 노드에 매핑하는 방법을 정의합니다. 작업 입력은 작업에 대한 SOAP 요청의 요소를 정의합니다.

출력 매핑

웹 서비스 작업 출력 노드를 변환 출력 포트에 매핑하는 방법을 정의합니다. 작업 출력은 작업에 대한 SOAP 응답의 요소를 정의합니다.

웹 서비스 보안

웹 서비스 연결에서 웹 서비스에 대한 보안을 활성화합니다. 다음과 같은 유형의 보안을 구성할 수 있습니다.

웹 서비스 보안

데이터 통합 서비스가 SOAP 요청을 웹 서비스 공급자에게 전송할 경우 웹 서비스 보안 헤더를 포함할 수 있습니다. 웹 서비스 보안 헤더에는 인증 정보가 포함되므로 웹 서비스 공급자가 데이터 통합 서비스를 인증할 수 있습니다.

웹 서비스 소비자 변환에서 사용자 이름 토큰을 제공합니다. 데이터 통합 서비스가 SOAP 요청에 별도의 보안 SOAP 헤더를 작성하고 요청을 웹 서비스 공급자에게 전달합니다.

웹 서비스 연결에서 다음 유형의 웹 서비스 보안을 사용할 수 있습니다.

- **PasswordText.** 데이터 통합 서비스가 WS-Security SOAP 헤더의 암호를 변경하지 않습니다.
- **PasswordDigest.** 데이터 통합 서비스가 암호에 임시 값과 타임스탬프를 결합시킵니다. 데이터 통합 서비스가 암호에 SHA 해시를 적용하고 base64 인코딩 형식으로 인코딩한 후 SOAP 헤더에서 해당 인코딩된 암호를 사용합니다.

TLS(Transport Layer Security)

SSL(Secure Sockets Layer)을 사용하여 TCP/IP의 전송 계층(TCP 계층) 위에 구현된 보안입니다. 웹 서비스가 보안 메시지 전송을 위해 HTTPS(Hypertext Transfer Protocol over SSL)를 웹 주소로 사용합니다. 웹 서비스 소비자 변환은 TLS 1.2, TLS 1.1 또는 TLS 1.0을 사용할 수 있습니다. 전송 계층 보안과 함께 HTTP 인증, 프록시 서버 인증 및 SSL 인증서를 사용할 수 있습니다.

SSL 인증

HTTPS 프로토콜을 통해 연결할 때 SSL 인증을 사용할 수 있습니다.

다음 유형의 SSL 인증을 사용할 수 있습니다.

- 단방향 SSL 인증
- 양방향 SSL 인증

HTTP 인증

HTTP 프로토콜을 통해 연결할 때 HTTP 인증을 사용할 수 있습니다.

다음 HTTP 인증 방법을 사용할 수 있습니다.

- 기본 인증
- 다이제스트 인증
- NTLM(NT LAN Manager) 인증

WSDL 선택

웹 서비스 소비자 변환을 작성하기 전에 WSDL 파일을 모델 리포지토리에 가져와야 합니다. SOAP 바인딩 스타일의 문서/리터럴로 정의된 WSDL 파일만 가져올 수 있습니다.

WSDL에서 사용자가 실행하려는 웹 서비스의 작업 서명을 정의합니다. WSDL을 가져올 경우 Developer tool에서 다른 변환에 재사용할 수 있는 실제 데이터 개체를 작성합니다.

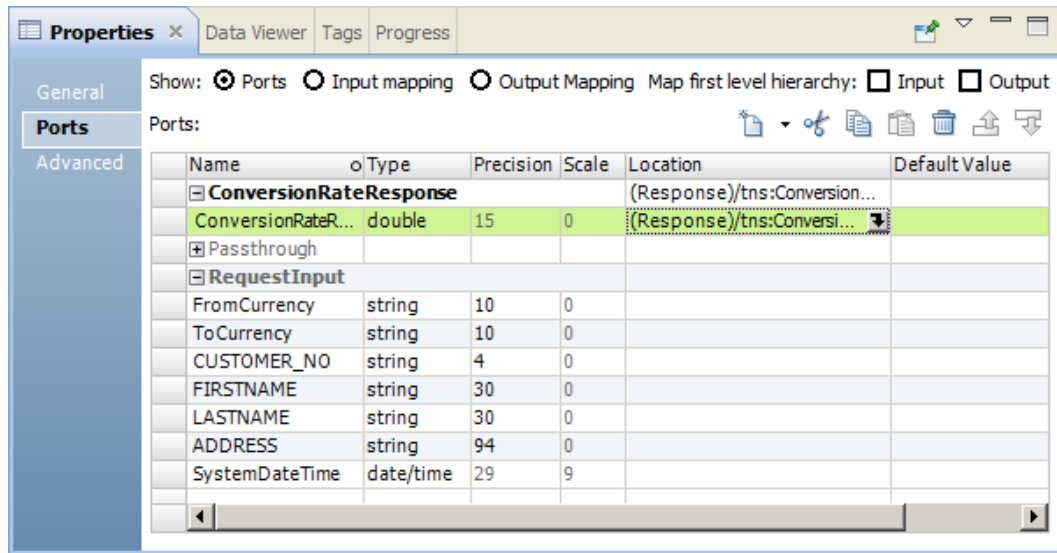
WSDL에서 여러 작업을 정의할 수 있습니다. 웹 서비스 소비자 변환을 작성할 경우 실행할 작업을 선택합니다. 웹 서비스 소비자 변환의 작업 입력 및 작업 출력 계층을 볼 수 있습니다. 계층에서 SOAP 요청 메시지 및 SOAP 응답 메시지의 구조를 정의합니다.

단방향 입력 작업으로 WSDL을 가져올 수도 있습니다. 단방향 입력 작업으로 WSDL을 가져올 경우 더미 출력 포트를 작성해야 합니다.

웹 서비스 소비자 변환 포트

변환 포트를 볼 때 작업 계층을 볼 필요가 없는 경우 포트를 표시하십시오. 포트를 표시할 경우 그룹을 정의하고, 포트를 정의하며, 작업 출력에서 출력 포트에 노드를 매핑할 수 있습니다.

다음 그림에서는 재사용 불가능한 웹 서비스 소비자 변환에 대한 포트를 보여 줍니다.



웹 서비스 소비자 변환에는 다중 입력 그룹 및 다중 출력 그룹이 있을 수 있습니다. 포트를 작성할 때 그룹을 작성하고 포트를 그룹에 추가합니다. 작업 입력 또는 작업 출력 계층의 구조를 기반으로 포트를 그룹 계층으로 정의합니다. 키를 추가하여 하위 그룹을 상위 그룹에 연결합니다. 계층의 최하위 그룹을 제외한 모든 그룹에는 기본 키가 있어야 합니다. 루트 그룹을 제외한 계층의 모든 그룹에는 외래 키가 있어야 합니다.

변환에는 RequestInput이라는 루트 입력 그룹이 포함됩니다. 이 루트 입력 그룹에 기본 키를 추가해야 합니다. 키는 문자열, bigint 또는 정수여야 합니다.

다른 통과 포트를 루트 입력 그룹에 추가할 수 있습니다. 통과 포트는 데이터를 수정하지 않고 변환에 전달합니다. 통과 포트는 입력 데이터에서 한 번만 발생할 수 있습니다. 통과 포트를 모든 출력 그룹에 추가할 수 있습니다. 출력 포트를 입력 포트에 연결합니다. SOAP 요청을 통해 전달하는 입력 값은 SOAP 응답의 출력 행에서 반복됩니다.

또한 HTTP 헤더, 쿠키 포트, 동적 URL 포트 및 웹 서비스 보안 인증 포트를 루트 입력 그룹에 추가할 수 있습니다. 루트 그룹의 데이터는 한 번 발생합니다.

작업 출력 노드를 출력 포트에 매핑하려면 위치 열의 필드를 클릭하고 위치 선택 대화 상자에서 계층을 확장합니다. 그런 다음, 계층에서 노드를 선택합니다.

HTTP 헤더 입력 포트

웹 서비스에 추가 HTTP 헤더가 필요할 수 있습니다. 루트 입력 그룹에 입력 포트를 작성하여 추가 헤더 정보를 웹 서비스 공급자에게 전달할 수 있습니다.

HTTP 헤더 및 HTTP 포트를 추가하려면 루트 입력 그룹을 선택하고 **새로 만들기** 단추 옆의 화살표를 클릭합니다. 그런 다음 **HTTP 헤더**를 클릭합니다. 헤더 이름 및 포트 이름을 입력합니다.

여러 개의 HTTP 헤더를 작성할 수 있습니다.

기타 입력 포트

미리 정의된 입력 포트를 웹 서비스 소비자 변환에 추가할 수 있습니다.

다음과 같은 미리 정의된 입력 포트를 추가할 수 있습니다.

쿠키 포트

쿠키 인증을 사용하도록 웹 서비스 소비자 변환을 구성할 수 있습니다. 원격 웹 서비스 서버는 쿠키를 기반으로 웹 서비스 소비자 사용자를 추적합니다. 매핑이 웹 서비스를 여러 번 호출하는 경우 성능을 향상시킬 수 있습니다.

쿠키 포트를 웹 서비스 요청 메시지에 연결하면 웹 서비스 공급자가 응답 메시지에 쿠키 값을 반환합니다. 쿠키 값을 매핑의 다른 변환 다운스트림에 전달하거나 파일에 쿠키 값을 저장할 수 있습니다. 쿠키 값을 파일에 저장할 때 쿠키를 웹 서비스 소비자 변환의 입력으로 구성할 수 있습니다.

쿠키 출력 포트를 웹 서비스 소비자 변환 출력 그룹 중 하나에 연결할 수 있습니다.

끝점 URL 포트

웹 서비스 소비자 변환이 끝점 URL을 사용하여 웹 서비스에 연결합니다. WSDL 파일, 웹 서비스 연결 또는 끝점 URL 입력 포트에서 끝점 URL을 정의할 수 있습니다. 변환이 포트에서 동적으로 URL을 수신할 경우 데이터 통합 서비스가 WSDL 파일 또는 웹 서비스 연결에서 정의된 URL을 재정의합니다.

웹 서비스 소비자 변환에는 각 웹 서비스 요청에 대한 하나의 URL 포트 값이 포함될 수 있습니다. 끝점 URL 포트는 루트 입력 그룹에 추가합니다.

WS-Security 포트

웹 서비스 보안은 웹 서비스 연결에서 활성화합니다. 웹 서비스 보안을 활성화할 경우 웹 서비스 연결 또는 WS-Security 입력 포트에서 사용자 이름 및 암호를 정의해야 합니다.

WS-Security 포트를 추가할 경우 변환의 입력 포트를 통해 사용자 이름과 암호를 전달해야 합니다. 변환이 포트에서 동적으로 사용자 이름과 암호를 수신할 경우 데이터 통합 서비스가 웹 서비스 연결에서 정의된 값을 재정의합니다.

참고: 웹 서비스 연결에는 HTTP 및 WS-Security 인증에 대한 하나의 사용자 이름과 암호가 포함됩니다.

미리 정의된 입력 포트를 추가하려면 **포트** 영역에서 루트 입력 그룹을 클릭합니다. **새로 만들기** 단추 옆에 있는 화살표를 클릭한 다음 **기타 포트**를 클릭합니다. 추가할 포트를 선택합니다.

웹 서비스 소비자 변환 입력 매핑

변환 포트를 보는 경우 작업 입력 계층을 보려면 입력 매핑을 표시합니다. 입력 매핑을 표시할 경우 입력 그룹 및 입력 포트를 정의하고 입력 포트를 작업 입력 노드에 매핑할 수 있습니다.

입력 매핑에는 다음 영역이 포함됩니다.

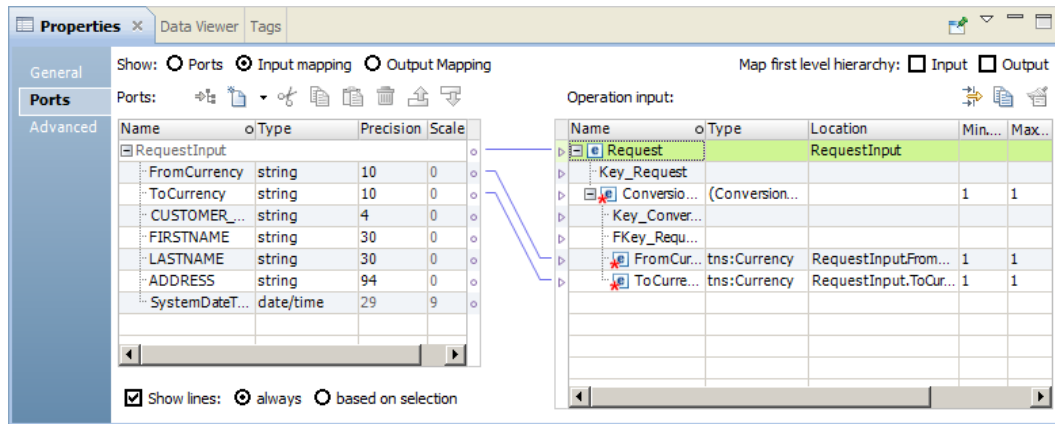
포트

포트 영역에서 변환 입력 그룹 및 입력 포트를 작성합니다.

작업 입력

작업 입력 영역에는 웹 서비스 소비자 변환이 웹 서비스에 전송하는 SOAP 요청 메시지의 노드가 표시됩니다. 변환을 작성하기 위해 사용하는 WSDL 데이터 개체에서 작업 입력 계층을 정의합니다.

다음 그림에서는 재사용 불가능한 웹 서비스 소비자 변환에 대한 입력 매핑을 보여 줍니다.



입력 포트를 작성한 후 **포트** 영역의 입력 포트를 **작업 입력** 영역의 노드에 매핑합니다. 입력 포트를 작업 입력의 노드에 매핑할 경우 포트의 위치가 **작업 입력** 영역의 위치 열에 표시됩니다.

입력 계층의 첫 번째 수준을 매핑하기로 선택한 경우 **Developer tool**에서 작업 입력의 첫 번째 수준에 있는 노드를 입력 포트에 매핑합니다. **Developer tool**은 매핑을 수행할 포트도 작성합니다. 계층의 첫 번째 수준에 하나 이상의 다중 발생 하위 노드가 있는 다중 발생 상위 노드가 포함된 경우 **Developer tool**은 계층의 첫 번째 수준을 매핑하지 않습니다.

하나의 문자열 또는 텍스트 입력 포트에서 전체 SOAP 요청 메시지로 XML 데이터를 매핑할 수 있습니다. XML 데이터를 전체 SOAP 요청으로 매핑할 경우 포트를 작업 입력의 노드에 매핑할 수 없습니다.

입력 포트를 작업 입력의 노드에 연결하는 행을 표시하도록 선택할 수도 있습니다.

입력 포트를 노드에 매핑하기 위한 규칙 및 지침

입력 포트를 작업 입력 계층의 노드에 매핑하는 경우 다음 규칙을 검토합니다.


- 입력 포트를 계층의 노드 하나에 매핑할 수 있습니다. 동일한 포트는 계층의 여러 키에 매핑할 수 있습니다.
- 입력 포트 및 노드의 데이터 유형은 호환되어야 합니다.
- 한 입력 그룹의 포트를 작업 입력의 여러 계층 수준에 매핑할 수 있습니다.
- 입력 포트를 작업 입력의 키에 매핑해야 합니다. 키에 매핑하는 포트는 문자열, 정수 또는 **bigint** 데이터 유형이어야 합니다. SOAP 메시지에 포함하는 계층 수준 이상의 작업 입력에 있는 모든 수준의 키에 데이터를 매핑하십시오. 매핑하는 수준을 포함하여 그 이상의 모든 수준에 대한 외래 키를 포함하십시오.

참고: 작업 입력 계층의 가장 낮은 수준만 매핑할 경우 입력 포트를 키에 매핑할 필요가 없습니다.

- 여러 문자열, **bigint** 또는 정수 입력 포트를 **작업 입력** 영역의 키에 매핑하여 복합 키를 작성할 수 있습니다. 복합 키의 **위치** 필드를 클릭하면 입력 포트의 순서를 다시 정렬하거나 포트 중 하나를 제거할 수 있습니다.

보기 사용자 지정의 옵션

작업 입력 영역에 키가 표시되도록 작업 입력 계층을 변경할 수 있습니다. 노드 정렬 방법을 정의하는 그룹화 구성을 표시할 수도 있습니다.

보기 사용자 지정 단추()를 **작업 입력** 영역에서 클릭합니다. 다음 옵션을 활성화합니다.

시퀀스, 선택 및 모두

요소 정의가 시퀀스, 선택 또는 모두인지 여부를 나타내는 행을 표시합니다.

모든 그룹의 노드가 SOAP 메시지에 모두 포함되어야 합니다.

시퀀스 그룹의 노드는 WSDL에 지정된 순서여야 합니다.

선택 그룹의 노드 하나 이상이 SOAP 메시지에 표시되어야 합니다.

키

작업 입력 영역에서 키를 봅니다. **작업 입력** 영역에는 각 그룹의 키가 포함됩니다. **포트** 영역의 입력 포트에 키를 추가할 수 있습니다.

작업 입력에 입력 포트 매핑

변환 입력 매핑을 표시할 때 입력 그룹을 정의하고 입력 포트를 정의하고 입력 포트를 작업 입력 노드에 매핑할 수 있습니다.

1. 웹 서비스 소비자 변환을 엽니다.
2. 변환 입력 매핑을 표시하려면 다음 방법 중 하나를 사용합니다.
 - 재사용 가능한 변환의 경우 **개요** 보기를 클릭합니다. 입력 매핑을 표시하도록 선택합니다.
 - 재사용 불가능한 변환의 경우 **속성** 보기에서 **포트** 탭을 클릭합니다. 입력 매핑을 표시하도록 선택합니다.
3. 루트 입력 그룹에 대한 기본 키를 정의합니다.
4. 입력 그룹 또는 포트를 **포트** 영역에 추가하려면 다음 방법 중 하나를 사용합니다.

옵션	설명
노드 끌기	작업 입력 영역에서 그룹 노드 또는 하위 노드를 포트 영역의 빈 열로 끌어서 놓습니다. 노드가 그룹 노드일 경우 Developer tool에서 포트 없이 그룹을 추가합니다.
그룹 또는 포트를 수동으로 추가	그룹을 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 그룹 을 클릭합니다. 포트를 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 필드 를 클릭합니다.
다른 변환에서 포트 끌기	편집기에서 다른 변환의 포트를 웹 서비스 소비자 변환으로 끌어서 놓습니다.
포트 복사	다른 변환에서 포트를 선택하여 포트 영역에 복사합니다. 포트를 복사하기 위해 키보드 바로가기기를 사용하거나 Developer tool의 복사 및 붙여넣기 단추를 사용할 수 있습니다.
계층의 첫 번째 수준 매핑 선택	계층의 첫 번째 수준 매핑 을 선택합니다. Developer tool이 작업 입력의 첫 번째 수준에 있는 노드를 입력 포트 및 그룹에 매핑합니다. Developer tool은 매핑을 수행할 입력 포트 및 그룹도 작성합니다.

5. 포트를 수동으로 작성하거나 다른 변환의 포트를 복사하는 경우 **작업 입력** 영역에서 **위치** 열을 클릭하고 목록에서 포트를 선택합니다.

6. 입력 포트를 복합 키로 매핑하려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
입력 포트 끌기	두 개 이상의 입력 포트를 선택하고 작업 입력 계층의 키로 끌어서 놓습니다.
위치 선택 대화 상자에서 입력 포트 선택	작업 입력 계층에서 키의 위치 열을 클릭한 다음 입력 포트를 선택합니다.

7. 노드의 위치를 지우려면 다음 방법 중 하나를 사용하십시오.

옵션	설명
지우기 클릭	작업 입력 영역에서 하나 이상의 노드를 선택하고 지우기를 클릭합니다.
포트를 노드에 연결하는 행 삭제	입력 포트를 작업 입력의 노드에 연결하는 하나 이상의 행을 선택하고 삭제를 누릅니다.

8. 연결된 WSDL 데이터 개체에 anyType 요소, 임의 요소, anyAttribute 특성, 파생된 유형 요소 또는 대체 그룹이 포함되는 경우 **작업 입력** 영역에서 개체를 선택합니다. 노드의 **유형** 열에서 **선택**을 클릭한 다음, 목록에서 하나 이상의 유형, 요소 또는 특성을 선택합니다.
9. 문자열 또는 텍스트 입력 포트의 XML 데이터를 전체 SOAP 요청에 매핑하려면 포트를 마우스 오른쪽 단추로 클릭하고 **XML로 매핑**을 선택합니다.

웹 서비스 소비자 변환 출력 매핑

변환 포트를 보는 경우 작업 출력 계층을 보려면 출력 매핑을 표시하십시오. 출력 매핑을 표시할 경우 출력 그룹을 정의하고, 출력 포트를 정의하며, 작업 출력 노드를 출력 포트에 매핑할 수 있습니다.

출력 매핑에는 다음 영역이 포함됩니다.

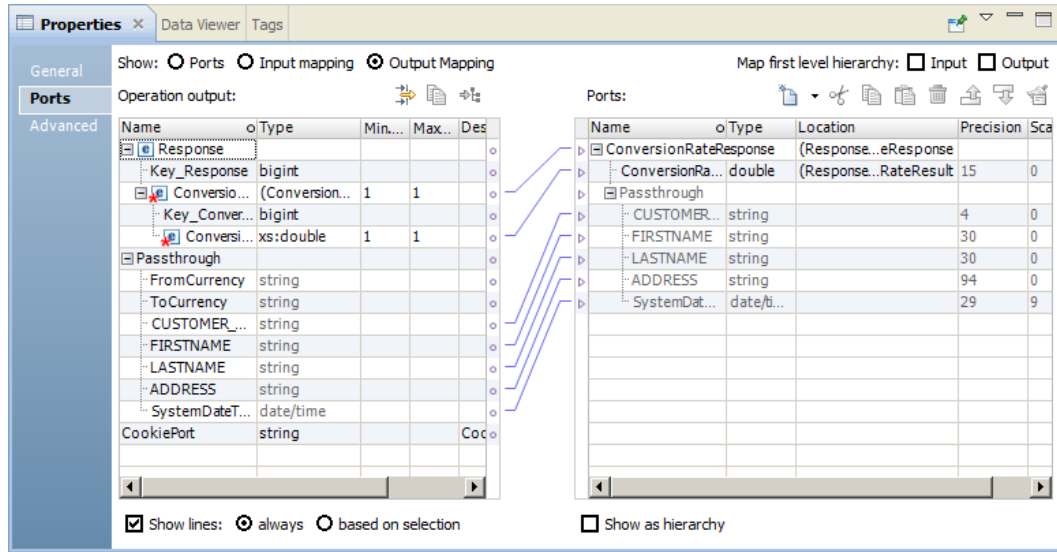
작업 출력

작업 출력 영역에는 웹 서비스가 웹 서비스 소비자 변환에 반환하는 SOAP 응답 메시지의 노드가 표시됩니다. 변환을 작성하기 위해 사용하는 WSDL 데이터 개체에서 작업 출력 계층을 정의합니다.

포트

포트 영역에서 변환 출력 그룹 및 포트를 작성합니다.

다음 그림에서는 재사용 불가능한 웹 서비스 소비자 변환에 대한 출력 매핑을 보여 줍니다.



출력 포트를 작성한 후 **작업 출력** 영역의 노드를 **포트** 영역의 포트에 매핑합니다. 작업 입력에서 출력 포트에 노드를 매핑할 경우 노드 위치가 **포트** 영역의 **위치** 열에 표시됩니다.

출력 계층의 첫 번째 수준을 매핑하기로 선택한 경우 **Developer tool**에서 작업 출력의 첫 번째 수준에 있는 노드를 출력 포트에 매핑합니다. **Developer tool**은 매핑을 수행할 포트도 작성합니다. 계층의 첫 번째 수준에 하나 이상의 다중 발생 하위 노드가 있는 다중 발생 상위 노드가 포함된 경우 **Developer tool**은 계층의 첫 번째 수준을 매핑하지 않습니다.

계층에 출력 포트를 표시하도록 선택할 수 있습니다. 각 하위 그룹은 상위 그룹 아래에 나타납니다. 작업 출력의 노드를 출력 포트에 연결하는 행을 표시하도록 선택할 수도 있습니다.

연결된 WSDL 데이터 개체가 리포지토리에서 삭제된 경우 **Developer tool**이 작업 노드의 위치를 출력 매핑에 유지합니다. 출력 매핑을 표시할 경우 **포트** 영역에서 출력 포트의 **위치** 열에 작업 노드의 위치를 계속 표시합니다. 다른 WSDL을 변환과 연결할 경우 **Developer tool**에서 각 위치가 유효한지 확인합니다. 위치가 더 이상 유효하지 않을 경우 **Developer tool**이 출력 매핑의 **포트** 영역에서 작업 노드의 위치를 지웁니다.

노드를 출력 포트에 매핑하기 위한 규칙 및 지침

작업 출력 계층의 노드를 출력 포트에 매핑할 경우 다음 규칙을 검토하십시오.

- 작업 출력 노드 및 출력 포트에는 호환 가능한 데이터 유형이 있어야 합니다.
- 그룹의 둘 이상의 출력 포트에 노드를 매핑할 수 없습니다.
- 통과 포트를 제외한 모든 출력 포트에는 올바른 위치가 있어야 합니다.
- 다중 발생 하위 노드를 빈 출력 포트에 끝 경우 그룹을 다른 출력 그룹에 연결해야 합니다. 그룹을 선택하면 **Developer tool**이 그룹을 연결하는 키를 작성합니다.
- 여러 번 발생하는 요소를 상위 요소가 포함된 그룹으로 끝면 포함할 하위 요소 발생 수를 구성할 수 있습니다. 또는 상위 그룹을 변환 출력에서 여러 번 발생하는 하위 그룹으로 대체할 수 있습니다.

SOAP 메시지를 XML로 매핑

데이터를 개별 출력 포트에 반환하지 않고 전체 SOAP 메시지를 XML로 매핑할 수 있습니다.


SOAP 메시지를 XML로 매핑하면 데이터 통합 서비스에서 전체 SOAP 메시지를 하나의 포트에 반환합니다. 출력 포트를 작성하지 마십시오.

전체 메시지를 매핑하려면 **작업 출력** 영역의 루트 그룹을 마우스 오른쪽 단추로 클릭합니다. **XML로 매핑**을 선택합니다.

Developer tool에서 문자열 출력 포트가 작성됩니다. 전체 자릿수는 65535바이트입니다.

보기 사용자 지정의 옵션

작업 출력 영역에 쿠키 포트, 통과 포트 및 키가 표시되도록 작업 출력 계층을 변경할 수 있습니다. 노드 정렬 방법을 정의하는 그룹화 구성을 표시할 수도 있습니다.

보기 사용자 지정 단추()를 **작업 출력** 영역에서 클릭합니다. 다음 옵션을 활성화합니다.

시퀀스, 선택 및 모두

요소 정의가 시퀀스, 선택 또는 모두인지 여부를 나타내는 행을 표시합니다.

모든 그룹의 노드가 SOAP 메시지에 모두 포함되어야 합니다.

시퀀스 그룹의 노드는 WSDL에 지정된 순서여야 합니다.

선택 그룹의 노드 하나 이상이 SOAP 메시지에 표시되어야 합니다.

키

작업 출력 영역에서 키를 봅니다. **작업 출력** 영역에는 각 그룹의 키가 포함됩니다. **포트** 영역의 출력 포트에 키를 추가할 수 있습니다.

통과 포트

작업 출력 영역에 통과 포트가 표시됩니다. 통과 포트는 데이터를 변경하지 않고 변환을 통해 데이터를 전달하는 포트입니다. 작업 출력의 통과 포트를 모든 웹 서비스 소비자 변환 출력 그룹에 연결할 수 있습니다. 통과 포트는 데이터를 한 번 수신하므로 응답 메시지의 SOAP 수준에 위치합니다.

쿠키 포트

쿠키 포트를 표시합니다. 쿠키 인증을 구성한 경우 원격 웹 서비스 서버가 쿠키를 바탕으로 웹 서비스 소비자 사용자를 추적합니다. 요청 메시지에서 웹 서비스 쿠키를 연결한 경우 웹 서비스가 응답 메시지에서 쿠키를 반환합니다. 작업 출력의 쿠키를 모든 웹 서비스 소비자 변환 출력 그룹에 연결할 수 있습니다.

출력 포트로 작업 출력 매핑

변환 출력 매핑을 표시할 때 출력 그룹을 정의하고 출력 포트를 정의하고 작업 출력 노드를 출력 포트에 매핑할 수 있습니다.

1. 웹 서비스 소비자 변환을 엽니다.
2. 변환 출력 매핑을 표시하려면 다음 방법 중 하나를 사용합니다.
 - 재사용 가능한 변환의 경우 **개요** 보기를 클릭합니다. 출력 매핑을 표시하도록 선택합니다.
 - 재사용 불가능한 변환의 경우 **속성** 보기에서 **포트** 탭을 클릭합니다. 출력 매핑을 표시하도록 선택합니다.
3. 출력 그룹 또는 포트를 **포트** 영역에 추가하려면 다음 방법 중 하나를 사용합니다.

옵션	설명
노드 끌기	작업 출력 영역의 그룹 노드 또는 하위 노드를 포트 영역의 빈 열로 끌어서 놓습니다. 노드가 그룹 노드일 경우 Developer tool에서 포트 없이 그룹을 추가합니다.

옵션	설명
그룹 또는 포트를 수동으로 추가	그룹을 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 그룹 을 클릭합니다. 포트를 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 필드 를 클릭합니다.
다른 변환에서 포트 끌기	편집기에서 다른 변환의 포트를 웹 서비스 소비자 변환으로 끌어서 놓습니다.
포트 복사	다른 변환에서 포트를 선택하여 포트 영역에 복사합니다. 포트를 복사하기 위해 키보드 바로 가기를 사용하거나 Developer tool의 복사 및 붙여넣기 단추를 사용할 수 있습니다.
계층의 첫 번째 수준 매핑 선택	계층의 첫 번째 수준 매핑 을 선택합니다. Developer tool이 작업 출력의 첫 번째 수준에 있는 노드를 출력 포트 및 그룹에 매핑합니다. Developer tool은 매핑을 수행할 출력 포트 및 그룹도 작성합니다.

4. 포트를 수동으로 작성하거나 다른 변환의 포트를 복사하는 경우 **포트** 영역에서 **위치** 열을 클릭하고 목록에서 노드를 선택합니다.
5. 포트의 위치를 지우려면 다음 방법 중 하나를 사용합니다.

옵션	설명
지우기 클릭	포트 영역에서 하나 이상의 포트를 선택하고 지우기 를 클릭합니다.
노드를 포트에 연결하는 행 삭제	작업 출력의 노드를 출력 포트에 연결하는 하나 이상의 행을 선택하고 삭제 를 누릅니다.

6. 연결된 WSDL 데이터 개체에 **anyType** 요소, 임의 요소, **anyAttribute** 특성, 파생된 유형 요소 또는 대체 그룹이 포함되는 경우 **작업 출력** 영역에서 개체를 선택합니다. 노드의 **유형** 열에서 **선택**을 클릭한 다음, 목록에서 하나 이상의 유형, 요소 또는 특성을 선택합니다.
7. 전체 SOAP 응답 메시지를 XML로 매핑하려면 **작업 출력** 영역에서 루트 그룹을 마우스 오른쪽 단추로 클릭하고 **XML로 매핑**을 선택합니다.

웹 서비스 소비자 변환 고급 속성

웹 서비스 소비자 변환 고급 속성에는 추적 수준, 일반 결함 포트, 웹 서비스 연결 및 동시 웹 서비스 요청 메시지가 포함되어 있습니다.

고급 탭에서 웹 서비스 소비자 변환에 대한 다음 고급 속성을 정의할 수 있습니다.

추적 수준

이 변환에 대해 로그에 표시되는 세부 정보의 양입니다. 간단, 보통, 자세한 정보 표시 초기화 또는 자세한 정보 표시 데이터를 선택할 수 있습니다. 기본값은 보통입니다.

SOAP 작업

WSDL에 정의된 SOAP 작업 값을 웹 서비스 소비자 변환에 대한 상수 값으로 재정의합니다.

일반 SOAP 결함 처리 활성화

WSDL에 정의되지 않은 결함 메시지를 반환합니다. **GenericFault** 출력 그룹에 출력 포트를 작성하여 결함 코드 및 메시지를 처리합니다.

다음 테이블에는 SOAP 1.1 및 SOAP 1.2에 대한 결함 출력 포트가 설명되어 있습니다.

SOAP 1.1용 결함 출력 포트	SOAP 1.2용 결함 출력 포트	설명
결함 코드	코드*	결함 ID 코드를 반환합니다.
결함 문자열	이유*	결함 메시지에 오류 설명을 반환합니다.
결함 세부 정보	세부 정보	일반 결함 메시지를 통해 웹 서비스 공급자가 웹 서비스 소비자 변환에 전달하는 사용자 지정 정보를 반환합니다.
결함 작업자	역할	결함을 발생시킨 개체에 대한 정보를 반환합니다.
-	노드	결함을 생성한 SOAP 노드의 URI를 반환합니다.
* 코드 및 이유 출력 포트는 계층적입니다.		

참고: 코드 결함 출력 포트를 확장하여 SubCode 결함 출력 포트를 한 수준 위로 추출할 수 있습니다.

HTTP 오류 처리 활성화

웹 서비스에서 HTTP 오류를 반환합니다. GenericFault 출력 그룹에 HTTP 오류 출력 포트를 작성합니다.

결함을 오류로 처리

결함 메시지를 매핑 로그에 추가합니다. 결함이 발생하면 데이터 통합 서비스가 매핑에 대한 오류 수를 증분합니다. 초기 선택 및 푸시인 최적화를 허용하려면 이 속성을 비활성화합니다. 기본값은 활성화됩니다.

연결

웹 서비스에 연결할 웹 서비스 연결 개체를 식별합니다. Developer tool에서 웹 서비스 연결을 작성합니다. Developer tool 또는 Administrator 도구에서 웹 서비스 연결을 편집합니다. 웹 서비스 연결을 구성할 경우 끝점 URL, 웹 서비스에 필요한 보안 유형 및 연결 제한 시간을 구성합니다.

웹 서비스 소비자 변환이 끝점 URL을 사용하여 웹 서비스에 연결합니다. WSDL 파일, 웹 서비스 연결 또는 끝점 URL 입력 포트에서 끝점 URL을 정의할 수 있습니다.

웹 서비스 연결을 구성하는 경우를 확인하려면 다음 지침을 사용하십시오.

- WSDL 파일의 URL과 다른 끝점 URL을 사용할 경우, 그리고 끝점 URL 입력 포트를 사용하지 않을 경우 연결을 구성합니다.
- 연결하는 웹 서비스에 웹 서비스 보안, HTTP 인증 또는 SSL 인증서가 필요한 경우 연결을 구성합니다.
- 기본 연결 제한 시간을 변경할 경우 연결을 구성합니다.

참고: 리포지토리의 WSDL 데이터 개체를 웹 서비스 연결과 연관시킬 수 있습니다. 연관된 연결은 해당 WSDL에서 작성하는 모든 웹 서비스 소비자 변환의 기본 연결이 됩니다.

압축 활성화

GZIP 압축 방법을 사용하여 SOAP 요청의 코딩을 활성화하고 GZIP 또는 deflate를 사용하여 SOAP 응답의 디코딩을 활성화합니다.

XML 스키마 유효성 검사

런타임 시 SOAP 응답 메시지의 유효성을 검사합니다. **올바르지 않은 XML에 대한 오류 또는 유효성 검사 없음**을 선택합니다.

정렬된 입력

데이터 통합 서비스에서 모든 입력 데이터를 처리하지 않고도 출력을 생성할 수 있습니다. 입력 데이터를 작업 입력 계층의 키별로 정렬한 경우 정렬된 입력을 활성화합니다.

푸시인 최적화

푸시인 최적화를 활성화합니다. **푸시인 최적화** 속성의 **열기** 단추를 클릭하여 필터 값을 수신하는 필터 포트를 선택합니다. 각 필터 포트에 대해 필터링된 열을 웹 서비스 응답에 포함하는 출력 포트를 선택합니다.

부작용 있음

웹 서비스가 행 반환 이외의 다른 기능을 수행함을 나타내는 확인란입니다. 웹 서비스가 행 반환 이외에 개체를 수정하거나 다른 개체 또는 함수와 상호 작용할 경우 웹 서비스 소비자 변환에 부작용이 있습니다. 웹 서비스에서 데이터베이스를 수정하거나, 합계에 추가하거나, 예외를 발생시키거나, 이메일을 작성하거나, 부작용이 있는 다른 웹 서비스를 호출할 수 있습니다. 푸시인 최적화 또는 초기 선택 최적화를 허용하기 위해 **부작용 있음** 속성을 비활성화합니다. 기본값은 활성화됩니다.

동시 실행 활성화

웹 서비스 소비자 변환을 활성화하여 여러 웹 서비스 요청을 병렬로 보내도록 웹 서비스에 대한 여러 동시 연결을 작성할 수 있습니다. 웹 서비스 소비자 변환을 활성화하여 웹 서비스에 대한 여러 동시 연결을 작성할 때 메모리 소비 제한 및 동시 연결 수 제한을 설정할 수 있습니다.

다음 테이블에는 옵션이 설명되어 있습니다.

옵션	설명
동시 실행 활성화	웹 서비스에 대한 여러 동시 연결을 작성합니다.
동시 실행 연결 제한	동시 실행 웹 서비스 연결 수입니다. 기본값은 20입니다.
총 동시 실행 메모리 제한(MB)	모든 동시 실행 연결에 대한 총 메모리 할당 한도입니다. 기본값이 100MB입니다.

웹 서비스 오류 처리

SOAP 결함 및 HTTP 오류 다운스트림을 매핑에서 전달하도록 웹 서비스 소비자 변환을 구성할 수 있습니다. 결함이 발생할 경우 오류 수를 증분할 수 있습니다. 변환 고급 속성에서 웹 서비스 오류 처리를 구성하십시오.

웹 서비스에서 응답 메시지를 반환하거나 결함을 반환합니다. 결함은 오류입니다. 웹 서비스는 발생하는 오류를 기반으로 다른 결함을 생성할 수 있습니다.

웹 서비스 소비자 변환에서 다음 유형의 결함을 반환할 수 있습니다.

SOAP 결함

WSDL에서 정의하는 SOAP 오류입니다. 웹 서비스 응답 메시지에서 결함을 반환하는 출력 오류 포트를 구성하십시오. SOAP 1.1 바인딩의 경우 데이터 통합 서비스가 결함에 대한 결함 메시지, 결함 코드, 결함 문자열 및 결함 작업자 요소를 반환합니다. SOAP 1.2 바인딩에 대해서는 데이터 통합 서비스가 결함에 대한 결함 메시지, 코드, 이유, 노트 및 역할 요소를 반환합니다.

일반 SOAP 결함

웹 서비스가 런타임 시 일반 SOAP 결함을 생성합니다. SOAP 1.1 바인딩 및 SOAP 1.2 바인딩의 경우 결함 요소가 다릅니다. WSDL에서 일반 SOAP 결함을 정의하지 않습니다. 일반 SOAP 결함에는 인증 실패 및 SOAP 요청 오류가 포함됩니다.

HTTP 오류

변환에서 HTTP 오류 처리를 활성화한 경우 Developer tool이 HTTP 결함 출력 포트를 추가합니다. 데이터 통합 서비스가 단일 문자열 포트를 통해 웹 서비스의 HTTP 오류를 반환합니다. HTTP 오류에는 오류 코드와 메시지가 포함됩니다.

웹 서비스의 SOAP 응답에 유효하지 않은 XML 데이터가 있는 경우 웹 서비스 소비자 변환에서 오류를 반환합니다.

SOAP 결함을 오류로 처리할지 구성할 수 있습니다. 결함을 오류로 처리를 활성화하고 SOAP 오류가 발생한 경우 데이터 통합 서비스가 매핑에 대한 오류 수를 증분합니다. 결함이 메시지 로그에 표시됩니다.

메시지 압축

SOAP 메시지 압축을 활성화하면 웹 서비스 소비자 변환이 웹 서비스 요청 메시지를 압축하고 압축된 웹 서비스 응답 메시지를 수신합니다.

웹 서비스 소비자 변환이 GZip 압축으로 SOAP 요청을 인코딩합니다. 또한 GZip 또는 deflate 압축으로 인코딩된 응답 메시지를 허용합니다.

데이터 통합 서비스는 웹 서비스에서 응답을 받을 때 SOAP 메시지의 Content-Encoding HTTP 헤더를 확인하고 메시지를 디코딩합니다.

기본값은 압축 인코딩 없음입니다. 웹 서비스는 SOAP 응답을 압축하지 않습니다.

다음 테이블에는 압축을 켜거나 끌 때의 요청 및 응답 메시지의 헤더가 표시되어 있습니다.

압축	헤더
켜짐	Content-Encoding 헤더: GZip Accept-Encoding 헤더: GZip, deflate
꺼짐	빈 Content-Encoding 헤더 빈 Accept-Encoding 헤더

웹 서비스에서 기본 압축으로 응답 메시지를 인코딩하는 경우도 간혹 있습니다. 웹 서비스 소비자 변환은 GZip 또는 deflate로 인코딩된 메시지를 디코딩합니다. 웹 서비스에서 예기치 않게 응답 메시지를 인코딩할 경우 웹 서비스 소비자 변환이 매핑 로그에 메시지를 기록합니다.

압축은 변환의 고급 속성에서 활성화할 수 있습니다.

동시 실행

웹 서비스 소비자 변환을 활성화하여 여러 웹 서비스 요청을 병렬로 보내도록 웹 서비스에 대한 여러 동시 연결을 작성할 수 있습니다.

예를 들어 은행 정보를 쿼리하는 동안 웹 서비스 소비자 변환을 동시 실행에 대해 구성하여 여러 행을 병렬로 보낼 수 있습니다. 입력 행이 20개인 경우 20개 요청을 동시에 보내 처리를 가속화할 수 있습니다.

웹 서비스 소비자 변환에서 동시 실행을 활성화할 경우 총 메모리 사용 제한을 구성할 수 있습니다.

웹 서비스 소비자 변환에서 동시 실행을 활성화할 경우 동시 웹 서비스 연결 수를 구성할 수 있습니다.

동시 실행에 대한 규칙 및 지침

동시 실행을 사용하는 동안에는 다음 규칙 및 지침을 사용합니다.

- 동시 실행은 정렬된 입력 행을 웹 서비스에 대한 여러 개의 동시 연결로 지원합니다. 순서가 지정된 출력 행은 지원되지 않습니다.
- 데이터 집합이 100개 행을 초과할 경우 동시 실행을 사용합니다.
- 동시 웹 서비스 연결의 수를 늘리지 않는 것이 좋습니다. 동시 웹 서비스 연결 수는 운영 체제에서 사용하는 소켓 수에 연결됩니다. 소켓 수를 늘리면 비용이 증가합니다.
- 동시 실행 기능을 사용하는 동안 성능을 최적화하려면 최소 100MB 이상의 RAM이 포함된 멀티코어 프로세서 시스템을 사용합니다.
- 동시 실행 메모리 제한은 웹 서비스를 호출하는 동안 동시 작업 흐름에서 사용하는 메모리를 나타냅니다.
- 웹 서비스 소비자 변환에서 동시 실행을 활성화할 경우 메모리 소비 제한을 구성할 수 있습니다. 메모리 사용량이 서버의 실제 RAM을 초과하지 않도록 하십시오.

동시 실행을 위한 모범 사례

동시 실행을 사용하는 동안 성능을 최적화하려면 다음 모범 사례를 따르십시오.

- 총 동시 실행 메모리 제한 및 동시 실행 연결 제한의 기본 값을 변경하지 않습니다.
- 행이 100개 미만인 데이터 집합에 대해 동시 실행을 사용하지 않습니다.
- 동시 실행을 사용하는 동안 매핑에 통과 포트를 사용하지 않습니다.

필터 최적화

필터 최적화는 매핑을 통과하는 행 수를 줄여 성능을 향상시킵니다. 데이터 통합 서비스에서는 초기 선택 최적화 또는 푸시인 최적화를 적용할 수 있습니다.

데이터 통합 서비스는 필터 최적화 방법을 적용할 때 매핑에서 가능한 소스에 근접한 위치로 필터를 이동합니다. 데이터 통합 서비스가 매핑에서 변환 앞으로 필터를 이동할 수 없는 경우에는 필터 논리를 변환으로 푸시할 수 있습니다.

웹 서비스 소비자 변환의 초기 선택 최적화 활성화

웹 서비스 소비자 변환에 부작용이 없고 변환이 결함을 오류로 처리하지 않을 경우 변환에 대한 초기 선택 최적화를 활성화합니다.

1. 웹 서비스 소비자 변환의 **고급 속성** 보기를 엽니다.
2. **결함을 오류로 처리**를 선택 취소합니다.
3. **부작용 있음**을 선택 취소합니다.

웹 서비스 소비자 변환의 푸시인 최적화

변환이 SQL 데이터 서비스의 가상 테이블에 있는 경우 웹 서비스 소비자 변환을 사용하여 푸시인 최적화를 구성할 수 있습니다. 웹 서비스 변환이 매핑에 있을 때 변환을 사용하여 푸시인 최적화를 구성할 수 있습니다.

매핑은 웹 서비스를 호출하여 최종 사용자 SQL 쿼리의 문에 따라 데이터 집합 또는 데이터의 하위 집합을 검색합니다. 최종 사용자 SQL 쿼리에는 선택적 필터 조건이 포함됩니다.

웹 서비스 소비자 변환은 푸시인 최적화를 통해 필터 포트의 필터 값을 검색합니다. 필터 포트는 푸시인 최적화를 구성할 때 필터 포트인식하는 연결되지 않은 입력 포트입니다. 필터 포트에는 최종 사용자 쿼리에 필터가 포함되지 않을 경우 웹 서비스가 모든 행을 반환하도록 하는 기본값이 있습니다. 필터 포트는 통과 포트가 아닙니다.

참고: 필터 필드는 웹 서비스 요청에서 루트 그룹의 일부여야 합니다.

필터 포트를 구성할 때는 웹 서비스 소비자 변환에서 웹 서비스 응답의 열 데이터를 수신할 출력 포트를 식별해야 합니다. 예를 들어 필터 포트가 **EmployeeID**라는 입력 포트일 경우 응답의 출력 포트는 **EmployeeNum**이라는 포트가 될 수 있습니다. **Developer tool**에서 가상 테이블 읽기의 필터 논리를 웹 서비스 소비자 요청으로 푸시하려면 입력 필터 포트와 출력 포트가 연결되어야 합니다. 일반적으로 웹 서비스 요청에 대한 입력 포트는 웹 서비스 응답의 출력 포트와 다릅니다.

필터 포트를 구성할 때는 웹 서비스 소비자 변환에서 웹 서비스 응답의 열 데이터를 수신할 출력 포트를 식별해야 합니다. 예를 들어 필터 포트가 **EmployeeID**라는 입력 포트일 경우 응답의 출력 포트는 **EmployeeNum**이라는 포트가 될 수 있습니다. **Developer tool**에서 테이블 읽기의 필터 논리를 웹 서비스 소비자 요청에 푸시하려면 입력 필터 포트와 출력 포트가 연결되어야 합니다. 일반적으로 웹 서비스 요청에 대한 입력 포트는 웹 서비스 응답의 출력 포트와 다릅니다.

필터 필드는 통과 포트가 될 수 없습니다. 필터 포트를 구성하면 포트의 기본값이 필터 조건의 값으로 변경되므로 통과 출력 포트 값도 변경됩니다. 출력 통과 포트에 기반하는 필터는 예기치 않은 결과를 반환합니다.

여러 필터 식을 웹 서비스 소비자 변환에 푸시할 수 있습니다. 각 필터 조건은 다음 형식이어야 합니다.

<필드> = <상수>

필터 조건은 **AND**로 조인되어야 합니다. 조건을 **OR**로 조인할 수 없습니다.

웹 서비스 소비자 변환의 푸시인 최적화 예제

SQL 데이터 서비스는 사용자로부터 수신한 SQL 쿼리에 따라 모든 고객에 대한 주문을 반환하거나 특정 고객에 대한 주문을 반환합니다.

데이터 서비스에는 다음 구성 요소의 논리적 데이터 개체가 포함됩니다.

Customer 테이블

고객 정보가 포함된 Oracle 데이터베이스 테이블입니다.

웹 서비스 소비자 변환

웹 서비스를 호출하여 고객의 최신 주문을 검색하는 변환입니다. 웹 서비스 소비자 변환에는 **customerID** 및 **orderNum**에 대한 입력 포트가 있습니다. 이 변환에는 **Customer** 테이블로부터 수신한 고객 데이터가 포함된 통과 포트가 있습니다. **orderNum** 포트는 필터 포트이고 연결되지 않습니다. **orderNum**의 기본값은 "*"입니다. 웹 서비스에서 웹 서비스 요청을 통해 이 값을 수신할 경우 모든 주문이 반환됩니다.

Orders 가상 테이블

웹 서비스의 고객 및 주문 데이터를 수신하는 가상 테이블입니다. 최종 사용자는 이 테이블을 쿼리합니다. **Orders**에는 고객 열, **orderID** 열 및 고객 및 주문 데이터가 포함됩니다.

최종 사용자는 다음 SQL 쿼리를 SQL 데이터 서비스에 전달합니다.

```
SELECT * from OrdersID where customer = 23 and orderID = 56
```

데이터 통합 서비스가 쿼리를 분할하여 매핑을 최적화합니다. 데이터 통합 서비스는 초기 선택 최적화를 사용하여 필터 논리 **customer = 23**을 **Customer** 테이블 읽기로 이동합니다. 데이터 통합 서비스는 푸시인 최적화를 사용하여 필터 논리 **orderID = 56**을 웹 서비스 소비자 변환 필터 포트에 푸시합니다. 웹 서비스 소비자 변환이 고객 **23**에 대한 **ordersID 56**을 검색합니다.

웹 서비스 소비자 변환의 푸시인 최적화 활성화

웹 서비스 소비자 변환에 부작용이 없고 변환이 결함을 오류로 처리하지 않을 경우 변환에 대한 푸시인 최적화를 활성화합니다.

1. 웹 서비스 소비자 변환의 **고급 속성** 보기를 엽니다.
2. **결함을 오류로 처리**를 선택 취소합니다.
3. **부작용 있음**을 선택 취소합니다.
4. **푸시인 최적화** 속성에서 **열기** 단추를 클릭합니다.
5. 입력 최적화 대화 상자에서 필터 포트 이름을 선택합니다.
여러 필터 포트를 선택할 수 있습니다.
6. **출력 열**을 클릭합니다.
7. 각 필터 포트에 대해 필터링된 열을 웹 서비스 응답에 포함하는 출력 포트를 선택합니다.
8. 각 필터 포트에 대한 기본값을 입력합니다.

참고: 웹 서비스 소비자 포트가 필터 포트가 아닌 경우 기본값을 구성할 수 없습니다.

웹 서비스 소비자 변환 작성

재사용 가능하거나 재사용 불가능한 웹 서비스 소비자 변환을 작성할 수 있습니다. 재사용 가능 변환은 여러 매핑에 존재할 수 있습니다. 재사용 불가능 변환은 단일 매핑에만 존재합니다.

단일 WSDL 개체에서 SOAP 1.1 바인딩 및 SOAP 1.2 바인딩에 대한 웹 서비스 소비자 변환을 작성할 수 있습니다.

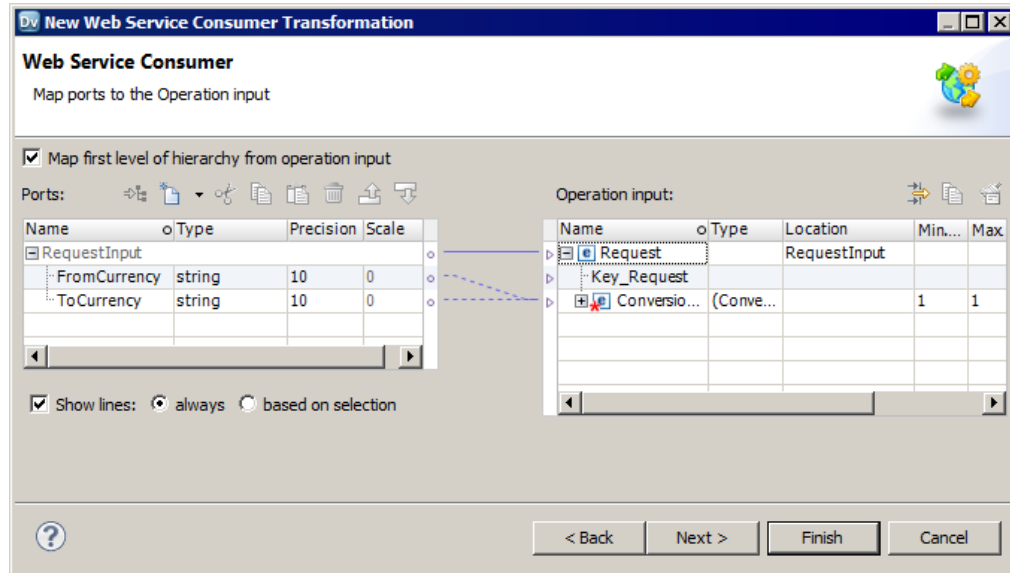
1. 변환을 작성하려면 다음 방법 중 하나를 사용합니다.

옵션	설명
재사용 가능	개체 탐색기 보기에서 프로젝트나 폴더를 선택합니다. 파일 > 새로 만들기 > 변환 을 클릭합니다. 웹 서비스 소비자 변환을 선택하고 다음 을 클릭합니다.
재사용 불가능	매핑 또는 맵렛에서 변환 색상표의 웹 서비스 소비자 변환을 편집기로 끌어서 놓습니다.

새 웹 서비스 소비자 변환 대화 상자가 나타납니다.

2. WSDL 데이터 개체를 찾아보고 선택하여 웹 서비스 요청 및 응답 메시지를 정의합니다.
WSDL이 리포지토리에 없는 경우 새 웹 서비스 소비자 변환 대화 상자에서 WSDL을 가져올 수 있습니다.
3. WSDL에서 작업을 찾아보고 선택합니다.
SOAP 1.1 바인딩 또는 SOAP 1.2 바인딩이 있는 작업을 선택할 수 있습니다.
4. **다음**을 클릭합니다.

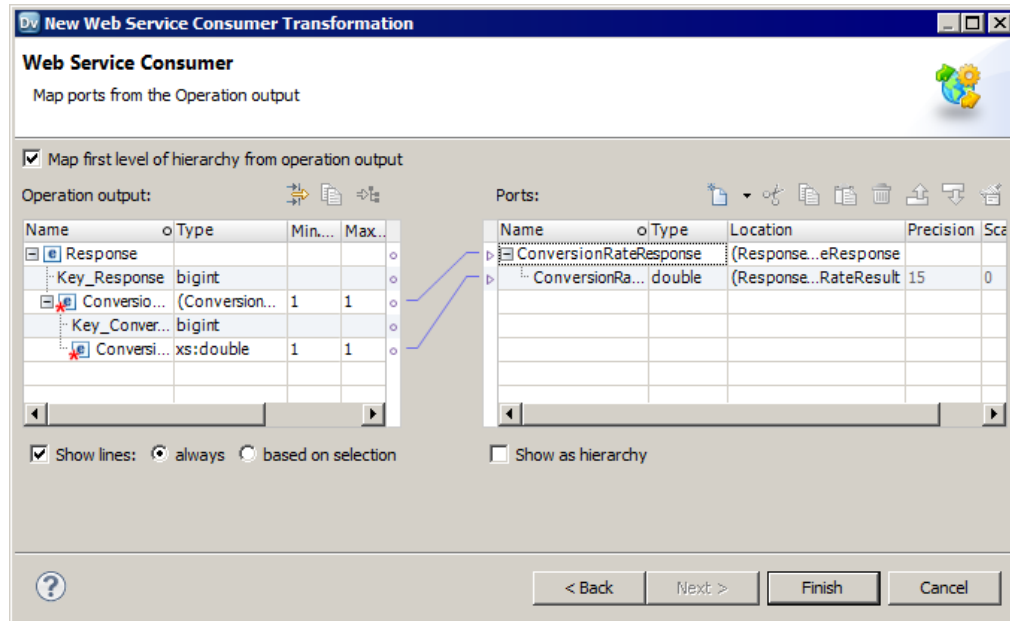
작업 입력에서 포트 매핑 화면이 나타납니다. 포트 영역에 변환 입력 그룹 및 입력 포트가 표시됩니다. 작업 입력 영역에 요청 메시지 계층이 표시됩니다.



5. 입력 그룹 및 입력 포트를 정의하고 입력 포트를 작업 입력 노드로 매핑합니다.

6. 다음을 클릭합니다.

작업 출력에서 포트 매핑 화면이 나타납니다. 작업 출력 영역에 응답 메시지 계층이 표시됩니다. 포트 영역에 변환 출력 그룹 및 출력 포트가 표시됩니다.



7. 출력 그룹 및 출력 포트를 정의하고 작업 출력 노드를 출력 포트에 매핑합니다.

8. 마침을 클릭합니다.

9. 고급 보기를 클릭하여 변환 속성 및 웹 서비스 연결을 구성합니다.

웹 서비스 소비자 변환 예제

회사에서는 RT100 제품군에 대한 주문 정보를 판매 조직에 공개해야 합니다. 판매 팀에서는 매일 주문 요약 및 주문 세부 정보를 쿼리해야 합니다.

이 경우 가상 테이블로 일일 주문 정보를 공개하는 논리적 데이터 개체를 작성합니다. 읽기 매핑에는 최신 RT100 주문을 반환하는 웹 서비스 소비자 변환을 포함합니다. 웹 서비스 소비자 변환은 RT100 제품군에 대한 일일 주문 요약 및 주문 세부 정보를 반환하는 웹 서비스를 사용합니다.

입력 파일

입력 파일은 제품 행 번호가 포함된 플랫폼 파일입니다.

실제 데이터 개체를 작성하여 입력 파일을 정의합니다. 이 파일에는 **Product_Line** 필드 1개가 포함됩니다. 필드 값은 RT100입니다. **런타임 속성** 보기에서 실제 데이터 개체의 위치를 정의합니다.

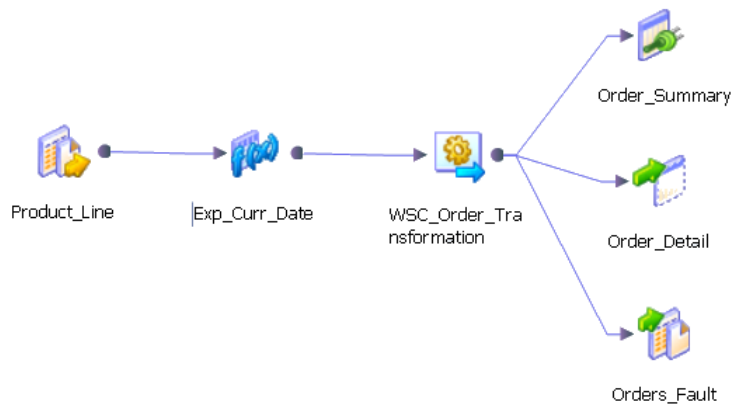
논리적 데이터 개체 모델

조직의 비즈니스 분석가는 주문 요약 및 주문 세부 테이블 구조를 설명하는 논리적 데이터 모델을 작성합니다. 논리적 데이터 모델에는 **Order_Summary** 및 **Order_Detail** 논리적 데이터 개체가 포함됩니다.

분석가는 논리적 데이터 모델을 정의하는 모델링 도구에서 스키마를 작성합니다. 사용자는 이 스키마에서 논리적 데이터 모델을 가져와 **Order_Summary** 및 **Order_Detail** 논리적 데이터 개체를 작성합니다.

논리적 데이터 개체 매핑

논리적 데이터 개체 매핑은 논리적 데이터 개체를 통해 데이터에 액세스하는 방법을 설명합니다.



읽기 매핑에는 다음 개체가 포함됩니다.

Product_Line

제품 행 번호가 포함된 입력 플랫폼 파일입니다.

Exp_Curr_Date 변환

웹 서비스 소비자 변환의 루트 수준 입력 그룹에 대한 현재 날짜 및 기본 키를 반환하는 식 변환입니다.

WSC_Order 변환

웹 서비스를 사용하여 주문 정보를 검색하는 웹 서비스 소비자 변환입니다. 변환은 제품 행과 현재 날짜를 요청 메시지로 웹 서비스에 전달합니다. 변환은 웹 서비스의 주문 정보를 응답 메시지로 수신합니다.

Order_Summary 테이블

Order_No, Customer_Id, Qty 및 Order_Date와 같은 주문 정보가 포함된 논리적 데이터 개체입니다.

Order_Detail 테이블

Order_No, Product_Id, Qty 및 Status와 같은 주문 세부 정보가 포함된 논리적 데이터 개체입니다.

Orders_Fault

일반 결함 메시지를 수신하는 출력 플랫폼 파일입니다.

웹 서비스 소비자 변환

웹 서비스 소비자 변환에서 제품군, 날짜 및 일련 번호를 입력으로 수신합니다. 변환에서 **Get_Order_Info** 웹 서비스 작업을 사용하여 주문 정보를 검색합니다.

웹 서비스 소비자 변환을 작성할 경우 요청 및 응답 웹 서비스 메시지를 설명하는 **WSDL** 데이터 개체를 선택합니다. 웹 서비스 메시지는 **XML** 요소의 계층 그룹이 포함되어 있습니다. 요소에 다른 요소가 포함될 수 있습니다. 일부 요소는 여러 번 발생합니다. 리포지토리의 **Order_Info WSDL** 개체에서 변환을 작성합니다.

변환 입력 포트를 구성하고 포트를 작업 입력 계층에 매핑합니다. 작업 출력 계층의 노드를 출력 포트에 매핑합니다. 웹 서비스 연결 및 런타임 속성을 정의합니다.

변환 입력 매핑

포트 보기에 입력 매핑을 표시할 경우 입력 포트를 정의하고 작업 입력의 노드에 매핑할 수 있습니다.

변환 **포트** 영역에는 루트 그룹과 주문 그룹이 있습니다. 루트 그룹이 요청 입력 그룹입니다. 포트 하나를 요청 입력 그룹에 추가하여 기본 키를 나타냅니다.

주문 그룹에는 **Select_Date** 및 **Select_Product_Line** 입력 포트가 있습니다.

입력 포트를 **작업 입력** 영역의 **Order_Date** 및 **Product_Line** 노드에 매핑합니다.

웹 서비스 소비자 변환이 웹 서비스에 전달하는 요청 메시지를 **작업 입력** 영역에서 정의합니다. 노드는 기본적으로 **작업 입력** 영역에 표시됩니다.

변환 출력 매핑

포트 보기에서 출력 매핑을 표시할 경우 작업 출력의 노드를 변환 출력 그룹에 매핑하여 출력 포트를 정의할 수 있습니다.

웹 서비스가 웹 서비스 응답 메시지를 통해 다음 계층을 반환합니다.

```
Response
  Orders
    Order
      Key_Order
      Order_ID
      Order_Date
      Customer_ID
      Total_Qty
      Order_Details
        Order_Detail
          Product_ID
          Description
          Qty
          Status
```

웹 서비스에서 여러 주문을 반환합니다. 주문은 주문 수준에서 다중 발생 노드입니다. 각 주문마다 웹 서비스가 여러 주문 세부 정보를 반환할 수 있습니다. **Order_Detail**은 **Order_Details** 수준에서 다중 발생 노드입니다.

참고: Developer tool이 사용자 인터페이스에서 **Key_Order** 노드를 추가합니다. 키를 출력 그룹에 매핑하여 그룹 간 관계를 정의할 수 있습니다. 이 예제에서는 **Order_ID**가 **Order**에서 기본 키이고 **Order_Details**에서 외래 키입니다.

포트 영역에서 다음 출력 그룹을 작성합니다.

```
Order
  Order_ID
  Order_Date
  Customer_ID
  Total_Qty

Order_Detail
  Order_ID
  Product_ID
  Description
  Qty
  Status
```

Order_ID 값이 변경될 때마다 데이터 통합 서비스가 **Order** 그룹의 행을 기록합니다.

Order_ID 및 **Product_ID**의 값이 변경될 때마다 데이터 통합 서비스가 **Order_Detail** 그룹의 행을 기록합니다.

변환 고급 속성

웹 서비스 소비자 변환에 대한 다음 고급 속성을 구성합니다.

일반 SOAP 결함 처리 활성화

SOAP 결함 메시지를 수신하는 출력 포트를 추가합니다.

연결

웹 서비스 액세스를 위한 웹 서비스 연결을 선택합니다.

압축 활성화

웹 서비스 소비자 변환이 웹 메시지를 **GZIP**로 압축합니다.

제 10 장

REST 웹 서비스

이 장에 포함된 항목:

- [REST 웹 서비스 개요, 103](#)
- [REST 웹 서비스 프로세스, 104](#)
- [웹 서비스 소비자 변환 프로세스, 104](#)
- [REST 웹 서비스 리소스, 105](#)
- [리소스 매핑, 108](#)
- [REST 웹 서비스 출력 변환, 109](#)
- [요청 메시지, 111](#)
- [응답 메시지 형식, 113](#)
- [응답 데이터 미리보기, 114](#)

REST 웹 서비스 개요

Informatica REST 웹 서비스는 데이터에 대한 HTTP 요청을 처리하고 JSON 파일 또는 XML 파일인 응답을 반환합니다.

외부 응용 프로그램, 웹 브라우저 또는 REST 웹 서비스 소비자 변환이 REST 웹 서비스에 연결하여 요청을 보낼 수 있습니다. REST 웹 서비스는 요청을 처리하여 클라이언트에 응답을 다시 보냅니다.

예를 들어 웹 서비스 작업을 실행하는 요청을 웹 서비스 클라이언트가 보냅니다. 웹 서비스 클라이언트가 요청에서 고객 ID를 전달합니다. 웹 서비스가 주문 테이블에서 고객 및 주문 정보를 검색합니다. 웹 서비스가 정보를 JSON 파일로 클라이언트에 반환합니다.

Informatica REST 웹 서비스는 **Developer tool**에서 데이터 서비스를 정의하여 생성합니다. 데이터 개체를 REST 웹 서비스로 배포할 수도 있습니다.

Informatica REST 웹 서비스에는 다음과 같은 구성 요소가 있습니다.

리소스

리소스에는 REST 웹 서비스가 실행하는 매핑 및 웹 서비스가 반환하는 응답 메시지의 정의가 포함됩니다. 리소스에는 출력 데이터의 키 필드인 리소스 ID도 포함됩니다. 데이터 개체에서 리소스를 생성하거나 리소스를 수동으로 정의할 수 있습니다. 개체를 REST 웹 서비스로 배포할 경우 **Developer tool**이 리소스를 생성합니다. 웹 서비스는 리소스를 여러 개 가질 수 있습니다.

요청 메시지

웹 서비스 클라이언트가 태스크 수행을 위해 웹 서비스에 보내는 요청입니다. Informatica 웹 서비스는 HTTP GET 메서드를 수행할 수 있습니다. 요청 메시지는 웹 서비스의 이름, 태스크를 수행할 리소스의 이름과 네트워크 위치, 그리고 출력을 필터링하는 매개 변수가 포함된 문자열입니다.

리소스 매핑

웹 서비스 클라이언트에 반환할 데이터를 반환하는 매핑입니다. 기본 리소스 매핑을 생성하거나 사용자 지정 리소스 매핑을 생성할 수 있습니다. 기본 리소스 매핑에는 동일한 포트를 가진 읽기 변환과 출력 변환이 포함됩니다. 사용자 지정 매핑에는 읽기 변환 및 출력 변환과 함께 다른 변환이 포함될 수 있습니다. 사용자 지정 리소스 매핑에는 출력 변환과 다른 포트를 사용하는 읽기 변환이 포함될 수 있습니다.

응답 메시지

웹 서비스 클라이언트에 반환할 데이터가 포함된 JSON 또는 XML 파일입니다. 응답 메시지에는 여러 번 발생하는 데이터 및 요소의 계층이 포함될 수 있습니다.

REST 웹 서비스 프로세스

REST 웹 서비스는 외부 응용 프로그램, 웹 브라우저 또는 REST 웹 서비스 소비자 변환 같은 웹 서비스 클라이언트의 요청을 처리합니다.

다음 프로세스에서는 데이터 통합 서비스가 웹 서비스 클라이언트의 웹 서비스 요청을 처리하는 방법을 설명합니다.

1. 데이터 통합 서비스가 웹 서비스 클라이언트로부터 요청을 수신합니다.
2. 데이터 통합 서비스의 REST 웹 서비스 모듈은 데이터 개체에서 행을 검색하기 위한 리소스 매핑을 실행하여 요청을 처리합니다.
3. 요청에 필터 매개 변수가 포함된 경우 데이터 통합 서비스가 출력 행을 필터링합니다.
4. REST 웹 서비스 모듈이 응답 메시지를 웹 서비스 클라이언트에 보냅니다.

웹 서비스 소비자 변환 프로세스

외부 응용 프로그램 또는 웹 서비스 소비자 변환이 웹 서비스 클라이언트로 웹 서비스에 연결할 수 있습니다.

다음 프로세스에서는 웹 서비스 소비자 변환이 요청을 전송하고 웹 서비스에서 응답을 수신하는 방법을 설명합니다.

1. 웹 서비스 소비자 변환이 요청을 생성하고, 연결 개체를 사용하여 웹 서비스에 연결합니다.
2. 웹 서비스 소비자 변환이 웹 서비스로부터 응답을 수신합니다.
3. 웹 서비스 소비자 변환이 응답에서 데이터를 추출하고, 변환 출력 포트에서 데이터를 반환합니다.

REST 웹 서비스 리소스

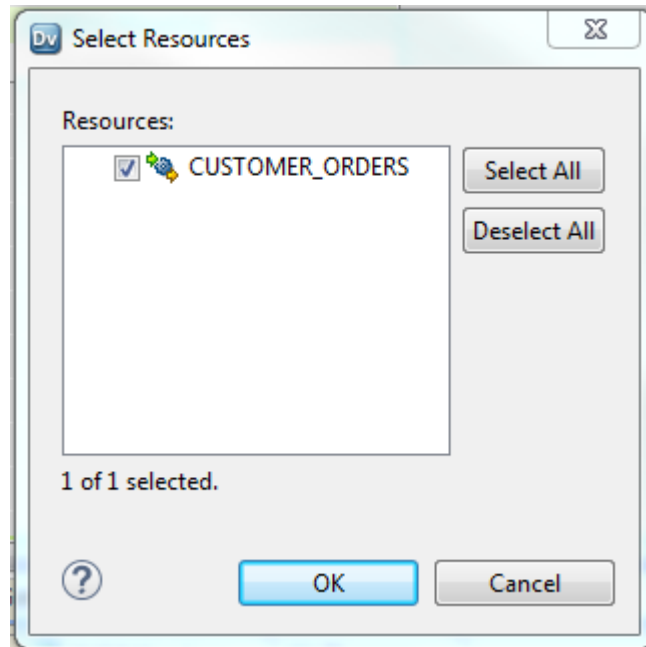
리소스에는 REST 응답 메시지에 대한 정의 및 응답을 반환할 데이터 액세스 방법이 포함됩니다. Informatica REST 웹 서비스를 생성할 때 리소스를 수동으로 정의하거나, 데이터 개체에서 리소스를 생성할 수 있습니다.

리소스를 관계형 데이터 개체 또는 플랫폼 파일 데이터 개체에서 생성할 수 있습니다. 데이터 개체에서 리소스를 생성할 경우, **Developer tool**은 읽기 변환 및 출력 변환이 포함된 기본 리소스 매핑을 생성합니다. 출력 변환에는 읽기 변환과 동일한 포트가 포함됩니다.

출력 응답 메시지의 구조를 수동으로 정의할 수 있습니다. 리소스를 수동으로 생성할 때는 응답 메시지에 요소를 정의합니다. 그런 다음 응답 메시지에 데이터를 반환하는 매핑을 정의합니다.

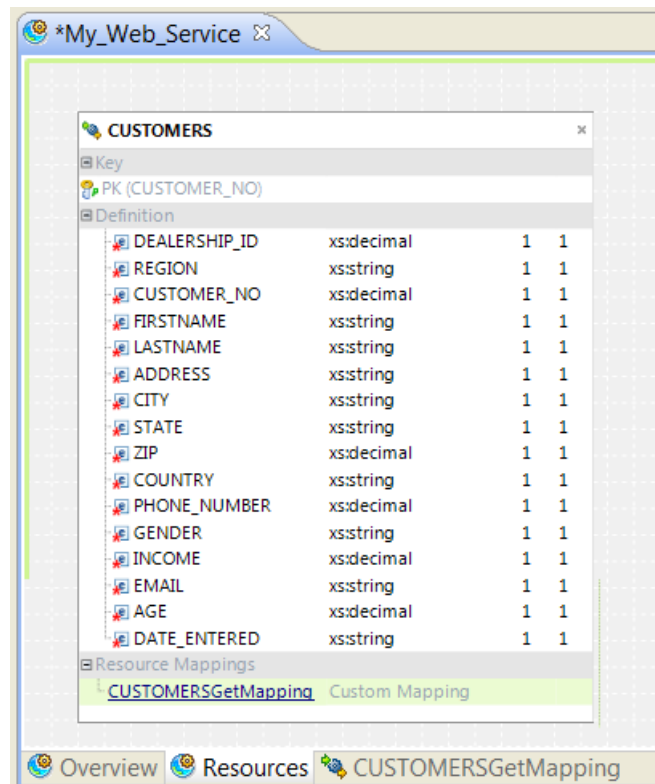
REST 웹 서비스에는 리소스가 여러 개 있을 수 있습니다. 각 리소스에는 데이터를 검색하는 리소스 매핑과 출력 응답에 대한 정의가 있습니다. 리소스 탭에서 보려는 리소스를 선택할 수 있습니다. 편집기에서 마우스 오른쪽 단추를 클릭하고 **리소스 표시**를 선택합니다.

다음 이미지는 리소스 선택 대화 상자를 보여 줍니다.



리소스 탭에서 리소스의 구성 요소를 볼 수 있습니다.

다음 이미지는 REST 웹 서비스의 **리소스** 탭에 있는 리소스 구성 요소를 보여 줍니다.



리소스에는 다음과 같은 구성 요소가 포함됩니다.

키

응답 메시지에 있는 데이터의 인덱스입니다. 웹 서비스 클라이언트는 특정 키에 대한 데이터를 요청할 수 있습니다. 모든 단순 유형 요소를 출력의 키로 설정할 수 있습니다.

정의

출력 응답 메시지의 요소입니다. 리소스에서 정의를 확장하거나, **리소스** 속성의 **스키마** 보기로 이동하여 요소를 볼 수 있습니다.

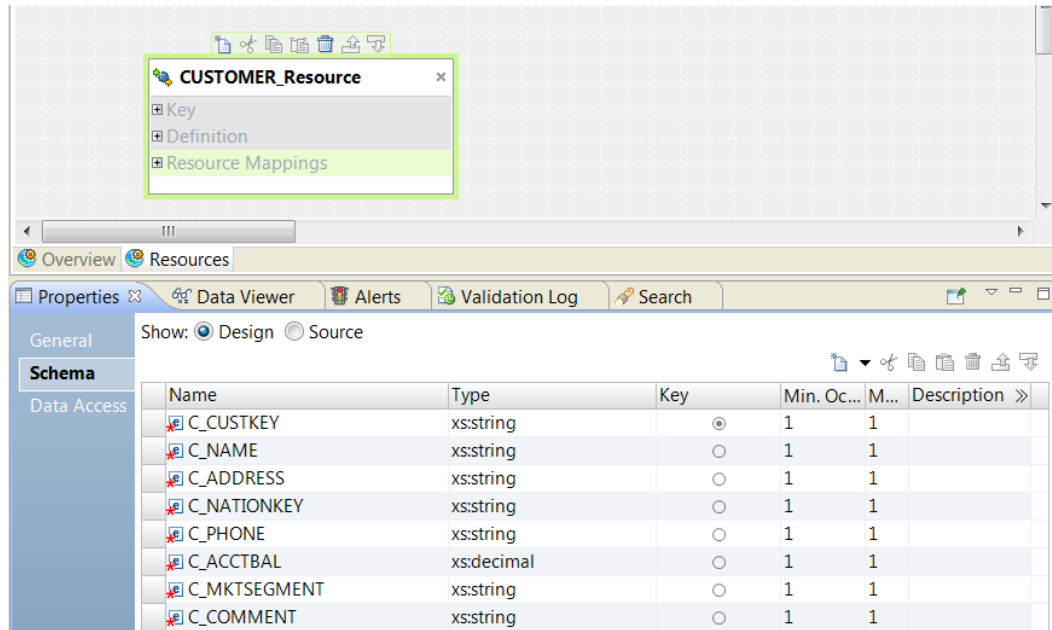
리소스 매핑

데이터를 검색하여 변환한 후 응답 메시지에 반환하는 매핑입니다. 기본적으로 리소스 매핑에는 읽기 변환과 출력 변환이 포함됩니다. 원하는 변환을 매핑에 추가할 수 있습니다. 리소스 매핑은 리소스에 있는 링크를 클릭하여 볼 수 있습니다.

REST 웹 서비스 스키마 보기

리소스 속성의 스키마 보기에서는 REST 리소스 정의의 구조를 보거나 수정할 수 있습니다.

다음 이미지는 스키마 보기를 보여 줍니다.



스키마의 디자인 또는 소스 보기를 선택할 수 있습니다. 스키마를 XML 형식으로 보려면 소스 보기를 선택합니다.

스키마의 요소 또는 요소 유형을 변경하려면 디자인 보기를 사용합니다. 요소의 순서를 변경할 수 있습니다. 디자인 보기에서 키를 변경할 수 있습니다. 여러 번 발생하는 필드를 정의할 수도 있습니다. 필드가 발생해야 하는 최소 및 최대 수를 입력합니다. **바운드되지 않음**을 선택하면 제한 없이 여러 번 발생할 수 있는 필드를 생성할 수 있습니다.

스키마의 형식을 변경하면 변경 내용이 출력 변환에 나타납니다. 필요한 경우 출력 변환의 포트를 업데이트해야 할 수 있습니다.

데이터 개체 동기화

리소스에 기본 매핑이 포함되어 있으면 Developer tool은 리소스 정의를 데이터 개체와 동기화할 수 있습니다.

리소스 키가 데이터 개체의 기본 키와 다르면 Developer tool이 리소스 정의를 데이터 개체와 동기화할 수 없습니다.

리소스 키

리소스 정의에 키를 정의할 수 있습니다. 키를 정의하면 데이터 통합 서비스가 출력 데이터의 열을 인덱싱합니다. 웹 서비스 클라이언트는 키를 기준으로 특정 출력 행을 요청할 수 있습니다.

데이터 개체에서 리소스를 생성할 경우 기본적으로 Developer tool은 데이터 개체의 기본 키를 리소스 키로 사용합니다. 기본 키에 여러 행이 포함되어 있으면 Developer tool은 열 데이터를 "+"로 구분하여 기본 키를 생성합니다.

소스의 기본 키 대신 다른 출력 열을 기본 키로 사용하도록 Developer tool을 구성할 수 있습니다.

리소스를 수동으로 생성할 경우 리소스 키는 단순 유형의 요소여야 합니다.

리소스 매핑

리소스 매핑은 데이터 개체를 읽은 후 데이터를 **REST** 응답 메시지에 반환하는 매핑입니다. **Informatica** 리소스 매핑은 **GET** 메서드를 수행합니다. 매핑은 웹 서비스 요청 메시지를 구문 분석하지 않습니다.

리소스 매핑에는 클라이언트 요청에 대해 데이터를 필터링하는 필터 변환 또는 조희 변환이 포함되지 않습니다. 웹 서비스 요청 메시지에 필터 쿼리가 포함된 경우, 매핑이 데이터를 검색한 후에 데이터 통합 서비스가 해당 데이터를 필터링합니다. **Administrator** 도구에서 데이터 통합 서비스에 대해 결과 집합 캐싱 속성이 활성화되어 있으면 데이터 통합 서비스가 응답 메시지의 데이터를 캐싱합니다. 웹 서비스 요청에 필터 매개 변수가 포함되어 있으면 데이터 통합 서비스는 캐시의 행을 필터링합니다.

리소스 매핑을 정의할 때 다음 리소스 매핑 유형 중 하나를 생성합니다.

기본 리소스 매핑

기본 리소스 매핑에는 읽기 변환과 출력 변환이 포함됩니다. 다른 변환은 매핑에 포함되지 않습니다. 매핑은 데이터 개체의 모든 행을 변경하지 않고 반환합니다. 기본 리소스 매핑을 생성하려면 데이터 개체에서 웹 서비스를 생성하거나 데이터 개체를 웹 서비스로 배포합니다.

사용자 지정 리소스 매핑

사용자 지정 리소스 매핑은 기본 매핑이 아닌 모든 매핑입니다. 사용자 지정 리소스 매핑에는 출력 변환과 다른 열을 사용하는 읽기 변환이 포함될 수 있습니다. 사용자 지정 리소스 매핑의 경우 읽기 변환과 출력 변환 사이의 변환이 매핑에 포함될 수 있습니다. 변환은 파이프라인의 열을 변경하거나 열을 추가할 수 있습니다.

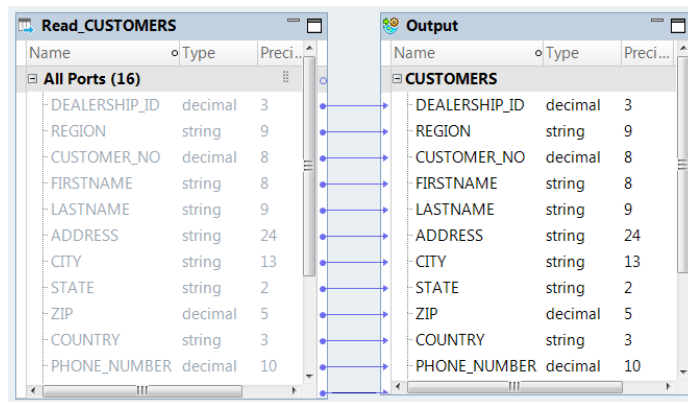
예를 들어 열 두 개를 결합하여 세 번째 열을 반환하는 식 변환이 리소스 매핑에 포함될 수 있습니다. 또는 여러 번 발생하는 행을 반환하는 변환이 리소스 매핑에 포함될 수 있습니다. 사용자 지정 리소스 매핑은 리소스를 수동으로 생성할 때 생성할 수 있습니다. 읽기 변환을 변경하거나 매핑에 변환을 더 추가하는 방법으로 기본 매핑을 변경할 수도 있습니다.

기본 리소스 매핑

기본 리소스 매핑에는 읽기 변환과 출력 변환이 포함됩니다. 데이터 개체에서 웹 서비스를 생성하고 변경하지 않으려는 경우에 기본 리소스 매핑을 생성합니다.

데이터 개체에서 리소스를 생성할 경우, 마법사는 읽기 변환 및 출력 변환이 포함된 기본 리소스 매핑을 생성합니다. 읽기 변환은 리소스를 생성하는 데 사용된 데이터 개체를 읽습니다. 기본 리소스 매핑에서 출력 변환에는 읽기 변환과 동일한 열이 포함됩니다.

다음 이미지는 고객 데이터 개체의 모든 열을 반환하는 기본 리소스 매핑을 보여 줍니다.



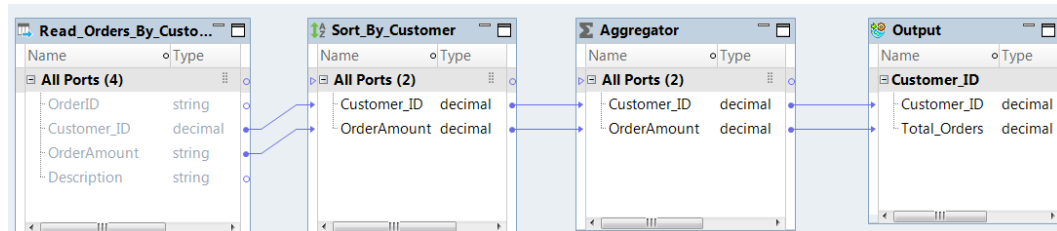
리소스 매핑에는 데이터를 필터링하는 조회 변환 또는 필터 변환이 포함되지 않습니다. 하지만 REST 웹 서비스 클라이언트가 필터 매개 변수가 포함된 요청을 기본 리소스 매핑이 포함된 웹 서비스에 보낼 수 있습니다. 데이터 통합 서비스는 리소스 매핑이 생성하는 출력 데이터에서 데이터를 필터링합니다.

사용자 지정 리소스 매핑

사용자 지정 리소스 매핑에는 출력 변환과는 다른 포트를 사용하는 읽기 변환이 포함될 수 있습니다. 사용자 지정 리소스 매핑에는 읽기 변환 및 출력 변환보다 더 많은 변환이 포함될 수 있습니다.

기본 리소스 매핑을 수정하거나 수동으로 REST 웹 서비스를 생성하여 사용자 지정 리소스 매핑을 생성할 수 있습니다.

다음 이미지는 사용자 지정 리소스 매핑을 보여 줍니다.



매핑에는 다음과 같은 변환이 포함됩니다.

읽기 변환

주문 파일을 읽습니다. 각 주문에는 고객 ID가 포함됩니다. 고객 ID는 여러 번 발생할 수 있습니다.

분류기 변환

고객 ID별로 주문을 정렬합니다.

집계 변환

각 고객의 주문 총액을 더합니다.

출력 변환

고객별 주문 총액을 반환합니다.

이 사용자 지정 리소스 매핑을 생성하려면 REST 웹 서비스 리소스 정의를 수동으로 정의합니다. 리소스 정의를 수동으로 정의할 때 응답 메시지에 요소를 정의합니다. 이 예제에서는 응답 메시지에 고객 ID와 주문 총액만 포함됩니다.

리소스 정의를 정의하면 출력 변환이 포함된 리소스 매핑을 Developer tool이 생성합니다. 그런 다음 읽기 변환과 기타 변환을 매핑에 추가합니다.

이전 이미지의 고객 ID가 키입니다. 웹 서비스 클라이언트가 특정 고객의 주문 수를 요청할 수 있습니다. 데이터 통합 서비스는 키를 기준으로 출력 데이터를 필터링합니다. 매핑에는 필터 변환이 포함되지 않습니다.

REST 웹 서비스 출력 변환

출력 변환은 작업 매핑의 관계형 데이터 그룹에서 REST 웹 서비스 응답 메시지를 생성합니다. Developer tool은 사용자가 REST 웹 서비스를 정의할 때 출력 변환을 생성합니다.

REST 웹 서비스를 생성하면 Developer tool은 사용자가 정의한 리소스 정의를 기반으로 출력 변환을 생성합니다. Developer tool은 응답 메시지 구조를 기반으로 변환의 입력 포트를 생성합니다. 출력 변환에는 입력 포트를 응답 메시지의 노드에 매핑하는 매핑이 포함됩니다.

스키마를 변경하여 출력 메시지 계층의 요소를 변경할 수 있습니다. 스키마 변경 내용에 맞게 변환의 입력 포트를 변경할 수 있습니다.

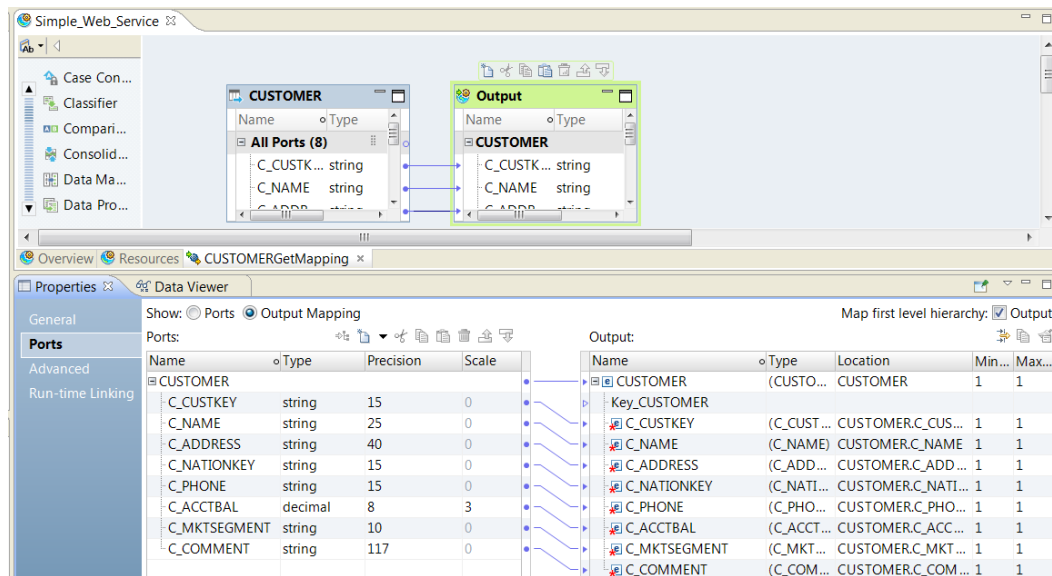
REST 응답 메시지 계층은 출력 변환 **포트** 탭에서 볼 수 있습니다. 변환 포트를 보거나, 변환 입력 포트와 응답 메시지 계층 사이의 매핑을 보도록 선택할 수 있습니다.

포트를 표시할 때 수동으로 그룹 및 포트를 추가하거나 다른 변환의 포트를 출력 변환에 복사할 수 있습니다. 키보드 바로 가기를 사용하거나 **Developer tool**에서 복사 및 붙여넣기 단추를 사용할 수 있습니다.

출력 매핑을 표시할 경우 입력 그룹 및 입력 포트를 정의하고 입력 포트를 응답 메시지 요소에 매핑할 수 있습니다. 탭의 왼쪽에는 **포트** 영역이 있고, 오른쪽에는 **출력** 영역이 있습니다. **출력** 영역에 응답 메시지 계층이 표시됩니다. **포트** 영역에서 입력 그룹과 포트를 정의합니다. **포트** 영역의 입력 포트를 **출력** 영역의 노드에 매핑하면 **출력** 영역의 **위치** 열에 입력 포트 위치가 나타납니다.

출력 매핑을 표시할 때 입력 포트를 작업 입력의 노드에 연결하는 선을 표시하도록 선택할 수 있습니다.

다음 이미지는 출력 변환의 단순 출력 매핑을 보여 줍니다.



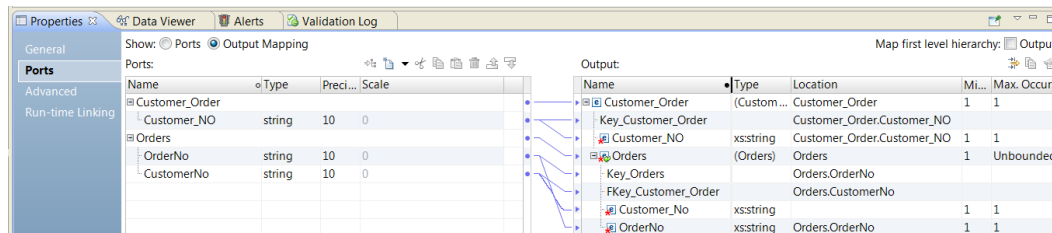
이전 이미지에서 변환은 고객에 대해 포트 그룹 하나를 수신합니다. 그리고 한 번 발생하는 요소를 반환합니다.

계층의 첫 번째 수준을 매핑하기로 선택한 경우 **Developer tool**은 입력 포트를 작업 출력의 첫 번째 수준에 있는 노드에 매핑합니다. **첫 번째 수준 계층 매핑** 옵션은 기본적으로 활성화됩니다. **Developer tool**은 데이터를 매핑할 때 필요한 입력 포트도 생성합니다. 계층의 첫 번째 수준에 하나 이상의 다중 발생 하위 노드가 있는 다중 발생 상위 노드가 포함된 경우 **Developer tool**은 포트를 생성하지 않거나 계층의 첫 번째 수준을 매핑하지 않습니다.

REST 출력 변환에서 여러 번 발생하는 데이터

REST 출력 변환은 여러 데이터 그룹을 수신하여 여러 수준의 출력 데이터를 응답 메시지에 반환할 수 있습니다. 응답 메시지의 각 데이터 그룹은 키로 연결됩니다.

다음 이미지는 고객의 여러 주문을 반환하는 출력 매핑을 보여 줍니다.



출력 매핑의 **포트** 영역에는 두 개의 그룹 **Customer_Order** 및 **Orders**가 있습니다. 각 그룹에는 고객 번호가 포함됩니다. 고객 번호는 응답 메시지에서 고객 데이터와 주문 데이터를 연결하는 키로 사용됩니다.

리소스 정의의 기본 키는 **Customer_No**입니다. **Customer_No**는 **Customer_Order**의 키입니다. **Orders**에서는 외래 키입니다. 기본 키는 고유한 값을 포함해야 합니다. 데이터 통합 서비스는 여러 번 발생하는 주문을 처리할 경우, 키를 기반으로 고객의 모든 주문을 반환할 수 있습니다. 응답 메시지에는 기본 키와 외래 키가 나타나지 않습니다.

출력 매핑의 **출력** 영역에는 응답 메시지가 포함됩니다. 입력 포트를 출력의 각 키에 연결해야 합니다.

참고: REST 출력 변환이 중복 기본 키 값을 수신하면 오류가 발생하여 미리보기가 실패합니다.

요청 메시지

웹 서비스 클라이언트는 요청 메시지를 **URI** 문자열로 웹 서비스에 보냅니다. **URI**는 호스트 포트, 웹 서비스 이름 및 웹 서비스에서 액세스할 리소스를 식별합니다. 요청 메시지는 웹 서비스 출력에서 특정 행을 필터링하는 쿼리 매개 변수를 포함할 수 있습니다.

URI에 식별자나 매개 변수가 포함되어 있지 않으면 **REST** 웹 서비스는 데이터 개체의 모든 행이 포함된 목록을 **REST** 웹 서비스 응답에 반환합니다. 웹 서비스는 모든 고객 행이 포함된 **JSON** 파일 또는 **XML** 파일을 반환합니다.

다음 예제에서는 모든 고객의 목록을 반환하는 요청을 보여 줍니다.

`http://myhost:8095/DataIntegrationService/RESTSERVICE/Rsrc_CUSTOMER/CUSTOMER`

다음 텍스트는 **JSON** 응답 메시지를 보여 줍니다.

```
{
  "CUSTOMERS": {
    "CUSTOMER": [
      {
        "C_ACCTBAL": 9331.13,
        "C_ADDRESS": "38 Summit Drive",
        "C_COMMENT": "Call immediately if delay",
        "C_CUSTKEY": {
          "@url": "http://myhost:8095/DataIntegrationService/RESTSERVICE/Rsrc_CUSTOMER/CUSTOMER/63",
          "$": 63
        },
        "C_MKTSEGMENT": "AUTOMOBILE",
        "C_NAME": "Customer#000000063",
        "C_NATIONKEY": 21,
        "C_PHONE": "31-952-552-9584"
      },
      {
        "C_ACCTBAL": -646.64,
        "C_ADDRESS": "44 Ocean Avenue",
        "C_COMMENT": "Has dangerous animal in the house",
        "C_CUSTKEY": {
          "@url": "http://myhost:8095/DataIntegrationService/RESTSERVICE/Rsrc_CUSTOMER/CUSTOMER/64",
          "$": 64
        },
        "C_MKTSEGMENT": "BUILDING",
        "C_NAME": "Customer#000000064",
        "C_NATIONKEY": 3,
        "C_PHONE": "13-558-731-7204"
      }
    ]
  }
}
```

리소스 매핑의 데이터 필터링

요청 메시지에 쿼리 매개 변수가 포함된 경우, 데이터 통합 서비스는 쿼리 매개 변수 값을 사용하여 출력 데이터를 필터링합니다.

데이터 통합 서비스는 기본 리소스 매핑 및 사용자 지정 리소스 매핑의 출력 행을 필터링할 수 있습니다.

요청 메시지 쿼리에는 다음의 조건을 포함할 수 있습니다.

<> = != >= <=

URI에 매개 변수를 구성하려면 다음 절을 포함합니다.

?filtercondition=<열 이름> <피연산자><값>

예를 들어 다음 요청 메시지는 "BUILDING" 시장 세그먼트 값을 가진 모든 고객 행을 검색합니다.

```
http://myServer:8095/DataIntegrationService/RESTSERVICE/REST_Web_Service/CUSTOMER/?
filterCondition=C_MKTSEGMENT='BUILDING'
```

열 데이터 유형이 문자열인 경우에는 검색 값을 작은따옴표(")로 묶습니다. 열 데이터 유형이 숫자인 경우에는 검색 값을 따옴표로 묶지 않습니다.

예를 들어 다음 요청에는 숫자인 검색 매개 변수가 포함됩니다.

```
http://myServer:8095/DataIntegrationService/RESTSERVICE/REST_Web_Service/CUSTOMER/?
filterCondition=C_ACCTBAL=9331.13
```

REST 웹 서비스는 잔고가 9331.13인 고객을 반환합니다.

다음 텍스트는 XML 형식의 REST 응답 메시지를 보여 줍니다.

```
<CUSTOMERS>
<tns:CUSTOMER xmlns:tns="http://www.informatica.com">
<tns:C_CUSTKEY url="http://myServer:8095/DataIntegrationService/RestService/REST_Web_Service/CUSTOMER/
63">63</tns:C_CUSTKEY>
<tns:C_NAME>Customer#000000063</tns:C_NAME>
<tns:C_ADDRESS>IXRSpVWZraKII</tns:C_ADDRESS>
<tns:C_NATIONKEY>21</tns:C_NATIONKEY>
<tns:C_PHONE>31-952-552-9584</tns:C_PHONE>
<tns:C_ACCTBAL>9331.13</tns:C_ACCTBAL>
<tns:C_MKTSEGMENT>AUTOMOBILE</tns:C_MKTSEGMENT>
<tns:C_COMMENT>Apply discount</tns:C_COMMENT>
</tns:CUSTOMER>
</CUSTOMERS>
```

쿼리에 포함할 매개 변수가 여러 개인 경우에는 'AND'를 사용하여 매개 변수를 조인합니다.

```
http://usw1mj02ee4j:8095/DataIntegrationService/RESTSERVICE/REST_Web_Service0/CUSTOMER/?
filterCondition=C_ACCTBAL=9331.13 AND C_NATIONKEY='21'
```

열 데이터 유형이 날짜/시간인 경우에는 URI의 매개 변수 문자열을 날짜/시간 형식으로 변환하고 매개 변수 문자열의 형식을 지정할 수 있습니다.

예를 들어 다음 문은 매개 변수 문자열을 날짜/시간 형식으로 변환합니다.

?filterCondition= O_ORDERDATE=TO_DATE('1994-11-17 00:00:00.000000000','YYYY-MM-DD HH24:MI:SS.NS')

키로 검색

리소스 키에 대해 REST 웹 서비스 쿼리를 구성할 수 있습니다. 요청에 키 값이 포함된 경우에는 쿼리에서 열 이름을 참조하지 않아도 됩니다.

CUSTOMER라는 리소스의 리소스 키에서 숫자 64를 검색하려면 다음 형식을 사용합니다.

```
http://myServer:8095/DataIntegrationService/RESTSERVICE/REST_Web_Service/CUSTOMER/64
```


URI에는 리소스 이름과 리소스 키 값이 포함되어 있습니다.

/CUSTOMER/64

키로 사용되는 열의 이름을 쿼리에서 참조할 필요가 없습니다.

리소스 키를 기준으로 쿼리할 경우, 웹 서비스는 키와 연결된 URI로 고객을 검색할 수 있습니다. 응답 메시지 키 값에서 특정 행의 URI를 볼 수 있습니다. 다음 예제에서 키는 C_CUSTKEY이고 키 값은 64입니다.

```
<tns:CUSTOMER xmlns:tns="http://www.informatica.com">
  <tns:C_CUSTKEY url="http://my Server:8095/DataIntegrationService/RestService/REST_Web_Service/CUSTOMER/64">64</tns:C_CUSTKEY>
  <tns:C_NAME>Customer#000000064</tns:C_NAME>
  <tns:C_ADDRESS>MbCeGY20kakK3oa1JD,0T</tns:C_ADDRESS>
  <tns:C_NATIONKEY>3</tns:C_NATIONKEY>
  <tns:C_PHONE>13-558-731-7204</tns:C_PHONE>
  <tns:C_ACCTBAL>-646.64</tns:C_ACCTBAL>
  <tns:C_MKTSEGMENT>BUILDING</tns:C_MKTSEGMENT>
  <tns:C_COMMENT>
    Customer has an angry dog in the yard
  </tns:C_COMMENT>
</tns:CUSTOMER>
```

응답 메시지 형식

JSON 또는 XML 형식으로 응답 메시지를 반환하도록 REST 웹 서비스를 구성할 수 있습니다.

응답 메시지에서 C_CUSTKEY 필드가 리소스 키입니다. C_CUSTKEY 필드에는 키 값 및 응용 프로그램이 출력 데이터의 특정 키를 통해 고객에 액세스하는 데 사용할 수 있는 URL이 포함됩니다.

JSON 형식의 경우 URL에 사용된 백슬래시는 \u002f로 유니코드 인코딩되어 있습니다.

다음과 같은 이유로 URL 내의 리소스 키 값에는 인코딩된 값이 포함될 수도 있습니다.

- 키 값에 URL 주소(예: http://www.informatica.com)가 포함됩니다.
- 키 값에 퍼센트 기호(%) (예: 20%)가 포함됩니다.
- 키 값에 공백이 포함됩니다.

다음 텍스트는 JSON 형식의 응답 메시지입니다.

```
{
  "CUSTOMERS": {
    "CUSTOMER": [
      {
        "C_ACCTBAL": 9331.13,
        "C_ADDRESS": "IXRSpVwWZraKII",
        "C_COMMENT": "Good customer",
        "C_CUSTKEY": {
          "@url": "http:\u002F\u002FHostName:8095\u002FDataIntegrationService\u002FSimple_Web_Service\u002FRsrc_CUSTOMER\u002FCUSTOMER\u002F63",
          "$": 63
        },
        "C_MKTSEGMENT": "AUTOMOBILE",
        "C_NAME": "Customer#000000063",
        "C_NATIONKEY": 21,
        "C_PHONE": "31-952-552-9584"
      }
    ]
  }
}
```

다음 텍스트는 XML 형식의 응답 메시지입니다.

```
<?xml version="1.0" encoding="UTF-16LE" standalone="no"?>
<tns:CUSTOMER xmlns:tns="http://www.informatica.com">
  <tns:C_CUSTKEY url="Simple_Web_Service/CUSTOMER/63">63</tns:C_CUSTKEY>
  <tns:C_NAME>Customer#000000063</tns:C_NAME>
```

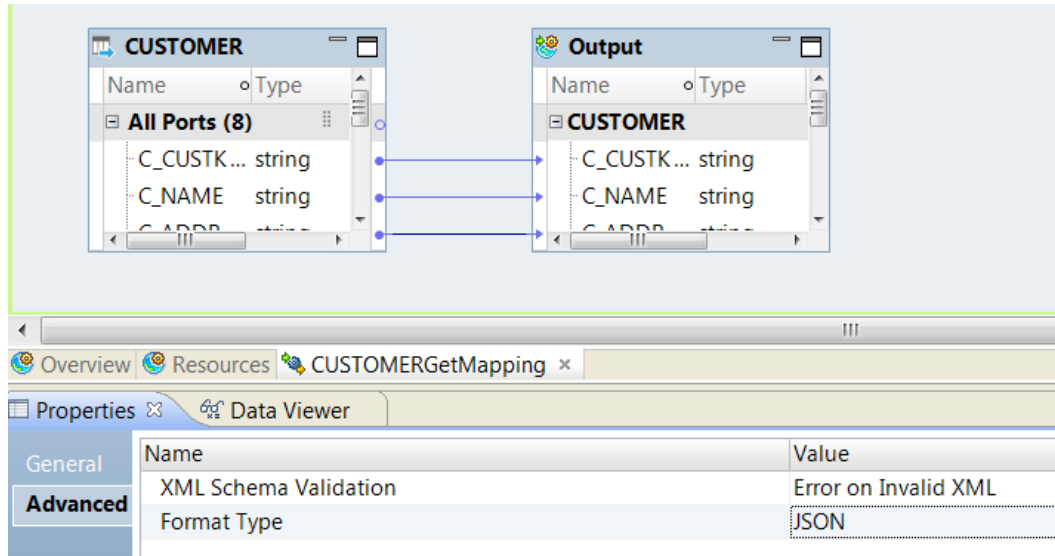
```

<tns:C_ADDRESS>IXRSpVwWZraKII                                </tns:C_ADDRESS>
<tns:C_NATIONKEY>21</tns:C_NATIONKEY>
<tns:C_PHONE>31-952-552-9584</tns:C_PHONE>
<tns:C_ACCTBAL>9331.13</tns:C_ACCTBAL>
<tns:C_MKTSEGMENT>AUTOMOBILE</tns:C_MKTSEGMENT>
<tns:C_COMMENT>Good                                          </tns:C_COMMENT>
customer
</tns:CUSTOMER>

```

응답 메시지의 형식을 설정할 수 있습니다. 응답 메시지 형식은 리소스 매핑 **속성** 보기의 **고급** 탭에서 변경합니다. **형식 유형**에 대해 JSON 또는 XML을 선택합니다.

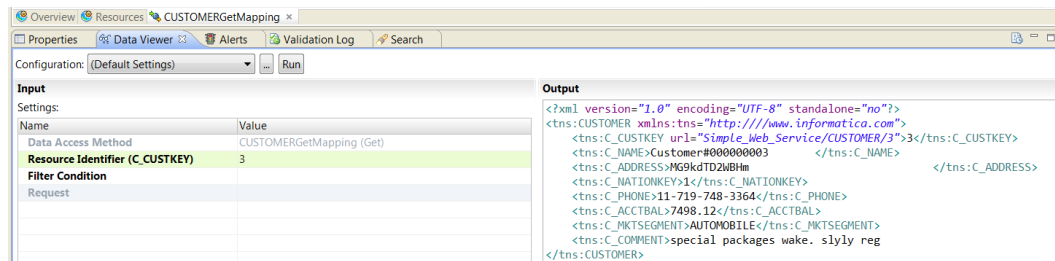
다음 이미지는 리소스 매핑 **속성** 보기의 **고급** 탭을 보여 줍니다.



응답 데이터 미리보기

Developer tool의 **데이터 뷰어** 보기에서 웹 서비스를 테스트할 수 있습니다. 리소스 키를 기준으로 데이터를 필터링하거나, 데이터를 필터링할 식을 구성할 수 있습니다.

다음 이미지는 데이터 뷰어 보기를 보여 줍니다.



다음의 입력 설정을 구성하여 **데이터 뷰어** 보기의 응답 메시지를 필터링합니다.

데이터 액세스 방법

실행할 리소스 매핑입니다. 웹 서비스는 리소스 매핑을 여러 개 포함할 수 있습니다.

리소스 식별자

검색에 사용할 키 값입니다. Developer tool은 리소스 키의 이름을 식별합니다.

필터 조건

식 편집기에서 구성할 수 있는 필터 식입니다. 식은 여러 출력 열을 참조할 수 있습니다. 출력에 계층 데이터가 포함된 경우, 필터 조건은 상위 그룹에서 요소를 참조해야 합니다. 필터 조건과 리소스 식별자를 동시에 입력할 수 있습니다.

제 11 장

REST 웹 서비스를 생성하는 방법

이 장에 포함된 항목:

- [REST 웹 서비스 생성 , 116](#)
- [REST 웹 서비스를 수동으로 생성하는 방법 , 117](#)
- [1단계. REST 웹 서비스 리소스 생성, 117](#)
- [2단계. 리소스 매핑 정의, 120](#)
- [3단계. 출력 매핑 구성, 123](#)
- [4단계. 데이터 뷰어 보기에서 매핑 테스트, 125](#)
- [5단계. 응용 프로그램 배포 , 127](#)
- [6단계. 브라우저에서 웹 서비스 쿼리, 128](#)
- [데이터 개체에서 REST 웹 서비스를 생성하는 방법, 130](#)
- [데이터 개체를 REST 웹 서비스로 배포하는 방법, 133](#)

REST 웹 서비스 생성

데이터 개체를 REST 웹 서비스로 배포하거나 마법사를 사용하여 웹 서비스를 정의할 수 있습니다.

웹 서비스를 생성하려면 먼저 응답 메시지에 포함할 요소를 결정해야 합니다. 다음의 방법 중 하나를 사용하여 응답 메시지 구조를 기반으로 웹 서비스를 생성합니다.

웹 서비스를 수동으로 생성합니다.

리소스 매핑이 계산하거나 형식을 지정해야 하는 데이터가 응답 메시지에 포함된 경우에는 웹 서비스를 수동으로 생성합니다. 매핑에 여러 소스가 있으면 웹 서비스를 수동으로 생성합니다. 응답 메시지에 계층 데이터 또는 여러 번 발생하는 데이터가 포함되어 있으면 구조를 수동으로 정의해야 합니다.

데이터 개체에서 웹 서비스를 생성합니다.

데이터 개체 하나에서 가져온 데이터가 응답 메시지에 포함된 경우에는 해당 데이터 개체를 기반으로 리소스 정의를 생성할 수 있습니다. 데이터 개체에서 특정 열을 선택할 수 있고 웹 서비스를 생성한 후 리소스 매핑을 변경할 수 있습니다.

데이터 개체를 웹 서비스로 배포합니다.

응답 메시지에 데이터 개체의 모든 열을 포함할 수 있는 경우에는 데이터 개체를 웹 서비스로 배포합니다.

REST 웹 서비스를 수동으로 생성하는 방법

REST 웹 서비스 리소스의 열을 수동으로 정의할 수 있습니다. 리소스를 수동으로 정의할 경우 **Developer tool**은 응답 메시지를 정의하는 출력 변환이 포함된 리소스 매핑을 생성합니다. **Developer tool**은 리소스 매핑에 읽기 변환은 생성하지 않습니다. 리소스를 정의한 후 매핑에 읽기 변환을 추가해야 합니다. 응답 메시지 구조가 소스 데이터와 다른 경우에는 리소스를 수동으로 생성합니다.

다음 단계를 사용하여 REST 웹 서비스를 생성합니다.

1. REST 웹 서비스 리소스를 생성합니다.
2. 데이터를 검색하는 리소스 매핑을 정의합니다.
3. 입력 포트의 데이터를 출력 계층의 요소에 매핑하는 REST 출력 변환 매핑을 구성합니다.
4. 매핑을 테스트합니다.
5. 응용 프로그램을 데이터 통합 서비스에 배포합니다.
6. 브라우저에서 웹 서비스에 액세스합니다.

예제 REST 웹 서비스

Hypostores에는 고객의 모든 주문을 반환하는 REST 웹 서비스가 있습니다. 영업 사원은 웹 브라우저에서 주문에 대한 요청을 웹 서비스로 보낼 수 있습니다. 요청에는 하나 이상의 고객 번호가 포함됩니다. 웹 서비스는 각 고객 이름 및 고객의 모든 주문 목록을 JSON 파일로 반환합니다.

REST 웹 서비스에는 고객 테이블을 읽어 고객 이름을 검색하는 매핑이 포함됩니다. 매핑에는 조회 변환이 포함됩니다. 조회 변환은 각 고객의 모든 주문을 주문 테이블에서 검색합니다. 매핑에는 계층 JSON 파일을 반환하는 REST 출력 변환이 포함됩니다. JSON 파일 구조의 상위 그룹에는 고객 번호와 고객 이름이 있습니다. JSON 파일에는 고객 내의 주문에 대한 하위 그룹이 포함됩니다. 그룹에는 여러 번 발생하는 주문이 포함됩니다. 각 주문에는 주문 번호, 가격 및 주문 날짜가 있습니다.

1단계. REST 웹 서비스 리소스 생성

REST 웹 서비스 리소스에는 REST 웹 서비스 응답 메시지에 대한 정의 및 응답을 반환할 데이터 액세스 방법이 포함됩니다. Informatica REST 웹 서비스를 생성할 경우 데이터 개체에서 리소스를 정의하거나 리소스를 수동으로 정의할 수 있습니다.

이 예제에서는 응답 메시지 형식이 계층적이고 여러 번 발생하는 데이터를 포함하기 때문에 리소스를 수동으로 생성합니다. 리포지토리에 있는 데이터 개체에서는 리소스를 생성할 수 없습니다.

REST 웹 서비스 리소스 생성

웹 서비스 리소스를 생성할 때 응답 메시지의 구조를 정의합니다.

1. Developer tool에서 **파일 > 새로 만들기 > 데이터 서비스**를 클릭합니다.
2. **REST 웹 서비스**를 선택하고 **다음**을 클릭합니다.
3. 웹 서비스 이름으로 **Orders_Web_Service**를 입력합니다. **다음**을 클릭합니다.
4. **REST 리소스** 대화 상자에서 **빈 항목에서 생성**을 클릭합니다.

기본 리소스가 나타납니다.

REST Resource
Create REST resources and define mappings on REST resources.

Name: type filter text
Name: Resource
Description:

Definition:

Name	Type	Key	Min. Occurs	Max. Occurs	Description

HTTP method(s):
☒ Get

< Back Next > **Finish** Cancel

- 이름 필드에서 리소스 이름을 Resource에서 Orders_Resource로 변경합니다.
- 정의 패널에서 새로 만들기 > 요소를 클릭합니다.
- 다음 요소를 입력합니다.

이름	유형	키	최소 발생	최대 발생
Customer_Key	문자열	예	1	1
Customer_Name	문자열		1	1
Orders	문자열		1	바운드되지 않음

- Orders 요소를 선택하고 새로 만들기 > 하위 항목을 클릭합니다.
- Orders 아래에 다음 하위 요소를 입력합니다.

이름	유형	키	최소 발생	최대 발생
Order_Key	정수		1	1
Order_Price	소수		1	1
Order_Date	정수		1	1

- Get HTTP 메시드가 활성화되었는지 확인합니다.

11. 마침을 클릭합니다.

다음 이미지는 Orders_Resource의 요소를 보여 줍니다.

REST Resource
Create REST resources and define mappings on REST resources.

Name: type filter text
Orders_Resource

Name: Orders_Resource
Description: Returns orders by customer

Definition:

Name	Type	Key	Min...	Max. Occurs	Description
Customer_Key	xs:string	<input checked="" type="radio"/>	1	1	
Customer_Name	xs:string	<input type="radio"/>	1	1	
Orders	(Orders)	<input type="radio"/>	1	Unbounded	
Order_Key	xs:int...	<input type="radio"/>	1	1	
Order_Price	xs:dec...	<input type="radio"/>	1	1	
Order_Date	xs:date	<input type="radio"/>	1	1	

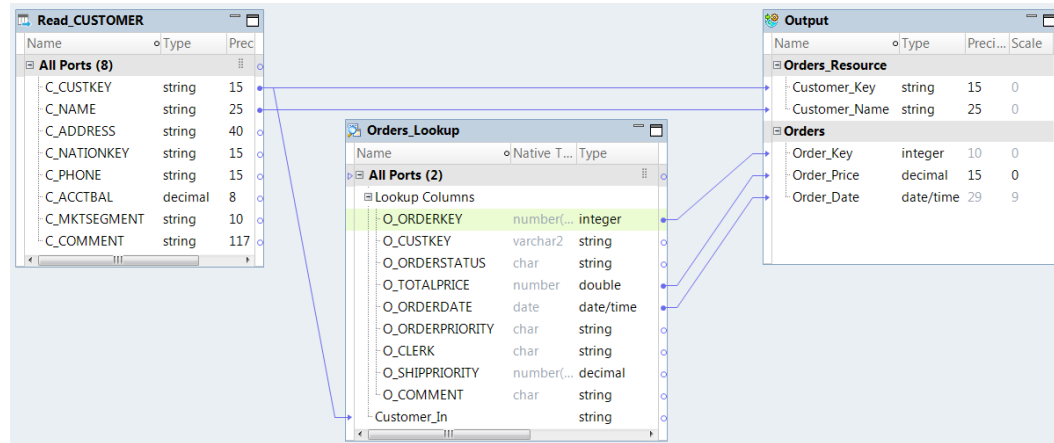
HTTP method(s):
☒ Get

< Back Next > Finish Cancel

2단계. 리소스 매핑 정의

사용자가 리소스를 정의한 후, REST 출력 변환이 포함된 리소스 매핑을 Developer tool이 생성합니다. REST 출력 변환 구조는 웹 서비스가 클라이언트에 반환하는 응답 메시지를 정의합니다. 매핑에 읽기 변환과 조회 변환을 추가합니다.

다음 이미지는 생성할 리소스 매핑을 보여 줍니다.



이 매핑에는 다음 개체가 포함됩니다.

Read_Customer

Read_Customer 변환은 고객 테이블을 읽습니다. 이 변환은 테이블에서 모든 고객 행을 반환합니다.

Orders_Lookup

조회 변환은 각 고객의 주문을 검색합니다.

출력

REST 출력 변환은 Read_Customer 변환에서 고객 정보를 수신합니다. 이 변환은 Orders_Lookup 변환에서 주문 정보를 수신합니다. REST 출력 변환은 각 고객의 주문이 나열되는 계층 JSON 파일을 생성합니다.

참고: 리소스 매핑에는 고객 테이블에서 검색하는 고객을 제한하는 필터 변환이 포함되지 않습니다.

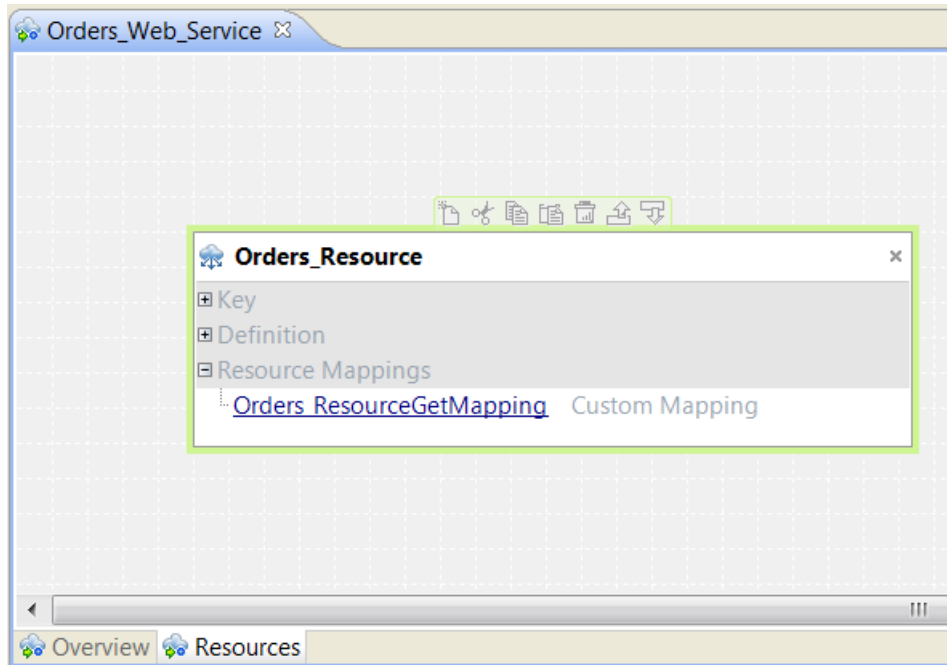
데이터 통합 서비스는 클라이언트 요청의 임의의 필터를 기반으로 출력 데이터를 필터링합니다.

리소스 매핑 정의

Developer tool은 사용자가 생성한 REST 출력 변환이 포함된 리소스 매핑을 생성합니다. 리소스 매핑에 읽기 변환과 조회 변환을 정의합니다.

1. 리소스 탭에서 Orders_Resource에 있는 리소스 매핑 링크를 확장합니다.

다음 이미지는 Orders_Resource에 있는 Orders_Resource 링크를 보여 줍니다.



2. Orders_Resource에서 Orders_ResourceGetMapping 링크를 클릭합니다.
편집기에 매핑이 표시됩니다.
3. Customer 데이터 개체를 읽기 변환으로 매핑에 추가합니다.
4. REST 출력 변환을 클릭합니다.
5. REST 출력 변환 속성 탭에서 포트 보기를 클릭합니다.
6. Customer_Key의 전체 자릿수를 15로 변경하고 Customer_Name의 전체 자릿수를 25로 변경합니다.

Properties Data Viewer Alerts Validation Log					
General		Show: <input checked="" type="radio"/> Ports <input type="radio"/> Output Mapping			
Ports		Ports:			
	Name	Type	Preci...	Scale	Description
Orders_Resource					
1	Customer_Key	string	15	0	
2	Customer_Name	string	25	0	
Orders					
1	Order_Key	integer	10	0	
2	Order_Price	decimal	15	0	
3	Order_Date	date/time	29	9	
4	Customer_FKey	string	15	0	

7. Read_Customer의 Customer Key 및 Customer Name 포트를 REST 출력 변환에 연결합니다.

다음 이미지는 Read_Customer 변환과 REST 출력 변환 간의 링크를 보여 줍니다.

Name	Type	Prec
All Ports (8)		
C_CUSTKEY	string	15
C_NAME	string	25
C_ADDRESS	string	40
C_NATIONKEY	string	15
C_PHONE	string	15
C_ACCTBAL	decimal	8
C_MKTSEGMENT	string	10
C_COMMENT	string	117

Name	Type	Preci...	Scale
Orders_Resource			
Customer_Key	string	15	0
Customer_Name	string	25	0
Orders			
Order_Key	integer	10	0
Order_Price	decimal	15	0
Order_Date	date/time	29	9

8. 조회 변환을 추가하려면 매핑에서 마우스 오른쪽 단추를 클릭하고 **변환 추가**를 클릭합니다.
9. 조회 변환을 선택합니다.
10. 관계형 데이터 개체 조회를 선택하고 **다음**을 클릭합니다.
11. **새 조회 변환** 대화 상자에서 **Orders** 실제 데이터 개체를 찾아서 선택합니다. 모든 행을 반환하도록 선택합니다.

New Lookup Transformation

Lookup
Create a relational data object Lookup transformation.

Relational Data Object:

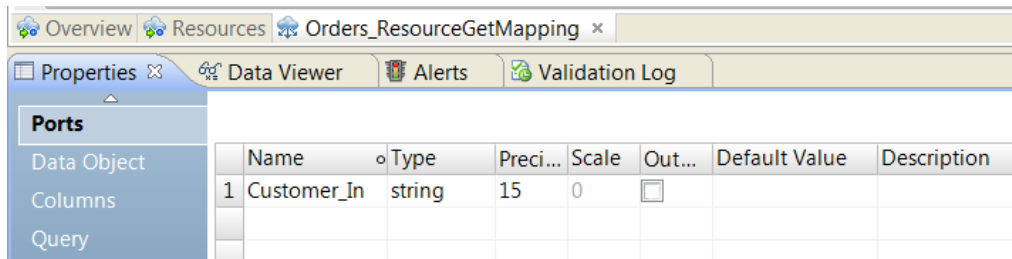
Name:

Location:

On multiple matches:

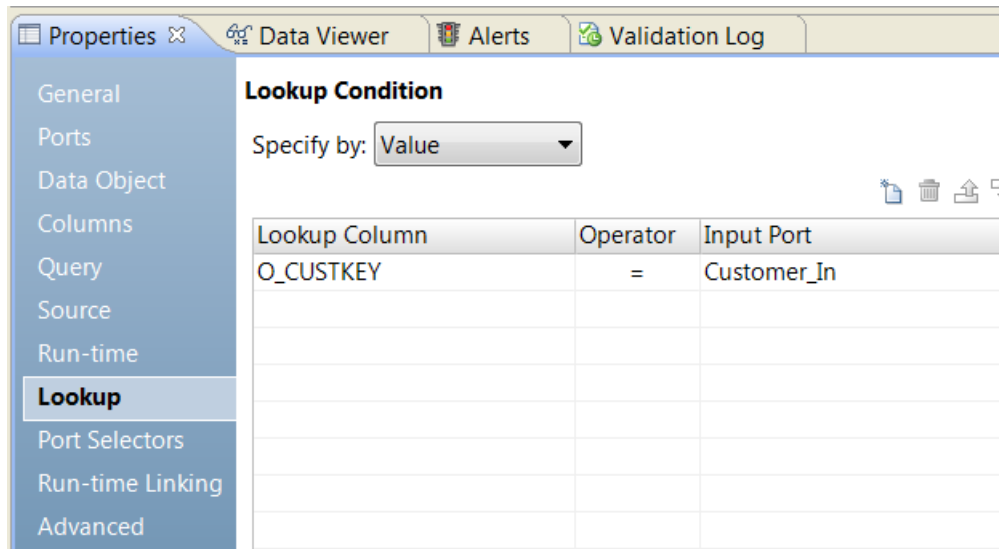
12. **마침**을 클릭합니다.
조회 변환이 리소스 매핑에 나타납니다.
13. 매핑에서 조회 변환을 클릭하여 선택합니다.
14. 조회 변환의 **속성** 보기에서 **포트** 탭을 선택합니다.

15. Customer_In이라는 포트를 추가합니다. 포트는 전체 자릿수가 15인 문자열입니다. 출력을 위해 포트를 활성화하지 않아도 됩니다.



16. 조회 변환의 **속성** 보기에서 **조회** 탭을 클릭합니다.
 17. 다음 조회 조건을 입력합니다.

O_CUSTKEY = Customer_In



18. 매핑 캔버스에서 Read_Customer의 C_CUSTKEY 포트를 조회 변환의 Customer_In 포트에 연결합니다.
 19. 조회 변환의 다음 출력 포트를 REST 출력 변환의 포트에 연결합니다.

조회 변환	출력 변환
O_ORDERKEY	Order_Key
O_TOTALPRICE	Order_Price
O_ORDERDATE	Order_Date

3단계. 출력 매핑 구성

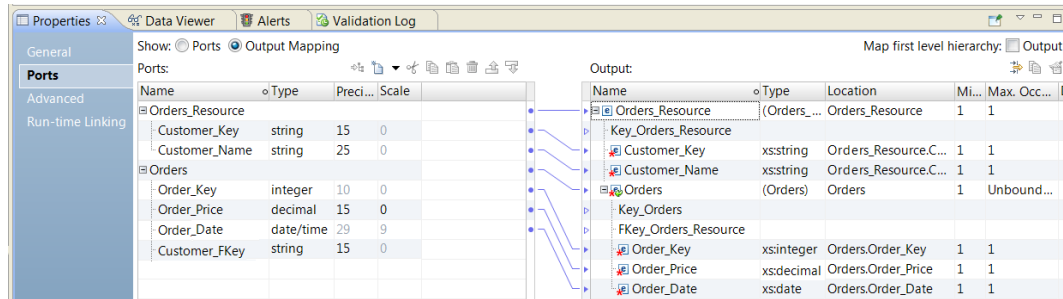
REST 출력 변환은 고객 및 주문 변환에서 데이터를 수신하여 JSON 파일을 반환합니다.

Orders_Resource 매핑에서 REST 출력 변환에는 두 가지 출력 데이터 그룹이 있습니다. 상위 그룹은 고객 정보를 포함하며, 각 고객별로 한 번 발생합니다. 하위 그룹은 주문을 포함합니다. 주문 그룹은 여러 번 발생합니다.

REST 출력 변환에 포트 그룹이 두 개 이상 포함되어 있으면 Developer tool은 해당 그룹을 연결하는 키를 생성합니다. Developer tool은 각 그룹에 기본 키를 생성합니다. Developer tool은 각 하위 그룹에 외래 키를 생성합니다. 주문 그룹의 외래 키에는 각 주문에 대한 고객 번호가 포함됩니다.

REST 출력 변환에서 포트를 볼 때 키는 표시되지 않습니다. 키에는 포트가 없습니다. 그러나 키 값을 정의하려면 데이터를 키에 연결해야 합니다. 키를 보려면 REST 출력 변환 속성 탭의 **포트** 보기에 출력 매핑을 표시합니다. **포트** 보기에는 표시하도록 선택한 옵션에 따라 입력 포트 또는 출력 매핑이 표시됩니다.

다음 이미지는 **출력 매핑** 보기를 보여 줍니다.



Developer tool은 다음 키를 출력에 생성합니다.

키 이름	설명
Key_Orders_Resource	Orders_Resource 그룹의 기본 키입니다.
Key_Orders	주문 그룹의 기본 키입니다.
FKey_Orders_Resource	주문 그룹을 Orders_Resource 그룹에 연결하는 외래 키입니다.

입력 포트를 출력 키에 연결하지 않으면 매핑이 유효하지 않습니다.

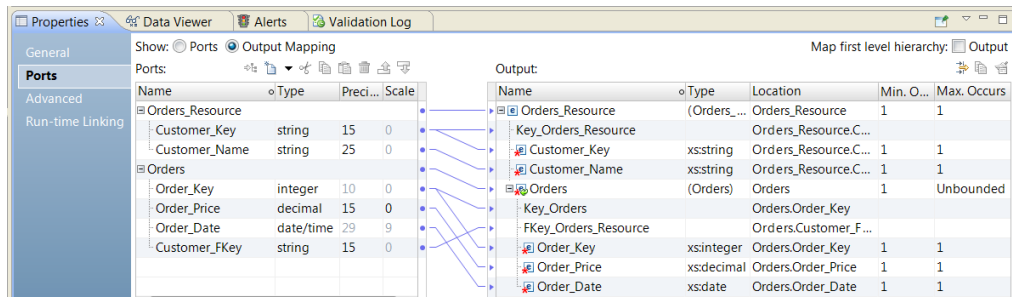
참고: 포트를 출력 패널에 있는 요소 두 개 이상에 연결할 수 있습니다. 그러나 **포트** 패널에 있는 그룹 하나의 필드를 출력 패널의 그룹 두 개 이상에 있는 요소에 연결할 수는 없습니다. 동일한 **포트** 그룹에 포함된 모든 필드는 동일한 출력 그룹에 있는 요소에 연결되어야 합니다.

출력 매핑 구성

고객과 주문 사이에 기본 키-외래 키 관계를 생성하려면 각 그룹에 고객 번호가 포함되어 있어야 합니다.

- REST 출력 변환의 주문 그룹에서 **Order_Date** 아래에 **Customer_Fkey** 포트를 추가합니다.
Customer_Fkey 포트가 고객 번호를 수신합니다. 주문 그룹에서 고객 번호를 수신하고 외래 키를 채우려면 이 포트를 추가해야 합니다. 고객 번호는 출력의 각 주문에 나타나지 않습니다.
- 포트** 패널의 **Customer_Key**를 **출력** 패널의 **Key_Orders_Resource**에 연결합니다. 링크를 생성하려면 **Customer_Key**를 클릭하여 선택한 다음 **Key_Orders_Resource**를 선택합니다. **포트** 패널에서 **맵** 아이콘을 클릭하여 링크를 생성합니다.
참고: 포트를 끌어서 링크를 생성할 수 없습니다.
- Order_Key** 포트와 **Key_Orders** 요소를 선택합니다. **맵**을 클릭하여 링크를 생성합니다.
Order_Key에는 Key_Orders 요소와 Order_Key 요소에 대한 링크가 있어야 합니다.

4. Customer_Fkey 포트와 Fkey_Orders_Resource 요소를 선택합니다. 맵을 클릭하여 링크를 생성합니다.
다음 이미지는 키에 대한 링크가 있는 출력 매핑을 보여 줍니다.



4단계. 데이터 뷰어 보기에서 매핑 테스트

데이터 뷰어 보기에서 REST 리소스 매핑을 실행할 수 있습니다. 데이터 뷰어 보기에서 출력 데이터를 리소스 ID를 기준으로 필터링하거나 필터 조건을 기준으로 필터링할 수 있습니다.

Orders_Resource 그룹의 요소 값을 입력하여 데이터를 필터링할 수 있습니다. 요소는 Customer_Key 또는 Customer_Name입니다. 주문 그룹의 요소를 사용하여 출력 데이터를 필터링할 수는 없습니다. 주문 그룹은 하위 그룹입니다.

출력 데이터를 리소스 식별자를 기준으로 필터링하거나, 필터 조건을 입력할 수 있습니다. 리소스 식별자는 키, 즉 Customer_Key 요소입니다. 필터 조건은 식 편집기에 입력하는 식입니다. 키로 검색하지 않는 경우에 필터 조건을 사용하여 출력 데이터를 필터링합니다.

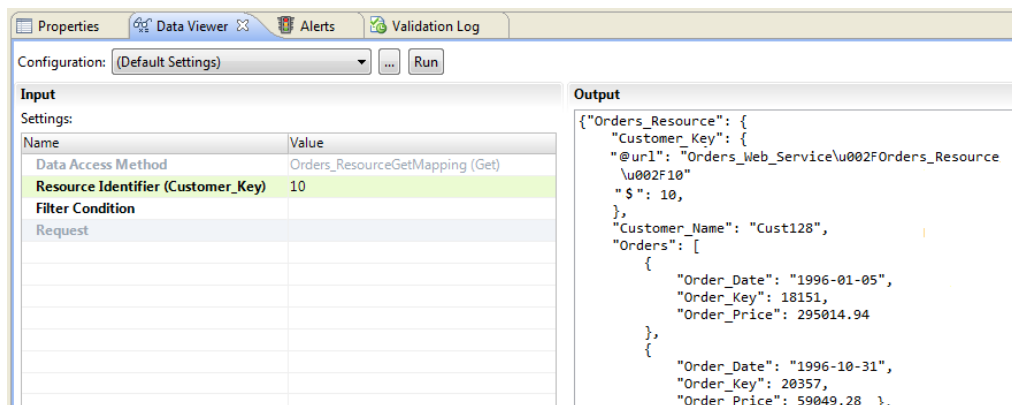
리소스 식별자 또는 필터 조건을 입력하지 않을 경우, 매핑은 데이터 뷰어 보기에 모든 행을 반환합니다.

리소스 ID를 기준으로 출력 필터링

리소스 ID를 입력하여 키 값을 기준으로 출력 데이터를 선택할 수 있습니다. 리소스 ID는 리소스를 생성할 때 선택했던 키입니다. 이 예제에서 리소스 ID는 Customer_Key입니다.

1. 데이터 뷰어 보기의 리소스 식별자 값 열에 유효한 고객 번호를 입력합니다.
2. 실행을 클릭합니다.

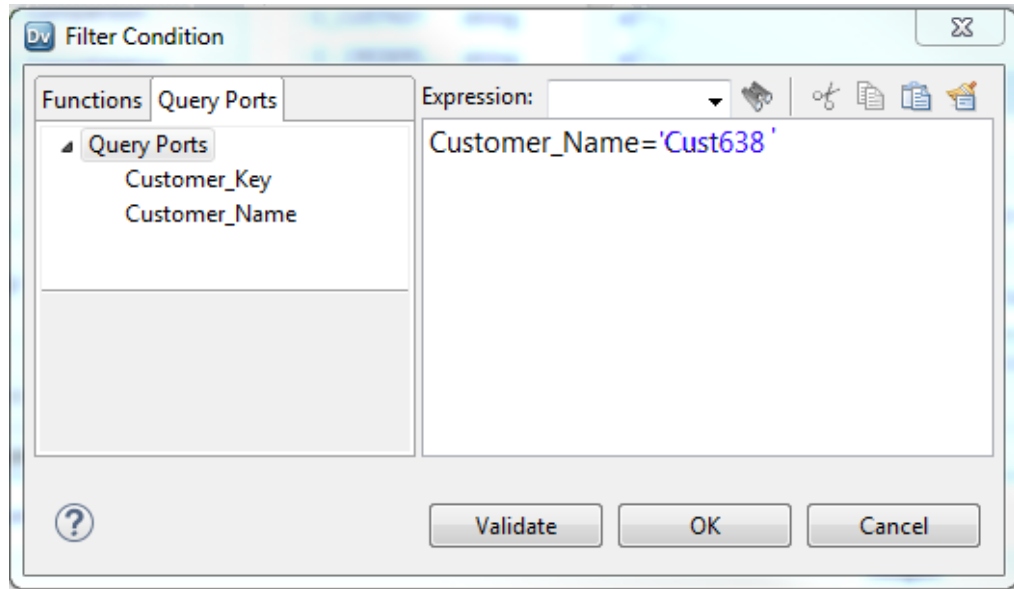
입력한 고객의 데이터가 출력에 포함됩니다.
다음 이미지는 고객 번호 10에 대한 출력을 보여 줍니다.



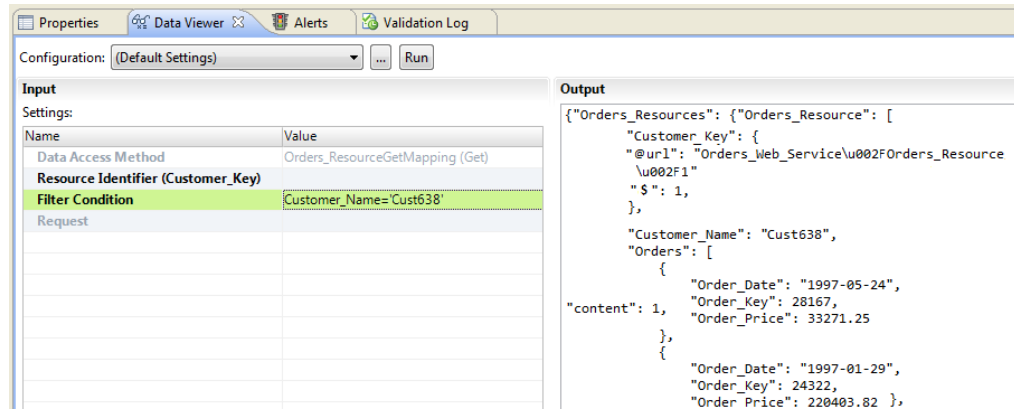
필터 조건을 기준으로 출력 필터링

필터 조건을 입력하여 특정 값을 기준으로 출력 데이터를 선택할 수 있습니다.

1. **데이터 뷰어** 보기의 필터 조건 값 열에서 선택 화살표를 클릭합니다.
식 편집기가 나타납니다.
2. **쿼리 포트** 탭을 클릭합니다.
포트 목록이 나타납니다.



3. 쿼리 포트를 선택하고 필터 식을 생성합니다.
검색 값을 작은따옴표로 묶어야 합니다.
4. **유효성 검사**를 클릭하여 식의 유효성을 검사합니다.
5. 식이 유효하면 **확인**을 클릭합니다.
식이 **데이터 뷰어** 보기에 필터 조건으로 나타납니다.
6. **데이터 뷰어** 보기에서 **실행**을 클릭합니다.
고객 이름 Cust638의 주문이 출력 패널에 나타납니다.
다음 이미지는 출력 패널의 결과를 보여 줍니다.



5단계. 응용 프로그램 배포

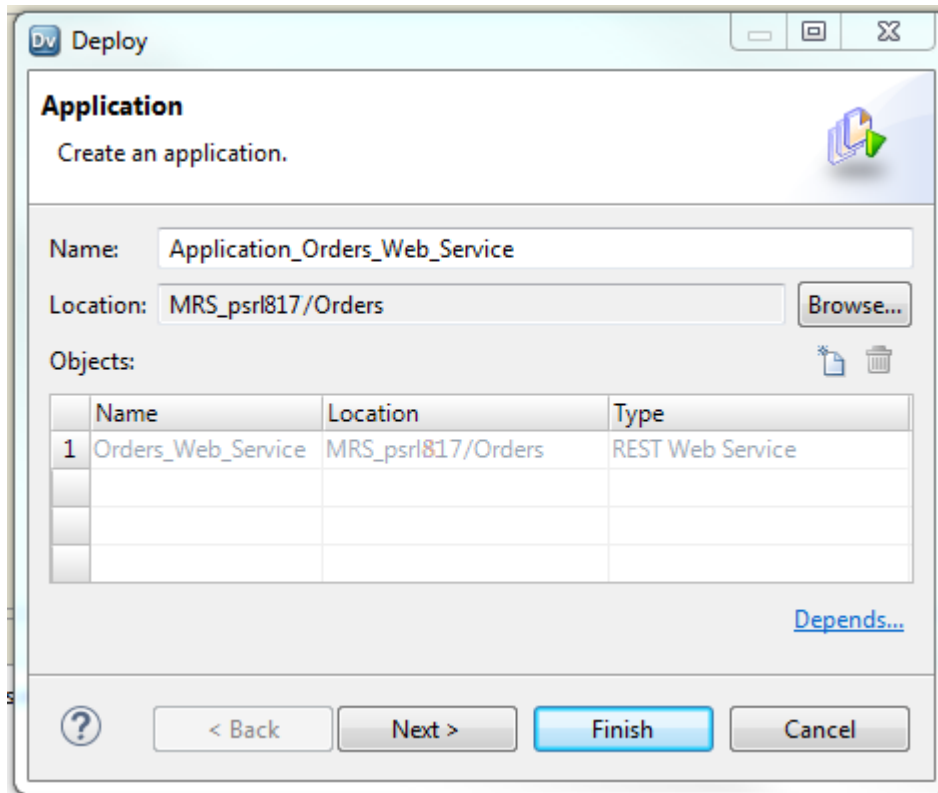
웹 서비스를 데이터 통합 서비스에 응용 프로그램으로 배포합니다. 웹 서비스 클라이언트가 웹 서비스에 연결할 수 있으려면 응용 프로그램을 배포해야 합니다.

웹 서비스를 배포한 후에는 Administrator 도구에서 응용 프로그램을 볼 수 있습니다.

응용 프로그램 배포

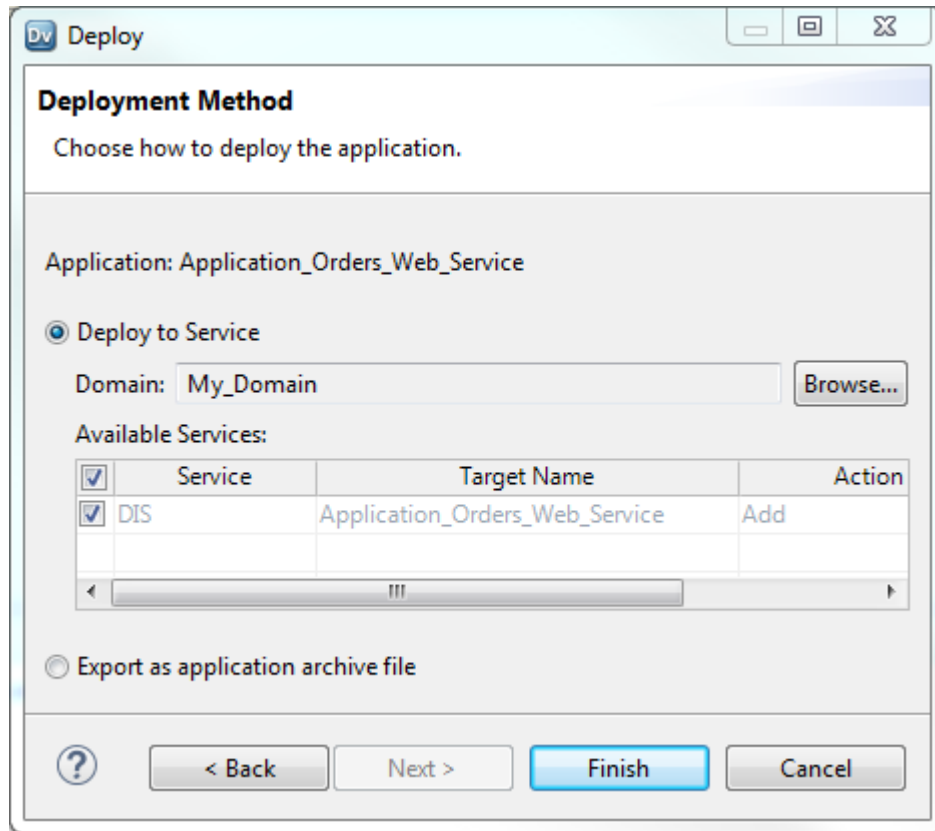
응용 프로그램을 데이터 통합 서비스에 배포합니다.

1. **Object Explorer** 보기에서 **Orders_Web_Service**를 마우스 오른쪽 단추로 클릭하고 **배포**를 클릭합니다.



2. 기본 이름과 위치를 허용합니다. **다음**을 클릭합니다.

3. 응용 프로그램을 배포할 도메인과 데이터 통합 서비스를 선택합니다.



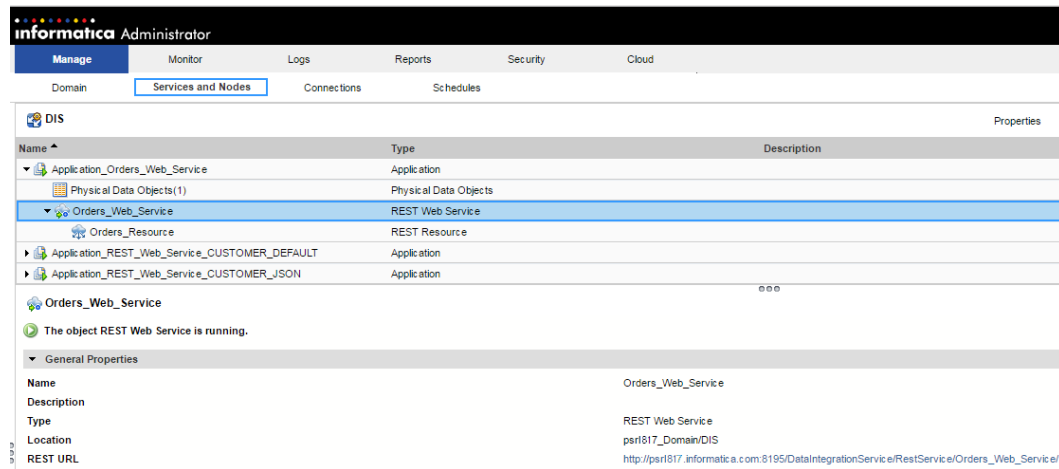
4. 마침을 클릭하여 응용 프로그램을 배포합니다.

6단계. 브라우저에서 웹 서비스 쿼리

브라우저에서 REST 웹 서비스를 쿼리합니다.

웹 서비스를 데이터 통합 서비스에 배포하고 나면 웹 서비스가 **Administrator** 도구에 응용 프로그램으로 나타납니다. 웹 서비스에 액세스하기 위한 URL을 응용 프로그램 **일반 속성** 패널에서 볼 수 있습니다.

다음 이미지는 웹 서비스의 **일반 속성** 패널에 있는 REST URL을 보여 줍니다.



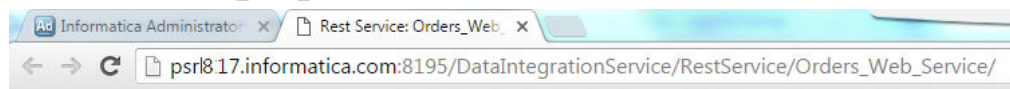
웹 서비스 쿼리

URL에 리소스 ID를 포함하거나 URL에 필터 식을 포함하는 방법으로 웹 서비스를 쿼리할 수 있습니다.

1. Administrator 도구에서 웹 서비스 응용 프로그램 **일반 속성** 패널에 있는 REST URL을 클릭합니다.

데이터 통합 서비스가 웹 서비스의 리소스를 나열합니다.

다음 이미지는 **Orders_Web_Service**의 리소스를 보여 줍니다.



Resources:

- [Orders_Resource](#)

2. **Orders_Resource** 링크를 클릭합니다.

웹 서비스는 리소스 매핑에서 모든 출력 데이터를 반환합니다.

3. 특정 고객을 검색하려면 브라우저에서 URL 끝에 리소스 이름과 리소스 ID를 추가합니다. 링크는 `http://psr1817.informatica.com:8195/DataIntegrationService/RestService/Orders_Web_Service/Orders_Resource/10`입니다.

다음 이미지는 Orders_Resource에서 고객 번호 10을 검색하는 쿼리의 결과를 보여 줍니다.

```
← → ↺ psrl817.informatica.com:8195/DataIntegrationService/RestService/Orders_Web_Service/Orders_Resource/10

{"Orders_Resources": {"Orders_Resource": {
  "Customer_Key": {
    "@url": "http://psrl817.informatica.com:8195/DataIntegrationService/RestService/Orders_Web_Service/Orders_Resource/10"
    "$": 10,
  },
  "Customer_Name": "Cust128",
  "Orders": [
    {
      "Order_Date": "1996-01-05",
      "Order_Key": 18151,
      "Order_Price": 295014.94
    },
    {
      "Order_Date": "1996-10-31",
      "Order_Key": 20357,
      "Order_Price": 59049.28
    },
    {
      "Order_Date": "1997-07-27",
      "Order_Key": 24678,
      "Order_Price": 112857.62
    },
    {
      "Order_Date": "1998-07-03",
      "Order_Key": 25122,
      "Order_Price": 92135.6
    }
  ]
}
```

4. 키 대신 고객 이름을 기준으로 검색하려면 URL에 리소스 ID 대신 필터 조건을 포함합니다.

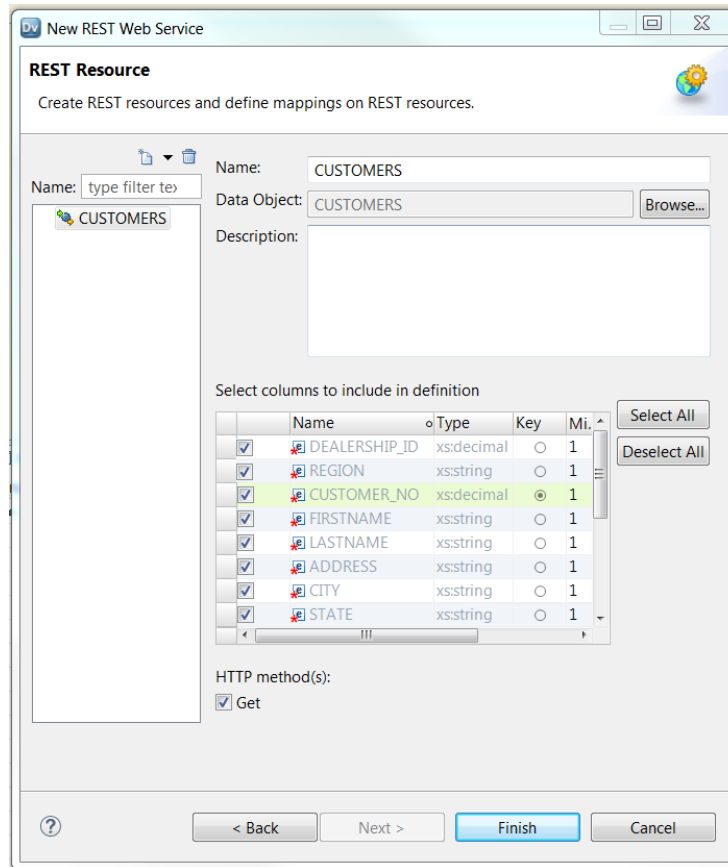
http://psrl817.informatica.com:8195/DataIntegrationService/RestService/Orders_Web_Service/Orders_Resource/?filterCondition=Customer_Name='Cust628'

데이터 개체에서 REST 웹 서비스를 생성하는 방법

모델 리포지토리의 데이터 개체에서 REST 웹 서비스를 생성할 수 있습니다. Developer tool은 읽기 변환 및 출력 변환이 포함된 기본 매핑을 생성합니다. REST 웹 서비스 리소스에 포함할 데이터 개체 열을 선택할 수 있습니다.

1. 파일 > 새로 만들기 > 데이터 서비스를 클릭합니다.
2. 마법사 선택 대화 상자에서 REST 웹 서비스를 선택합니다. 다음을 클릭합니다.
3. REST 웹 서비스 대화 상자에서 웹 서비스 이름을 입력합니다. 기본 이름은 REST_Web_Service입니다. 다음을 클릭합니다.
4. REST 리소스 대화 상자에서 새로 만들기 > 데이터 개체에서 생성을 클릭합니다.
5. 모델 리포지토리의 데이터 개체 목록에서 데이터 개체를 선택하고 확인을 클릭합니다.
선택한 데이터 개체의 REST 리소스가 REST 리소스 대화 상자에 표시됩니다.
6. 리소스에 포함할 열을 선택합니다. 리소스의 기본 키를 선택합니다.
기본적으로 모든 열이 선택되어 있습니다. 리소스에 포함하지 않을 열을 선택 취소할 수 있습니다. 기본적으로 GET 메서드가 선택되어 있습니다. 이 옵션은 지우지 마십시오.

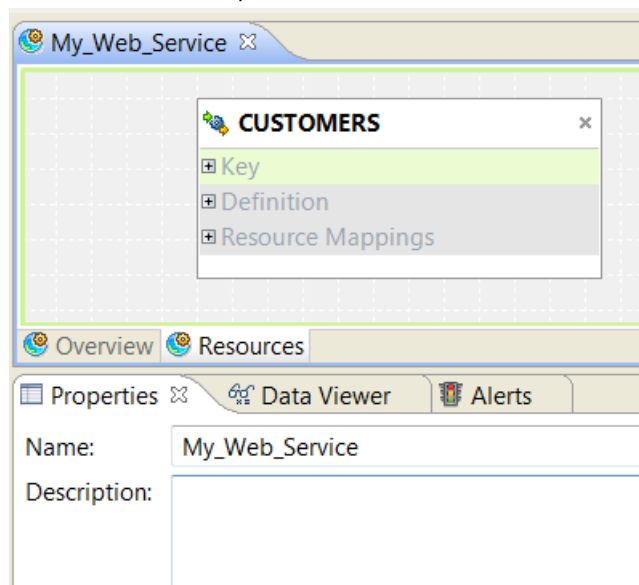
다음 이미지는 REST 리소스 대화 상자를 보여 줍니다.



7. 마침을 클릭하여 웹 서비스를 생성합니다.

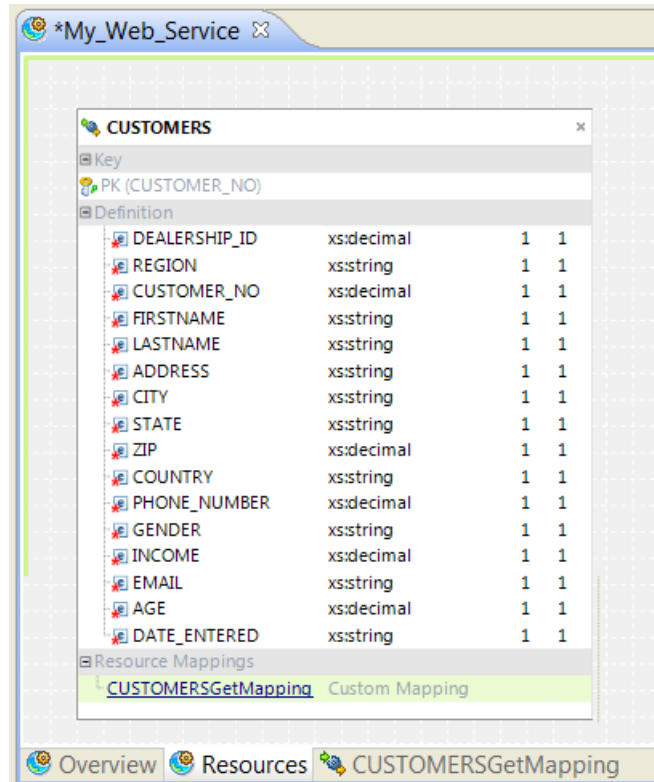
REST 웹 서비스 리소스 보기가 Developer tool에 나타납니다.

다음 이미지는 Developer tool의 리소스 보기를 보여 줍니다.



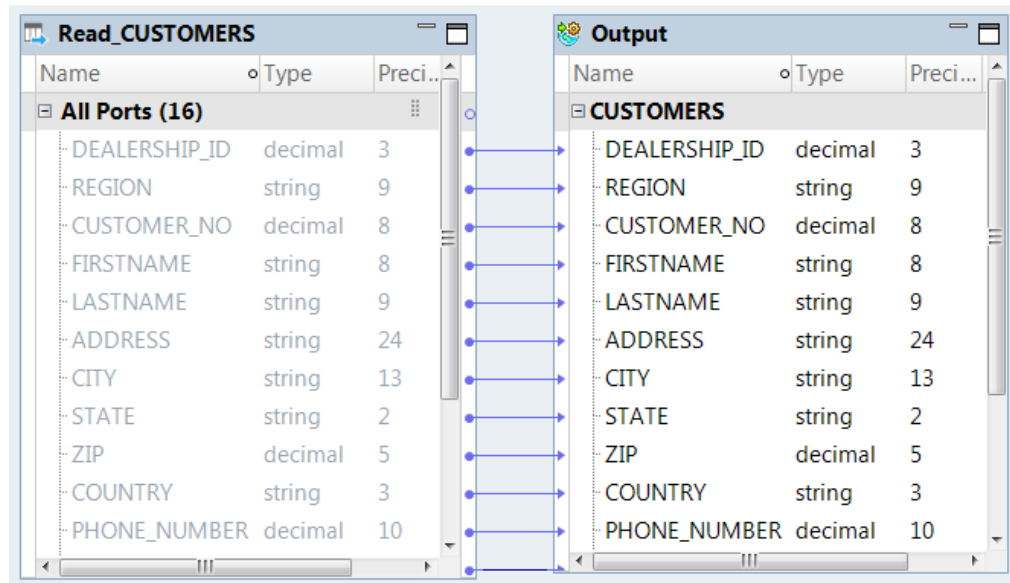
8. 리소스 보기에서 구성 요소를 확장하여 키, 정의에 포함된 열 및 리소스 매핑에 대한 링크를 볼 수 있습니다.

다음 이미지는 리소스 키, 정의 열 및 리소스 매핑에 대한 링크를 보여 줍니다.



9. 리소스 매핑 링크를 클릭하여 리소스 매핑을 봅니다.

다음 이미지는 기본 리소스 매핑을 보여 줍니다.

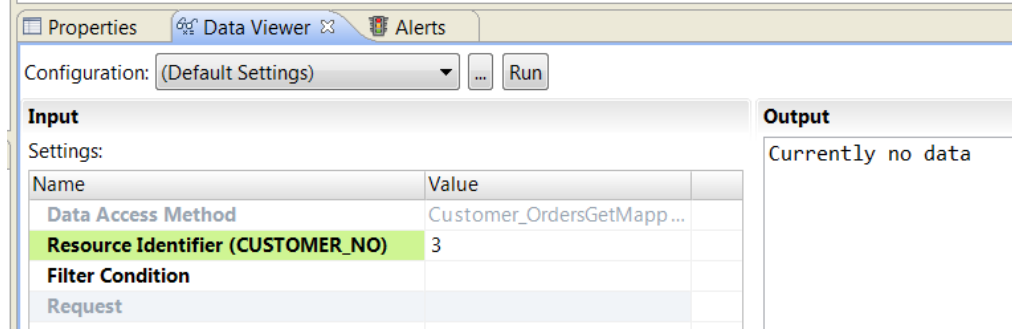


10. 필요한 경우 매핑에 변환을 추가합니다.
11. 웹 서비스를 테스트하려면 **데이터 뷰어** 보기로 이동합니다.
12. **입력 설정의 데이터 액세스 값** 필드에서 테스트할 리소스 매핑을 선택합니다.
13. 키를 기준으로 출력을 필터링하려면 검색할 키를 **리소스 식별자** 필드에 입력합니다.

- 다른 요소를 기준으로 출력을 필터링하려면 필터 조건을 입력합니다. 값 필드를 클릭하여 식 편집기를 엽니다.

식 편집기에는 식을 여러 개 입력할 수 있습니다.

다음 이미지는 데이터 뷰어 보기의 입력 설정을 보여 줍니다.



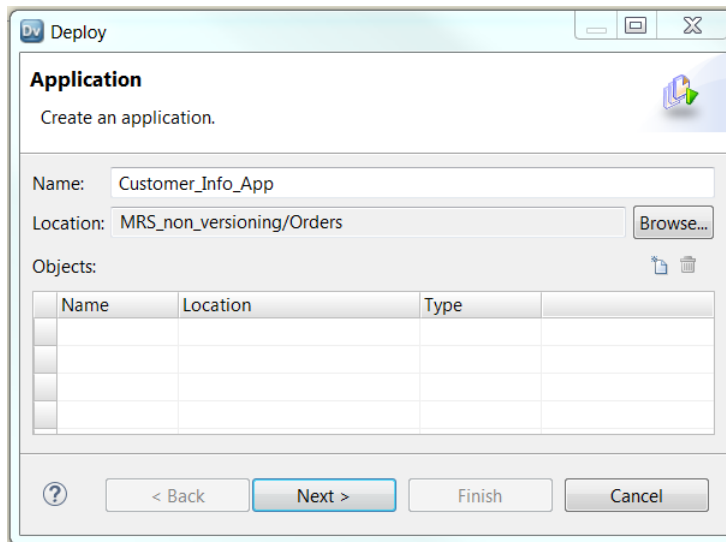
- 실행을 클릭하여 출력을 봅니다.
출력은 웹 서비스 구성에 기반하여 JSON 또는 XML입니다.
- 웹 서비스를 데이터 통합 서비스에 배포합니다.

데이터 개체를 REST 웹 서비스로 배포하는 방법

관계형 데이터 개체 또는 플랫폼 파일 데이터 개체를 REST 웹 서비스로 배포할 수 있습니다.

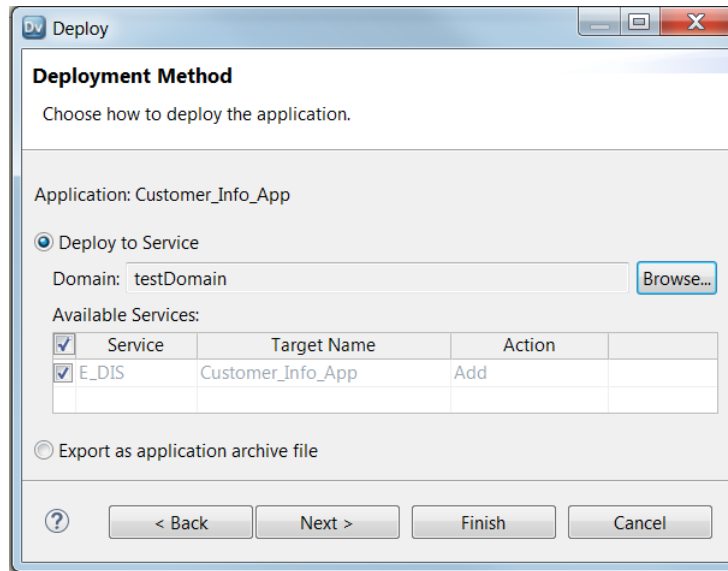
- Object Explorer에서 데이터 개체를 마우스 오른쪽 단추로 클릭하여 웹 서비스에 액세스합니다.
- 배포 > REST 웹 서비스로 배포를 클릭합니다.

응용 프로그램 배포 대화 상자가 나타납니다.



- 응용 프로그램의 이름을 입력하고 다음을 클릭합니다. 개체는 추가하지 않습니다.
배포 방법 대화 상자가 나타납니다.
- 응용 프로그램을 배포할 데이터 통합 서비스를 선택합니다.

다음 이미지는 **배포 방법** 대화 상자를 보여 줍니다.

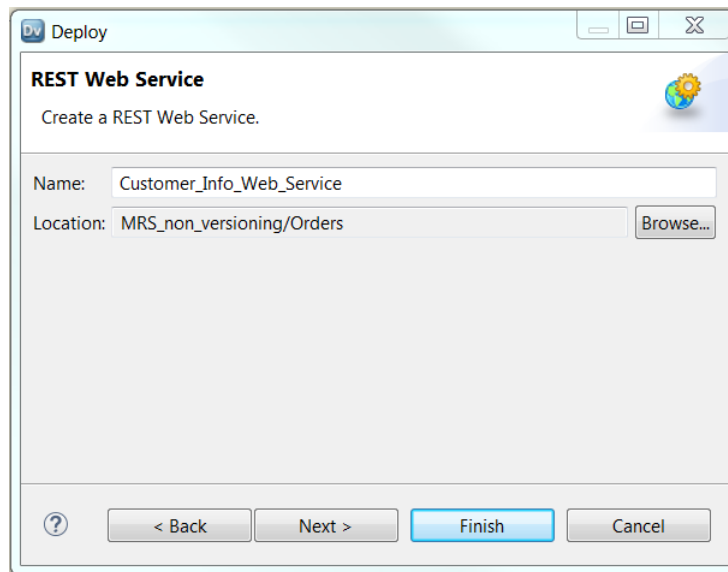


5. 다음을 클릭합니다.

REST 웹 서비스 대화 상자가 나타납니다.

6. 웹 서비스 이름을 입력합니다.

다음 이미지는 **REST 웹 서비스** 대화 상자를 보여 줍니다.



7. 리소스에 포함할 열을 선택하거나 더 많은 리소스를 추가하려면 다음을 클릭합니다. 마침을 클릭하여 데이터 개체의 모든 열이 포함된 리소스를 생성합니다.

다음을 클릭하면 **REST 리소스** 대화 상자가 나타납니다.

8. REST 리소스 대화 상자에서 리소스의 이름을 입력하고 정의에 포함할 열을 선택합니다.

기본적으로 모든 열이 선택되어 있습니다. 필드를 지우지 마십시오. 배포된 웹 서비스에는 기본 매핑이 필요합니다.

다음 이미지는 REST 리소스 대화 상자를 보여 줍니다.

REST Resource
Create REST resources and define mappings on REST resources.

Name: type filter text
CUSTOMERS

Name: CUSTOMERS
Data Object: CUSTOMERS
Description: Return customer information by customer ID

Select columns to include in definition

Name	Type	Key	Mi...	M...	Desc
DEALERSHIP_ID	xs:decimal	<input type="radio"/>	1	1	
REGION	xs:string	<input type="radio"/>	1	1	
CUSTOMER_NO	xs:decimal	<input checked="" type="radio"/>	1	1	
FIRSTNAME	xs:string	<input type="radio"/>	1	1	
LASTNAME	xs:string	<input type="radio"/>	1	1	
ADDRESS	xs:string	<input type="radio"/>	1	1	
CITY	xs:string	<input type="radio"/>	1	1	
STATE	xs:string	<input type="radio"/>	1	1	
ZIP	xs:decimal	<input type="radio"/>	1	1	
COUNTRY	xs:string	<input type="radio"/>	1	1	
PHONE_NUMBER	xs:decimal	<input type="radio"/>	1	1	
GENDER	xs:string	<input type="radio"/>	1	1	
INCOME	xs:decimal	<input type="radio"/>	1	1	

HTTP method(s):
☒ Get

< Back Next > Finish Cancel

9. 마침을 클릭하여 웹 서비스를 배포합니다.
리소스를 추가하거나 리소스 정의를 변경하지 마십시오.
10. 웹 서비스에 리소스를 더 추가하려면 새로 만들기를 클릭합니다.
데이터 개체에서 리소스를 생성하거나 리소스의 열을 수동으로 입력할 수 있습니다.

제 12 장

REST 웹 서비스 소비자 변환

이 장에 포함된 항목:

- [REST 웹 서비스 소비자 변환 개요, 136](#)
- [REST 웹 서비스 소비자 변환 구성, 138](#)
- [HTTP 메서드, 139](#)
- [REST 웹 서비스 소비자 변환 포트, 142](#)
- [REST 웹 서비스 소비자 변환 입력 매핑, 144](#)
- [REST 웹 서비스 소비자 변환 출력 매핑, 146](#)
- [REST 웹 서비스 소비자 변환의 고급 속성, 148](#)
- [REST 웹 서비스 소비자 변환 작성, 149](#)
- [배열이 포함된 JSON 응답 메시지 구문 분석, 150](#)

REST 웹 서비스 소비자 변환 개요

REST 웹 서비스 소비자 변환은 웹 서비스 클라이언트로 REST 웹 서비스에 연결하여 데이터에 액세스하거나 데이터를 변환하는 활성 변환입니다. REST 웹 서비스에 연결하려면 REST 웹 서비스 소비자 변환을 사용합니다. REST 웹 서비스 소비자 변환은 REST 웹 서비스에 요청을 전송하고 REST 웹 서비스의 응답을 수신합니다.

REST 웹 서비스 소비자 변환은 변환 또는 HTTP 연결에서 정의한 URL을 통해 웹 서비스에 연결합니다. HTTPS 연결도 사용할 수 있습니다. REST 웹 서비스 소비자 변환은 TLS 1.2, TLS 1.1 또는 TLS 1.0을 사용할 수 있습니다.

REST 웹 서비스에는 웹 서비스가 지원하는 각 작업에 대한 HTTP 메서드가 포함됩니다. 데이터 통합 서비스는 REST 웹 서비스에 연결하여 데이터를 가져오거나 게시하거나 배치하거나 삭제하는 요청을 전송할 수 있습니다. 요청은 개별 리소스 또는 리소스 컬렉션에 따라 작동할 수 있습니다. 데이터 통합 서비스는 요청 메시지를 전송한 후에 웹 서비스로부터 응답 메시지를 수신합니다.

요청 및 응답 메시지에는 계층을 형성할 수 있는 요소가 있는 XML 또는 JSON 데이터가 포함됩니다. 요청 또는 응답 메시지에 여러 번 발생하는 요소가 포함되는 경우 요소의 그룹에 의해 XML 또는 JSON 계층의 수준이 형성됩니다. 그룹은 한 수준이 다른 수준 안에 중첩될 경우에 연결됩니다.

REST 웹 서비스 소비자 변환에서 메서드 입력과 메서드 출력은 요청 및 응답 메시지의 구조를 정의합니다. 메서드 입력 및 메서드 출력에는 메시지 요소를 입력 및 출력 포트에 매핑하는 방법을 정의하는 매핑이 포함됩니다.

REST 웹 서비스 소비자 변환은 프록시 서버를 지원합니다. Microsoft SharePoint 응용 프로그램을 REST 웹 서비스 소비자 변환에 연결할 수도 있습니다.

예

한 온라인 매장에서 제품 데이터베이스에 대한 리소스를 정의합니다. 이 데이터베이스는 각 제품을 제품 번호로 식별합니다.

웹 서비스 클라이언트는 REST 웹 서비스를 통해 제품 세부 정보에 액세스합니다. 웹 서비스는 다음 URL을 사용합니다.

<http://www.HypoStores.com/products/ProductDetails>

이때 설명 및 단위 가격 등 특정 제품에 대한 세부 정보를 검색하고 세부 정보를 매핑의 변환 다운스트림에 전달하려면 제품에 대한 세부 정보를 검색하고 이러한 세부 정보를 다른 변환에 전달하는 REST 웹 서비스 소비자 변환을 작성해야 합니다.

다음 테이블에는 구성해야 하는 변환 세부 정보가 표시되어 있습니다.

변환 세부 정보	값
HTTP 메서드	Get
기본 URL	http://www.HypoStores.com/products/ProductDetails
입력 인수 포트	Part_No
출력 포트	설명, Unit_Price
메서드 출력	<응답 메시지의 구조입니다.>

메서드 출력에는 응답 메시지의 요소를 출력 포트에 매핑하는 방법을 정의하는 출력 매핑이 포함됩니다.

데이터 통합 서비스는 웹 서비스에 요청을 전송할 때 인수 포트의 값을 기본 URL에 추가합니다. 예를 들어 제품 0716에 대한 세부 정보를 검색하려는 경우 데이터 통합 서비스는 다음 URL을 사용합니다.

http://www.HypoStores.com/products/ProductDetails?Part_No=0716

데이터 통합 서비스는 응답을 수신할 때 응답 메시지의 제품 설명 및 단위 가격을 출력 포트에 대한 데이터로 변환합니다.

Part_No를 매개 변수로 전달하여 매핑을 실행하는 중간에 값을 대체할 수도 있습니다.

REST 웹 서비스 소비자 변환 프로세스

REST 웹 서비스 소비자 변환은 입력 포트 및 메서드 입력의 데이터를 바탕으로 요청 메시지를 작성합니다. 또한 메서드 출력에 따라 응답 메시지의 요소를 출력 포트에 대한 데이터로 변환합니다.

REST 웹 서비스 소비자 변환의 입력 포트에는 매핑의 업스트림 변환으로부터 수신한 관계형 데이터가 포함됩니다. 데이터 통합 서비스는 메서드 입력을 사용하여 입력 포트의 데이터를 요청 메시지의 요소로 변환합니다.

데이터 통합 서비스는 변환 속성 또는 HTTP 연결에서 구성된 기본 URL을 읽어 웹 서비스에 연결합니다. 또한 URL 포트 또는 인수 포트의 값을 기본 URL에 추가하여 사용자가 가져오거나 게시하거나 배치하거나 삭제하려는 리소스를 식별합니다.

데이터 통합 서비스는 응답을 수신한 다음 응답 메시지의 데이터를 변환의 출력 포트에 전달합니다. 데이터 통합 서비스는 메서드 출력의 구성에 따라 데이터를 전달합니다. 출력 포트에는 관계형 데이터가 포함됩니다. 데이터 통합 서비스는 출력 포트의 데이터를 매핑의 다운스트림 변환 또는 대상으로 전송합니다.

REST 웹 서비스 소비자 변환 구성

REST 웹 서비스 소비자 변환을 작성할 때 HTTP 메서드를 선택하고 메서드 입력 및 출력을 정의합니다. Get 메서드를 선택한 경우에는 메서드 입력을 정의하지 않습니다.

HTTP 요청 메시지의 입력 요소는 입력 포트에 매핑됩니다. HTTP 요청 메시지의 출력 요소는 출력 포트에 매핑됩니다. Developer tool에서는 첫 번째 수준 요소에 대한 포트가 작성됩니다.

변환을 구성할 때 다음 태스크를 완료하십시오.

1. HTTP 메서드를 선택합니다.
2. 요청 및 응답 메시지의 헤더 및 본문에 있는 요소를 나타내는 포트를 구성합니다.
3. 입력 매핑을 구성합니다.
4. 출력 매핑을 구성합니다.
5. 웹 서비스에 대한 연결 및 URL과 같은 고급 속성을 구성합니다.

REST 웹 서비스에 인증이 필요한 경우 HTTP 연결 개체를 작성합니다.

메시지 구성

데이터 통합 서비스는 REST 웹 서비스 소비자 변환에서 구성한 메서드 입력 및 출력과 포트에 따라 요청 메시지를 생성하고 응답 메시지를 해석합니다.

입력 포트는 요청 메시지의 서로 다른 부분을 나타냅니다. 검색하거나 변경할 리소스를 식별하는 입력 포트를 추가할 수 있습니다. 또한 요청 메시지에서 HTTP 헤더, 쿠키 정보 및 요소를 나타내는 입력 포트를 추가할 수도 있습니다.

출력 포트는 매핑의 다운스트림 변환 또는 대상으로 전송할 응답 메시지의 요소를 나타냅니다. 응답 메시지에서 HTTP 헤더, 쿠키 정보, 응답 코드 및 요소를 나타내는 출력 포트를 추가할 수 있습니다.

리소스 식별

데이터 통합 서비스는 HTTP 요청의 리소스를 식별하기 위해 특정 입력 포트의 값을 기본 URL에 추가합니다. 기본 URL은 HTTP 연결 또는 변환 속성에 정의합니다. URL 또는 인수 포트를 사용하여 특정 리소스를 식별할 수 있습니다.

웹 서비스가 고유한 문자열을 통해 리소스를 식별하는 경우 URL 포트를 사용합니다.

예를 들어 HypoStores REST 웹 서비스는 다음 URL을 통해 제품 번호로 제품을 식별합니다.

`http://www.HypoStores.com/products/ProductDetails/<Part_No>`

제품을 식별하려면 다음 변환 세부 정보를 정의합니다.

1. 기본 URL을 다음 URL로 설정합니다.
`http://www.HypoStores.com/products/ProductDetails`
2. URL 포트를 정의하고 URL 포트를 통해 제품 번호를 변환에 전달합니다.

매핑이 제품 번호 500을 URL 포트에 전달할 경우 데이터 통합 서비스는 다음 URL을 요청 메시지에 사용합니다.

`http://www.HypoStores.com/products/ProductDetails/500`

웹 서비스가 인수를 통해 리소스의 위치를 식별하는 경우에는 인수 포트를 사용합니다.

예를 들어 제품 번호를 "Part_No" 인수를 통해 HypoStores REST 웹 서비스에 전달하려고 합니다.

제품을 식별하려면 다음 변환 세부 정보를 정의합니다.

1. 기본 URL을 다음 URL로 설정합니다.

`http://www.HypoStores.com/products/ProductDetails`

2. 인수 이름 "Part_No"를 사용하여 인수 포트를 작성하고 이 인수 포트를 사용하여 제품 번호를 변환에 전달합니다.

매핑이 제품 번호 600을 URL 포트에 전달할 경우 데이터 통합 서비스는 다음 URL을 요청 메시지에 사용합니다.

`http://www.HypoStores.com/products/ProductDetails?Part_No=600`

여러 인수를 정의하려면 여러 인수 포트를 작성합니다. 데이터 통합 서비스에서는 앰퍼샌드(&)를 사용하여 각 인수를 구분합니다.

예를 들어 REST 웹 서비스에서 직원 세부 정보를 검색하고 직원의 이름과 성을 "First_Name" 및 "Last_Name" 인수를 통해 전달하려고 합니다. 그러려면 "First_Name" 및 "Last_Name"이라는 인수 이름을 사용하여 인수 포트를 작성합니다. 매핑이 이름 "John Smith"를 변환에 전달하면 데이터 통합 서비스는 다음과 같은 URL을 요청 메시지에 사용합니다.

`http://www.HypoStores.com/employees/EmpDetails?First_Name=John&Last_Name=Smith`

URL 또는 인수 포트를 지정하지 않을 경우 데이터 통합 서비스는 변환 속성 또는 HTTP 연결의 기본 URL을 사용하여 리소스를 식별합니다. HTTP 연결의 기본 URL은 변환의 기본 URL을 재정의합니다.

HTTP 메서드

REST 웹 서비스 소비자 변환을 작성할 때는 데이터 통합 서비스에서 요청 메시지에 사용할 HTTP 메서드를 선택해야 합니다. 변환을 작성한 후에는 HTTP 메서드를 변경할 수 없습니다.

다음 HTTP 메서드 중 하나를 사용하도록 변환을 구성합니다.

Get

웹 서비스에서 리소스 또는 리소스 컬렉션을 검색합니다. 예를 들어 제품의 테이블을 검색하거나 단일 제품에 대한 정보를 검색할 수 있습니다.

Post

데이터를 웹 서비스로 전송합니다. Post 메서드는 리소스 또는 리소스 컬렉션을 작성할 때 사용됩니다. 예를 들어 신규 매장의 트랜잭션 세부 정보를 추가할 수 있습니다.

Put

리소스 또는 리소스 컬렉션을 바꿉니다. 데이터가 존재하지 않는 경우에는 Put 메서드가 데이터를 게시합니다. 예를 들어 고객의 배송 주소를 업데이트할 수 있습니다.

Delete

리소스 또는 리소스 컬렉션을 삭제합니다. 예를 들어 조직에서 더 이상 일하지 않는 직원의 레코드를 삭제할 수 있습니다.

HTTP Get 메서드

데이터 통합 서비스에서는 REST 웹 서비스에서 데이터를 검색할 때 HTTP Get 메서드를 사용합니다. Get 메서드는 리소스 또는 리소스 컬렉션을 검색할 때 사용됩니다.

Get 메서드를 사용하도록 REST 웹 서비스 소비자 변환을 구성하는 경우 입력 포트, 메서드 출력 및 출력 포트를 구성해야 합니다. 메서드 입력은 구성하지 않습니다.

예

HypoStores 제품 데이터베이스에서 제품 번호 500에 대한 설명 및 가격을 검색하려고 합니다. 웹 서비스에서는 다음 URL을 사용하여 제품을 식별합니다.

`http://www.HypoStores.com/products/ProductDetails?Part_No=<Part_No>`

다음 기본 URL을 입력합니다.

`http://www.HypoStores.com/products/ProductDetails`

다음 테이블에는 정의할 수 있는 입력 포트가 표시되어 있습니다.

포트 유형	인수 이름	입력 값
인수	Part_No	500

다음 테이블에는 정의할 수 있는 출력 포트가 표시되어 있습니다.

포트 유형	포트 이름	반환 값
출력	Part_Desc	...<desc>ACME 볼펜, 12pk, 검정, 0.7mm</desc>...
출력	Price_USD	...<price>9.89</price>...

HTTP Post 메서드

데이터 통합 서비스에서는 REST 웹 서비스에 데이터를 전송할 때 HTTP Post 메서드를 사용합니다. Post 메서드가 수행하는 실제 기능은 웹 서비스를 통해 결정됩니다. Post 메서드를 사용하여 리소스 또는 리소스 컬렉션을 작성할 수도 있습니다.

Post 메서드를 사용하도록 REST 웹 서비스 소비자 변환을 구성하는 경우 입력 포트, 메서드 입력, 메서드 출력 및 출력 포트를 구성해야 합니다.

예

HypoStores 제품 데이터베이스에 새 제품 501을 게시하려고 합니다. 웹 서비스에서는 제품 501에 대해 다음 URL을 사용합니다.

`http://www.HypoStores.com/products/ProductDetails/501`

다음 기본 URL을 입력합니다.

`http://www.HypoStores.com/products/ProductDetails`

다음 테이블에는 정의할 수 있는 입력 포트가 표시되어 있습니다.

포트 유형	포트 이름	입력 값
URL	URL_Part_No	501
입력	Part_Desc	ACME 볼펜, 12pk, 검정, 0.5mm
입력	Price_USD	9.89

다음 테이블에는 정의할 수 있는 출력 포트가 표시되어 있습니다.

포트 유형	포트 이름	반환 값
출력	응답	<웹 서비스에서 반환하는 응답>

HTTP Put 메서드

데이터 통합 서비스에서는 REST 웹 서비스를 통해 데이터를 업데이트할 때 HTTP Put 메서드를 사용합니다. Put 메서드는 리소스 또는 리소스 컬렉션을 업데이트할 때 사용됩니다. 데이터가 존재하지 않을 경우에는 데이터 통합 서비스가 리소스 또는 리소스 컬렉션을 작성합니다.

Put 메서드를 사용하도록 REST 웹 서비스 소비자 변환을 구성하는 경우 입력 포트, 메서드 입력, 메서드 출력 및 출력 포트를 구성해야 합니다.

예

HypoStores 제품 데이터베이스의 제품 501에 대한 단위 가격을 업데이트하려고 합니다. 웹 서비스에서는 제품 501에 대해 다음 URL을 사용합니다.

`http://www.HypoStores.com/products/ProductDetails/501`

다음 기본 URL을 입력합니다.

`http://www.HypoStores.com/products/ProductDetails`

다음 테이블에는 정의할 수 있는 입력 포트가 표시되어 있습니다.

포트 유형	포트 이름	입력 값
URL	URL_Part_No	501
입력	Price_USD	9.99

다음 테이블에는 정의할 수 있는 출력 포트가 표시되어 있습니다.

포트 유형	포트 이름	반환 값
출력	응답	<웹 서비스에서 반환하는 응답>

HTTP Delete 메서드

데이터 통합 서비스에서는 REST 웹 서비스를 통해 데이터를 제거할 때 HTTP Delete 메서드를 사용합니다. Delete 메서드는 리소스 또는 리소스 컬렉션을 제거할 때 사용됩니다.

Delete 메서드를 사용하도록 REST 웹 서비스 소비자 변환을 구성하는 경우 입력 포트, 메서드 입력, 메서드 출력 및 출력 포트를 구성해야 합니다.

예

HypoStores 제품 데이터베이스에서 제품 번호 502를 삭제하려고 합니다. 웹 서비스에서는 다음 URL을 사용하여 제품을 식별합니다.

`http://www.HypoStores.com/products/ProductDetails?Part_No=<Part_No>`

다음 기본 URL을 입력합니다.

<http://www.HypoStores.com/products/ProductDetails>

다음 테이블에는 정의할 수 있는 입력 포트가 표시되어 있습니다.

포트 유형	인수 이름	입력 값
인수	Part_No	502

다음 테이블에는 정의할 수 있는 출력 포트가 표시되어 있습니다.

포트 유형	포트 이름	반환 값
출력	응답	<웹 서비스에서 반환하는 응답>

REST 웹 서비스 소비자 변환 포트

REST 웹 서비스 소비자 변환에는 여러 개의 입력 포트와 여러 개의 출력 포트가 포함될 수 있습니다. XML 또는 JSON 계층의 구조에 따라 포트를 그룹으로 작성합니다.

변환 포트를 볼 때 XML 또는 JSON 계층을 보지 않아도 될 경우에는 포트를 표시합니다. 포트를 표시할 때 그룹을 정의하고 포트를 정의하고 메서드 입력 및 출력의 요소를 입력 및 출력 포트에 매핑할 수 있습니다.

REST 웹 서비스 소비자 변환에는 여러 개의 입력 그룹과 여러 개의 출력 그룹이 포함될 수 있습니다. 포트를 작성할 때 그룹을 작성하고 포트를 그룹에 추가합니다. XML 또는 JSON의 입력 또는 출력 계층 구조에 따라 그룹 계층의 포트를 정의합니다. 키를 추가하여 하위 그룹을 상위 그룹에 연결합니다.

계층의 최하위 그룹을 제외한 모든 그룹에는 기본 키가 있어야 합니다. 루트 그룹을 제외한 계층의 모든 그룹에는 외래 키가 있어야 합니다.

변환에는 RequestInput이라는 루트 입력 그룹이 포함됩니다. 이 루트 입력 그룹에 기본 키를 추가해야 합니다. 키는 문자열, bigint 또는 정수여야 합니다. 루트 입력 그룹의 모든 포트를 통과 포트 구성할 수 있습니다.

요소를 포트에 매핑하려면 위치 열에서 필드를 클릭하고 위치 선택 대화 상자에서 계층을 확장합니다. 그런 다음 계층에서 요소를 선택합니다.

입력 포트

입력 포트는 웹 서비스에 전달할 업스트림 변환 또는 소스의 데이터를 나타냅니다. 여러 입력 포트를 구성할 수 있습니다. 각 입력 포트는 요청 메시지의 요소에 매핑됩니다.

입력 포트를 추가하려면 입력 그룹을 선택하고 새로 만들기 단추 옆의 화살표를 클릭한 다음 필드를 선택합니다.

출력 포트

출력 포트는 다운스트림 변환 또는 대상에 전달할 응답 메시지의 요소를 나타냅니다. 여러 출력 포트를 구성할 수 있습니다. 각 출력 포트는 응답 메시지의 요소에 매핑됩니다.

출력 포트를 추가하려면 출력 그룹을 선택하고 새로 만들기 단추 옆의 화살표를 클릭한 다음 필드를 선택합니다.

통과 포트

통과 포트는 데이터를 변경하지 않고 변환을 통해 데이터를 전달합니다. 루트 입력 그룹의 모든 포트를 통과 포트 구성할 수 있습니다.

통과 포트를 추가하려면 포트를 루트 입력 그룹에 추가합니다. 그런 다음 마우스 오른쪽 단추로 포트를 클릭하고 **매핑**을 선택합니다.

인수 포트

리소스에 대한 URL이 인수를 사용할 경우 인수 포트를 사용하여 리소스를 식별할 수 있습니다. 인수 포트는 루트 입력 그룹에 추가합니다.

인수 포트에는 포트 이름과 인수 이름이 포함됩니다. 인수 이름에 포트 이름으로 허용되지 않는 문자가 포함되는 경우 포트 이름과 다른 인수 이름을 입력합니다. 예를 들어 인수 "Cust-ID"를 웹 서비스에 전달하려고 합니다. 그런데 데이터 통합 서비스에서는 포트 이름에 대시 문자(-)를 사용할 수 없습니다. 이 경우 인수 이름으로 "Cust-ID"를 입력하고 포트 이름으로는 "CustID"를 입력합니다.

그러면 데이터 통합 서비스가 인수 이름과 각 인수 포트의 값을 이름=값 쌍으로 기본 URL에 추가합니다. 여러 인수 포트를 구성할 수 있습니다. 데이터 통합 서비스는 요청의 여러 인수를 앰퍼샌드 문자(&)로 분리합니다.

예:

`http://www.HypoStores.com/customers/CustDetails?Last_Name=Jones&First_Name=Mary`

인수 포트와 URL 포트를 변환에서 정의할 경우 데이터 통합 서비스가 URL 포트 값을 기본 URL에 추가하고 인수 이름과 값을 그 뒤에 추가합니다.

인수 포트를 추가하려면 루트 입력 그룹을 마우스 오른쪽 단추로 클릭하고 **새로 만들기 > 인수 포트**를 선택합니다. 인수 이름과 포트 이름을 입력합니다.

URL 포트

URL 포트를 사용하면 정적 URL을 통해 리소스를 식별할 수 있습니다. 리소스를 식별하기 위해 데이터 통합 서비스에서 URL 포트 값을 기본 URL에 추가합니다.

예:

`http://www.HypoStores.com/products/ProductDetails/<URL_port_value>`

URL 포트를 루트 입력 그룹에 추가합니다.

여러 URL 포트를 구성할 수 있습니다. 데이터 통합 서비스에서는 슬래시(/) 문자를 사용하여 각 URL 포트의 값을 구분합니다. 변환에 URL 포트 및 인수 포트를 정의한 경우 데이터 통합 서비스가 URL 포트 값을 기본 URL에 추가하고, 그 다음에 인수 이름과 값이 나옵니다.

URL 포트를 추가하려면 루트 입력 그룹에서 마우스 오른쪽 단추를 클릭하고 **새로 만들기 > URL 포트**를 선택합니다.

HTTP 헤더 포트

HTTP 헤더 포트는 요청 메시지의 HTTP 헤더를 나타냅니다. 여러 개의 HTTP 헤더 포트를 구성할 수 있습니다.

요청에서 헤더 정보를 웹 서비스에 전달하려면 포트를 루트 입력 그룹에 추가합니다. 루트 입력 그룹에 대해 하나의 HTTP 헤더 포트를 구성할 수 있습니다. 루트 입력 그룹에 HTTP 헤더를 추가하는 경우 통과 포트 구성할 수 있습니다.

HTTP 헤더 포트에는 포트 이름과 HTTP 헤더 이름이 포함됩니다. HTTP 헤더 이름에 포트 이름에서 허용되지 않는 문자가 포함될 경우 포트 이름과 다른 HTTP 헤더 이름을 입력하십시오. 예를 들어 헤더 이름 "Content-

Type"을 웹 서비스에 전달하려고 합니다. 그런데 데이터 통합 서비스에서는 포트 이름에 대시 문자(-)를 사용할 수 없습니다. 이 경우 "Content-Type"을 HTTP 헤더 이름으로 입력하되 포트 이름으로는 "ContentType"을 입력하십시오.

HTTP 헤더 포트를 추가하려면 루트 입력 그룹을 마우스 오른쪽 단추로 클릭하고 **새로 만들기 > HTTP 헤더**를 선택합니다. 헤더 이름 및 포트 이름을 입력합니다.

쿠키 포트

쿠키 인증을 사용하도록 REST 웹 서비스 소비자 변환을 구성할 수 있습니다. 원격 웹 서비스 서버에서는 쿠키를 기반으로 웹 서비스 소비자 사용자를 추적합니다. 매핑이 웹 서비스를 여러 번 호출하는 경우 성능을 향상시킬 수 있습니다.

쿠키 정보를 요청의 웹 서비스로 전달하려면 포트를 루트 입력 그룹에 추가합니다. 루트 입력 그룹에 대해 하나의 쿠키 포트를 구성할 수 있습니다. 쿠키 포트를 루트 입력 그룹에 추가할 경우 해당 쿠키 포트를 통과 포트로 구성할 수 있습니다.

응답에서 쿠키 정보를 추출하려면 쿠키 포트를 출력 그룹에 추가합니다. 각 출력 그룹에 대해 하나의 쿠키 포트를 구성할 수 있습니다.

쿠키 포트를 웹 서비스 요청 메시지에 연결하면 웹 서비스 공급자가 응답 메시지에 쿠키 값을 반환합니다. 쿠키 값을 매핑의 다른 변환 다운스트림에 전달하거나 파일에 쿠키 값을 저장할 수 있습니다. 쿠키 값을 파일에 저장할 경우 쿠키를 REST 웹 서비스 소비자 변환에 대한 입력으로 구성할 수 있습니다.

쿠키 포트를 추가하려면 루트 입력 그룹을 마우스 오른쪽 단추로 클릭하고 **새로 만들기 > 기타 포트**를 선택합니다. 그런 다음 **쿠키**를 선택하고 **확인**을 클릭합니다.

출력 XML 포트

출력 XML 포트는 웹 서비스의 응답을 나타냅니다. 출력 XML 포트는 문자열 포트입니다.

출력 XML 포트를 출력 그룹에 추가합니다. 각 출력 그룹에 대해 하나의 출력 XML 포트를 구성할 수 있습니다.

루트 입력 그룹을 마우스 오른쪽 단추로 클릭하고 **새로 만들기 > 기타 포트**를 선택합니다. 그런 다음 **출력 XML**을 선택하고 **확인**을 클릭합니다.

응답 코드 포트

응답 코드 포트는 웹 서비스의 HTTP 응답 코드를 나타냅니다. 응답 코드 포트는 정수 포트입니다.

응답 코드 포트는 출력 그룹에 추가합니다. 각 출력 그룹에 대해 하나의 응답 코드 포트를 구성할 수 있습니다.

응답 코드 포트를 추가하려면 출력 그룹을 선택하고 루트 입력 그룹을 마우스 오른쪽 단추로 클릭한 다음 **새로 만들기 > 기타 포트**를 선택합니다. 그런 다음 **응답 코드**를 선택하고 **확인**을 클릭합니다.

REST 웹 서비스 소비자 변환 입력 매핑

변환 포트를 볼 때 메서드 입력 계층을 보려면 입력 매핑을 표시합니다. 입력 매핑을 표시하면 입력 그룹을 정의하고 입력 포트를 정의하고 입력 포트를 메서드 입력 요소에 매핑할 수 있습니다.

입력 매핑에는 다음 영역이 포함됩니다.

포트

포트 영역에서 변환 입력 그룹 및 입력 포트를 작성합니다.

메서드 입력

메서드 입력 영역에는 REST 웹 서비스 소비자 변환이 웹 서비스에 전송하는 요청 메시지의 요소가 표시됩니다. 스키마 개체를 사용하여 변환을 작성하는 경우 스키마 개체가 메서드 입력 계층을 정의합니다.

입력 포트를 작성한 후 **포트** 영역의 입력 포트를 **메서드 입력** 영역의 요소에 매핑합니다. 입력 포트를 메서드 입력의 요소에 매핑하면 포트의 위치가 **메서드 입력** 영역의 위치 열에 나타납니다.

입력 계층의 첫 번째 수준을 매핑하기로 선택한 경우 **Developer tool**이 메서드 입력의 첫 번째 수준에 있는 노드를 입력 포트에 매핑합니다. **Developer tool**은 매핑을 수행할 포트도 작성합니다. 하나 이상의 다중 발생 하위 요소가 있는 다중 발생 상위 요소가 계층의 첫 번째 수준에 포함될 경우 **Developer tool**이 계층의 첫 번째 수준을 매핑하지 않습니다.

입력 포트를 메서드 입력의 요소에 연결하는 행을 표시하도록 선택할 수 있습니다.

입력 포트를 요소에 매핑하기 위한 규칙 및 지침

입력 포트를 메서드 입력 계층의 요소에 매핑하는 경우 다음 규칙을 검토합니다.

- 입력 포트는 계층의 한 요소에 매핑할 수 있습니다. 동일한 포트는 계층의 여러 키에 매핑할 수 있습니다.
- 입력 포트 및 요소의 데이터 유형은 호환되어야 합니다.
- 한 입력 그룹의 포트를 메서드 입력의 여러 계층 수준에 매핑할 수 있습니다.
- 입력 포트를 메서드 입력의 키에 매핑해야 합니다. 키에 매핑하는 포트는 문자열, 정수 또는 **bigint** 데이터 유형이어야 합니다. 요청 메시지에 포함하는 계층 수준 이상의 메서드 입력에 있는 모든 수준의 키에 데이터를 매핑하십시오. 매핑하는 수준을 포함하여 그 이상의 모든 수준에 대한 외래 키를 포함하십시오.

참고: 메서드 입력 계층의 가장 낮은 수준만 매핑할 경우 입력 포트를 키에 매핑할 필요가 없습니다.

- **RequestInput** 루트 요소는 메서드 입력 정의에 대한 **Rest_Consumer_input** 그룹의 하위 요소에 매핑해야 합니다.
- 여러 문자열, **bigint** 또는 정수 입력 포트를 **메서드 입력** 영역의 키에 매핑하여 복합 키를 작성할 수 있습니다. 복합 키의 **위치** 필드를 클릭하면 입력 포트의 순서를 다시 정렬하거나 포트 중 하나를 제거할 수 있습니다.
- 웹 서비스가 **JSON** 문서를 생성할 경우 응답 계층의 첫 번째 노드는 **xmlRoot**여야 합니다. **JSON** 응답이 포함된 웹 서비스의 첫 번째 노드가 **xmlRoot**가 아닐 경우 **null** 값이 나타날 수 있습니다.

메서드 입력에 입력 포트 매핑

변환 입력 매핑을 표시할 때 입력 그룹을 정의하고 입력 포트를 정의하고 입력 포트를 메서드 입력 요소에 매핑할 수 있습니다.

1. REST 웹 서비스 소비자 변환을 엽니다.
2. **포트** 보기에서 입력 매핑을 표시합니다.
3. 루트 입력 그룹에 대한 기본 키를 정의합니다.
4. 입력 그룹 또는 포트를 **포트** 영역에 추가하려면 다음 방법 중 하나를 사용합니다.

방법	설명
요소를 끌어서 놓습니다.	메서드 입력 영역에서 그룹 또는 하위 요소를 포트 영역의 빈 열로 끌어서 놓습니다. 그룹을 포트 영역으로 끌 경우 Developer tool 이 포트 없이 그룹을 추가합니다.
그룹 또는 포트를 수동으로 추가합니다.	그룹을 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 그룹 을 클릭합니다. 포트를 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 필드 를 클릭합니다.

방법	설명
다른 변환에서 포트를 끌어서 놓습니다.	편집기에서 다른 변환의 포트를 REST 웹 서비스 소비자 변환으로 끌어서 놓습니다.
포트를 복사합니다.	다른 변환에서 포트를 선택하여 포트 영역에 복사합니다. 포트를 복사하기 위해 키보드 바로 가기를 사용하거나 Developer tool의 복사 및 붙여넣기 단추를 사용할 수 있습니다.
계층의 첫 번째 수준 매핑을 선택합니다.	Developer tool이 메서드 입력의 첫 번째 수준에 있는 요소를 입력 포트 및 그룹에 매핑합니다. Developer tool은 매핑을 수행할 입력 포트 및 그룹도 작성합니다.

- 수동으로 포트를 작성하거나 다른 변환의 포트를 복사하는 경우 **메서드 입력** 영역에서 **위치** 열을 클릭하고 목록에서 포트를 선택합니다.
- 입력 포트를 복합 키로 매핑하려면 다음 방법 중 하나를 사용하십시오.

방법	설명
입력 포트를 끌어서 놓습니다.	두 개 이상의 입력 포트를 선택하고 메서드 입력 계층의 키로 끌어서 놓습니다.
위치 선택 대화 상자에서 입력 포트를 선택합니다.	메서드 입력 계층에 있는 키의 위치 열을 클릭한 다음 입력 포트를 선택합니다.

- 요소의 위치를 지우려면 다음 방법 중 하나를 사용합니다.

방법	설명
지우기 를 클릭합니다.	메서드 입력 영역에서 하나 이상의 요소를 선택하고 지우기 를 클릭합니다.
포트를 요소에 연결하는 행을 삭제합니다.	입력 포트를 메서드 입력의 요소에 연결하는 하나 이상의 행을 선택하고 삭제 를 누릅니다.

REST 웹 서비스 소비자 변환 출력 매핑

변환 포트를 볼 때 메서드 출력 계층을 보려면 출력 매핑을 표시합니다. 출력 매핑을 표시하면 출력 그룹을 정의하고 출력 포트를 정의하고 메서드 출력 요소를 출력 포트에 매핑할 수 있습니다.

출력 매핑에는 다음 영역이 포함됩니다.

메서드 출력

메서드 출력 영역에는 웹 서비스가 REST 웹 서비스 소비자 변환에 반환하는 응답 메시지의 요소가 표시됩니다. 스키마 개체를 사용하여 변환을 작성하는 경우 스키마 개체가 메서드 출력 계층을 정의합니다.

포트

포트 영역에서 변환 출력 그룹 및 포트를 작성합니다.

출력 포트를 작성한 후 **메서드 출력**의 요소를 **포트** 영역의 포트에 매핑합니다. 메서드 출력의 요소를 출력 포트에 매핑하면 요소의 위치가 **포트** 영역의 **위치** 열에 나타납니다.

출력 계층의 첫 번째 수준을 매핑하기로 선택한 경우 Developer tool이 메서드 출력의 첫 번째 수준에 있는 요소를 출력 포트에 매핑합니다. Developer tool은 매핑을 수행할 포트도 작성합니다. 하나 이상의 다중 발생 하위

요소가 있는 다중 발생 상위 요소가 계층의 첫 번째 수준에 포함될 경우 **Developer tool**이 계층의 첫 번째 수준을 매핑하지 않습니다.

계층에 출력 포트를 표시하도록 선택할 수 있습니다. 각 하위 그룹은 상위 그룹 아래에 나타납니다. 메서드 출력의 요소를 출력 포트에 연결하는 행을 표시하도록 선택할 수도 있습니다.

연결된 스키마 개체가 리포지토리에서 삭제된 경우 출력 매핑의 메서드 요소에 대한 위치는 **Developer tool**에 유지됩니다. 출력 매핑을 표시할 때 메서드 요소의 위치는 여전히 **포트** 영역에서 출력 포트에 대한 **위치** 열에 표시됩니다. 다른 스키마를 변환에 연결할 경우 **Developer tool**이 각 위치가 올바른지 확인합니다. 위치가 더 이상 올바르지 않은 경우 **Developer tool**이 출력 매핑의 **포트** 영역에서 메서드 요소의 위치를 지웁니다.

요소를 출력 포트에 매핑하기 위한 규칙 및 지침

메서드 출력 계층의 요소를 출력 포트에 매핑하는 경우 다음 규칙을 검토합니다.

- 메서드 출력 요소 및 출력 포트의 데이터 유형은 호환되어야 합니다.
- 그룹의 출력 포트 2개 이상에 요소를 매핑할 수 없습니다.
- 통과 포트를 제외한 모든 출력 포트에는 올바른 위치가 있어야 합니다.
- 여러 번 발생하는 하위 요소를 빈 출력 포트에 끝 경우 그룹을 다른 출력 그룹에 연결해야 합니다. 그룹을 선택하면 **Developer tool**이 그룹을 연결하는 키를 작성합니다.
- 여러 번 발생하는 요소를 상위 요소가 포함된 그룹으로 끝면 포함할 하위 요소 발생 수를 구성할 수 있습니다. 또는 상위 그룹을 변환 출력에서 여러 번 발생하는 하위 그룹으로 대체할 수 있습니다.
- 웹 서비스가 **JSON** 문서를 생성할 경우 응답 계층의 첫 번째 노드는 **xmlRoot**여야 합니다. **JSON** 응답이 포함된 웹 서비스의 첫 번째 노드가 **xmlRoot**가 아닐 경우 출력 포트에 **null** 값이 나타날 수 있습니다.

보기 사용자 지정의 옵션

메서드 출력 영역에 쿠키 포트, 통과 포트 및 키가 표시되도록 메서드 출력 계층을 변경할 수 있습니다. 요소 정렬 방법을 정의하는 그룹화 구성을 표시할 수도 있습니다.

메서드 출력 영역에서 **보기 사용자 지정**을 클릭합니다. 다음 옵션을 활성화합니다.

시퀀스, 선택 및 모두

요소 정의가 시퀀스, 선택 또는 모두인지 여부를 나타내는 행을 표시합니다.

시퀀스 그룹의 요소는 계층에 지정된 순서여야 합니다.

선택 그룹의 요소 하나 이상이 응답 메시지에 표시되어야 합니다.

모든 그룹의 요소가 응답 메시지에 포함되어야 합니다.

키

메서드 출력 영역에서 키를 봅니다. **메서드 출력** 영역에는 각 그룹의 키가 포함됩니다. **포트** 영역의 출력 포트에 키를 추가할 수 있습니다.

통과 포트

메서드 출력 영역에 통과 포트가 표시됩니다. 통과 포트는 데이터를 변경하지 않고 변환을 통해 데이터를 전달하는 포트입니다. 메서드 출력의 통과 포트를 모든 **REST** 웹 서비스 소비자 변환 출력 그룹에 연결할 수 있습니다. 통과 포트는 데이터를 한 번 수신하므로 응답 메시지의 루트 수준에 위치합니다.

출력 포트에 메서드 출력 매핑

변환 출력 매핑을 표시할 때 출력 그룹을 정의하고 출력 포트를 정의하고 메서드 출력 요소를 출력 포트에 매핑할 수 있습니다.

1. REST 웹 서비스 소비자 변환을 엽니다.
2. **포트** 보기에서 출력 매핑을 표시합니다.
3. 출력 그룹 또는 포트를 **포트** 영역에 추가하려면 다음 방법 중 하나를 사용합니다.

방법	설명
요소를 끌어서 놓습니다.	메서드 출력 영역에서 그룹 또는 하위 요소를 포트 영역의 빈 열로 끌어서 놓습니다. 그룹을 포트 영역으로 끌 경우 Developer tool이 포트 없이 그룹을 추가합니다.
그룹 또는 포트를 수동으로 추가합니다.	그룹을 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 그룹 을 클릭합니다. 포트를 추가하려면 새로 만들기 단추 옆의 화살표를 클릭한 다음 필드 를 클릭합니다.
다른 변환에서 포트를 끌어서 놓습니다.	편집기에서 다른 변환의 포트를 REST 웹 서비스 소비자 변환으로 끌어서 놓습니다.
포트를 복사합니다.	다른 변환에서 포트를 선택하여 포트 영역에 복사합니다. 포트를 복사하기 위해 키보드 바로 가기를 사용하거나 Developer tool의 복사 및 붙여넣기 단추를 사용할 수 있습니다.

4. 포트를 수동으로 작성하거나 다른 변환의 포트를 복사하는 경우 **포트** 영역에서 **위치** 열을 클릭하고 목록에서 요소를 선택합니다.
5. 포트의 위치를 지우려면 다음 방법 중 하나를 사용합니다.

방법	설명
지우기 를 클릭합니다.	포트 영역에서 하나 이상의 포트를 선택하고 지우기 를 클릭합니다.
요소를 포트에 연결하는 행을 삭제합니다.	메서드 출력의 요소를 출력 포트에 연결하는 하나 이상의 행을 선택하고 삭제 를 누릅니다.

REST 웹 서비스 소비자 변환의 고급 속성

데이터 통합 서비스가 REST 웹 서비스 소비자 변환의 데이터를 처리하는 방법을 결정하는 속성을 구성할 수 있습니다.

고급 탭에서 다음 속성을 구성합니다.

추적 수준

이 변환에 대해 로그에 표시되는 세부 정보의 양입니다. 간단, 보통, 자세한 정보 표시 초기화 또는 자세한 정보 표시 데이터를 선택할 수 있습니다. 기본값은 보통입니다.

연결

웹 서비스에 연결할 HTTP 연결 개체를 식별합니다. HTTP 연결은 Developer tool에서 작성하고 편집합니다. HTTP 연결을 구성할 때 기본 URL, 웹 서비스에 필요한 보안 유형 및 연결 제한 시간 기간을 구성합니다.

REST 웹 서비스 소비자 변환은 URL을 사용하여 웹 서비스에 연결합니다. 이 URL은 변환 속성 또는 HTTP 연결에서 정의할 수 있습니다.

HTTP 연결은 다음과 같은 상황에서 구성합니다.

- URL 입력 포트를 사용하지 않습니다.
- 웹 서비스에 HTTP 인증 또는 SSL 인증서가 필요합니다.
- 기본 연결 제한 시간 기간을 변경해야 합니다.

XML 스키마 유효성 검사

런타임 시 응답 메시지의 유효성을 검사합니다. **올바르지 않은 XML에 대한 오류** 또는 **유효성 검사 없음**을 선택합니다.

정렬된 입력

데이터 통합 서비스에서 모든 입력 데이터를 처리하지 않고도 출력을 생성할 수 있습니다. 입력 데이터를 XML 입력 계층의 키별로 정렬한 경우 정렬된 입력을 활성화합니다.

URL

REST 웹 서비스의 기본 URL입니다. 이 값은 HTTP 연결의 기본 URL로 재정의됩니다.

형식

웹 서비스 응답의 형식입니다. 웹 서비스 응답에 따라 **XML** 또는 **JSON**을 선택합니다.

REST 웹 서비스 소비자 변환 작성

재사용 가능하거나 재사용 불가능한 REST 웹 서비스 소비자 변환을 작성할 수 있습니다. 재사용 가능 변환은 여러 매핑에 존재할 수 있습니다. 재사용 불가능 변환은 단일 매핑에만 존재합니다.

REST 웹 서비스 소비자 변환을 작성할 때 요소 및 XML 계층을 수동으로 정의하거나 스키마 개체에서 요소 및 계층을 가져올 수 있습니다. 스키마 개체는 XML 파일 또는 텍스트 파일이 될 수 있습니다.

REST 웹 서비스 소비자 변환 작성

REST 웹 서비스 소비자 변환을 작성할 때는 메서드를 선택하고 선택한 메서드에 따라 메서드 입력 및 메서드 출력을 정의합니다.

1. REST 웹 서비스 소비자 변환을 작성하려면 다음 방법 중 하나를 사용합니다.

방법	설명
재사용 가능	개체 탐색기 보기에서 프로젝트나 폴더를 선택합니다. 파일 > 새로 만들기 > 변환 을 클릭합니다. REST 웹 서비스 소비자 변환을 선택하고 다음 을 클릭합니다.
재사용 불가능	매핑 또는 맵렛에서 변환 색상표의 REST 웹 서비스 소비자 변환을 매핑 또는 맵렛 편집기로 끌어서 놓습니다.

2. 변환 이름을 입력하고 위치 및 HTTP 메서드를 선택합니다.
3. **다음**을 클릭합니다.

- 메서드 입력을 정의하려면 다음 방법 중 하나를 사용합니다.

방법	설명
빈 항목으로 생성	XML 요소 및 계층을 수동으로 정의합니다.
스키마 개체의 요소에서 작성	스키마 개체에서 XML 요소 및 계층을 가져옵니다.

메서드 입력 정의 영역에 변환 입력 그룹 및 입력 포트가 표시됩니다. **입력 매핑** 영역에 요청 메시지 계층이 표시됩니다.

- 입력 그룹 및 입력 포트를 정의하고 입력 포트를 입력 요소로 매핑합니다.
- 다음을 클릭합니다.
- 메서드 출력을 정의하려면 **빈 항목으로 작성** 또는 **스키마 개체의 요소에서 작성**을 선택합니다.
메서드 출력 정의 영역에 변환 입력 그룹 및 입력 포트가 표시됩니다. **출력 매핑** 영역에 요청 메시지 계층이 표시됩니다.
- 출력 그룹 및 출력 포트를 정의하고 요소를 출력 포트에 매핑합니다.
- 마침을 클릭합니다.

배열이 포함된 JSON 응답 메시지 구문 분석

요소가 복합 유형의 하위 요소이고 이 요소의 최대 발생 수가 바인딩되지 않음인 경우 스키마가 올바르지 않습니다. JSON 파서는 요소의 여러 인스턴스를 추출하는 것을 제한합니다.

복합 유형에서 하위 요소의 최대 발생 수는 0 또는 1이어야 하고 스키마의 복합 유형에 대해 순서 표시기를 선택해야 합니다. 스키마 유효성 검사를 위해 최대 발생 수를 1로 변경하는 경우 한 번에 하나의 요소 인스턴스를 추출할 수 있습니다.

스키마의 복합 유형 선택 순서 표시기에서 최대 발생 수를 바인딩되지 않음으로 사용할 수 있습니다.

JSON 응답 메시지 예제

복합 유형 요소 `xmlRoot`의 요소 이름이 `Likes`이고 최대 발생 수가 바인딩되지 않은 다음과 같은 스키마가 있습니다.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://
www.w3.org/2001/XMLSchema">
  <xs:element name="xmlRoot">
    <xs:complexType>
      <xs:all>
        <xs:element type="xs:byte" name="Age"/>
        <xs:element type="xs:string" name="FirstName"/>
        <xs:element type="xs:string" name="Likes" maxOccurs="unbounded" minOccurs="0"/>
        <xs:element type="xs:string" name="FamilyName"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

이 경우 다음과 같은 형식으로 JSON 응답을 변경할 수 있습니다.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://
www.w3.org/2001/XMLSchema">
  <xs:element name="xmlRoot">
    <xs:complexType>
```

```

<xs:choice maxOccurs="unbounded">
  <xs:element type="xs:byte" name="Age"/>
  <xs:element type="xs:string" name="FirstName"/>
  <xs:element type="xs:string" name="Likes" />
  <xs:element type="xs:string" name="FamilyName"/>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

<xs:choice maxOccurs="unbounded">는 콘텐츠를 한 번 이상 원하는 순서로 반복할 수 있습니다.

응답 메시지의 명명되지 않은 배열

REST 웹 서비스 소비자 변환은 명명되지 않은 배열을 요청 메시지가 아닌 응답 메시지에서만 지원합니다. 메서드 출력 정의에 정의된 명명되지 않은 배열 스키마를 구문 분석하려면 **complexType**의 상위 요소 또는 단순 유형 배열 요소 이름이 **xmlRoot**여야 합니다.

REST 웹 서비스 소비자 변환에서는 **xmlRoot**를 최대 발생이 바운드되지 않음으로 설정된 **xmlRoot** 요소의 하위 요소로 정의하고 명명되지 않은 배열의 요소를 **xmlRoot** 요소의 하위 요소로 정의해야 합니다.

다음 이미지는 명명되지 않은 배열에 대해 정의된 메서드 출력을 보여 줍니다.

☐ Ports ☐ Method input ☒ Method output

Show: ☒ Method output definition ☐ Output mapping

Method output definition

Name	Type	Min...	Ma...	Description	>>
Rest_Consume...	(Rest_Cons...				
xmlRoot	(xmlRoot)	1	1		
xmlRoot	(xmlRoot)	1	Un...		
emp	xs:string	1	1		
empid	xs:string	1	1		

제 13 장

REST 웹 서비스 소비자 변환 사용 사례

이 장에 포함된 항목:

- [REST 웹 서비스 소비자 변환을 구성하여 데이터를 수신하는 방법, 153](#)
- [스키마를 사용하여 REST 웹 서비스 소비자 변환을 생성하는 방법, 159](#)
- [REST 웹 서비스 소비자 변환에서 사용자 지정 URL을 생성하는 방법, 168](#)

REST 웹 서비스 소비자 변환을 구성하여 데이터를 수신하는 방법

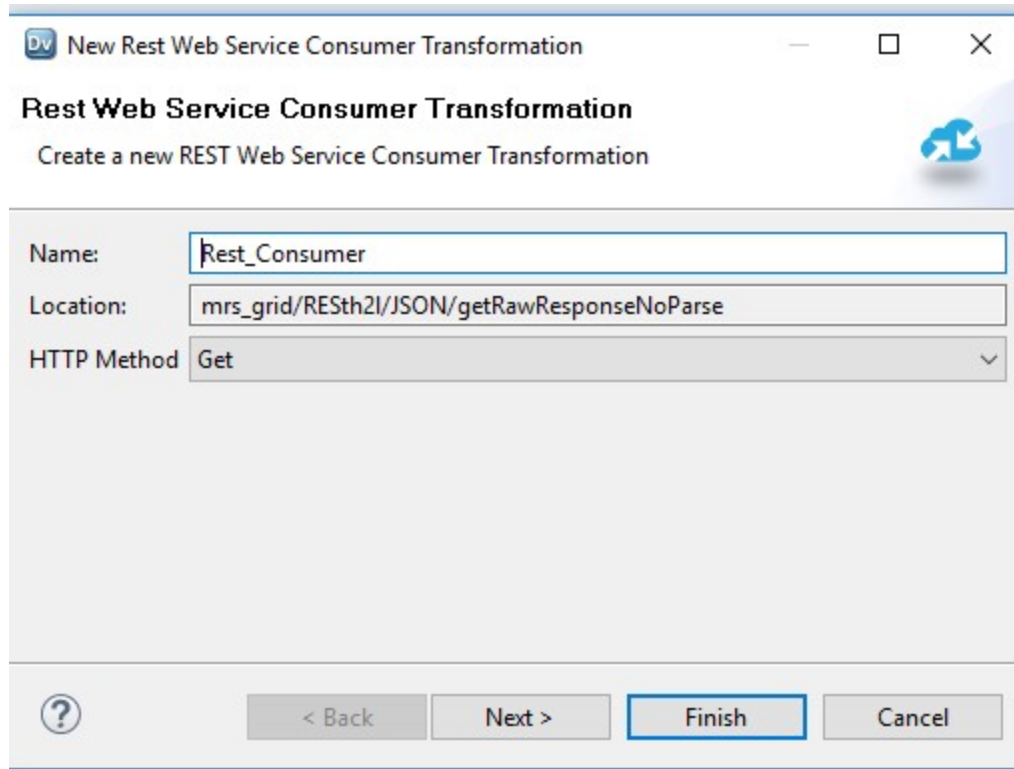
REST 웹 서비스 소비자 변환을 사용하여 구문 분석되지 않았거나 구문 분석된 데이터를 웹 서비스로부터 수신할 수 있습니다. 응답 메시지를 지정하는 경우 원래 메시지를 수신하도록 선택하거나 미리 정의된 필드를 지정하여 메시지를 구문 분석하고 메시지 요소를 나타내도록 선택할 수 있습니다.

웹 서비스에서 구문 분석되지 않은 데이터를 수신하는 방법

다음 단계를 완료하여 변환을 생성합니다.

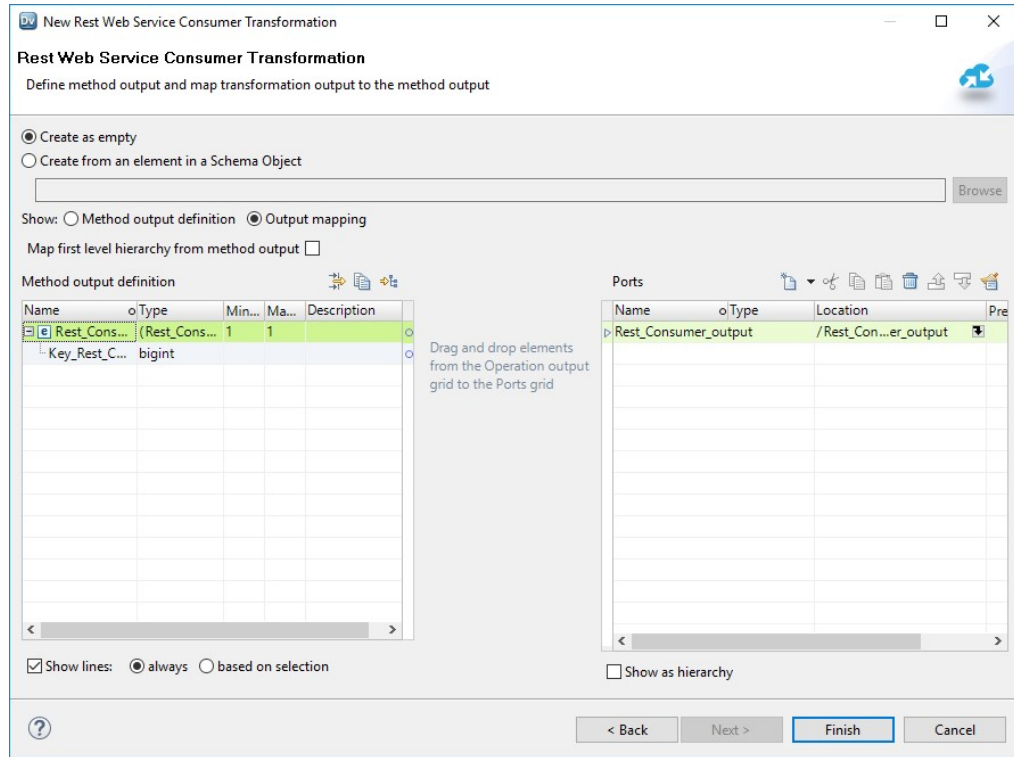
1. Developer tool 매핑 편집기에서 매핑 팔레트를 아래로 스크롤하여 REST 웹 서비스 소비자 변환을 찾고 맵핑으로 끌어서 놓습니다.

새 **REST 웹 서비스 소비자 변환** 마법사가 나타납니다.

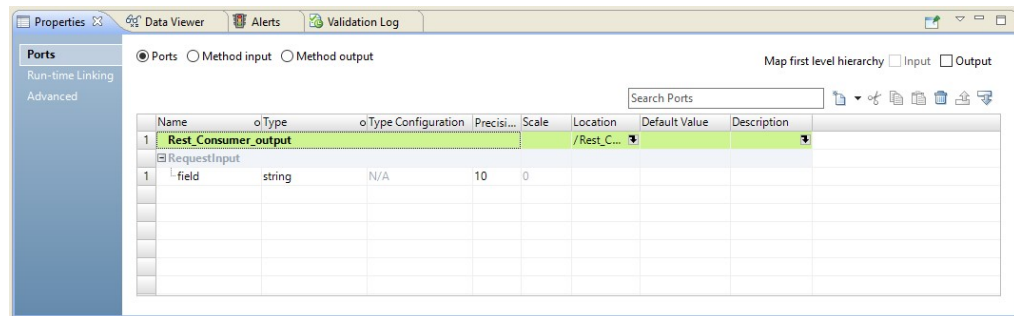


2. 변환 이름을 입력합니다.
3. **HTTP 메서드** 옆에서 **Get**을 선택합니다.
Get 메서드를 사용하는 경우 입력 매핑을 구성하지 않습니다.
4. **다음**을 클릭합니다.
마법사에 Get 메서드를 사용하는 경우 입력이 적용되지 않는 메시지가 표시됩니다.
5. **다음**을 클릭합니다.
6. **빈 항목으로 생성**을 선택합니다.
7. **표시** 옆에서 **출력 매핑**을 선택합니다.

다음 이미지는 마법사의 옵션을 보여 줍니다.

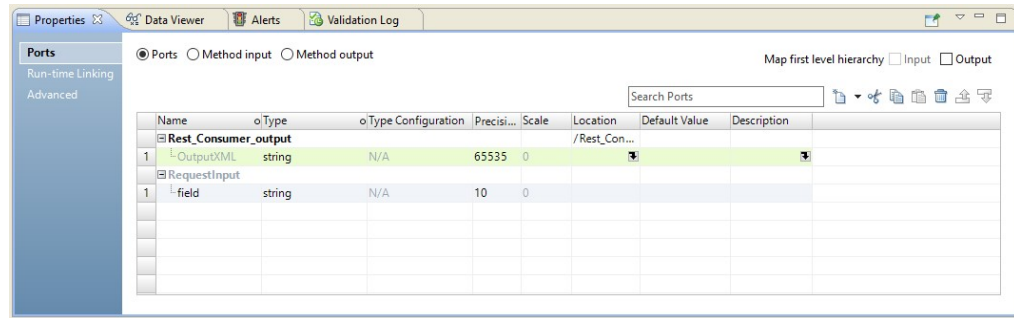


8. 왼쪽에 있는 **메서드 출력 정의** 영역의 출력 그룹을 오른쪽의 **포트** 영역으로 끌어서 놓습니다.
9. **마침**을 클릭합니다.
10. REST 웹 서비스 소비자 변환의 **포트** 탭으로 이동합니다.
11. 출력 그룹을 강조 표시합니다.



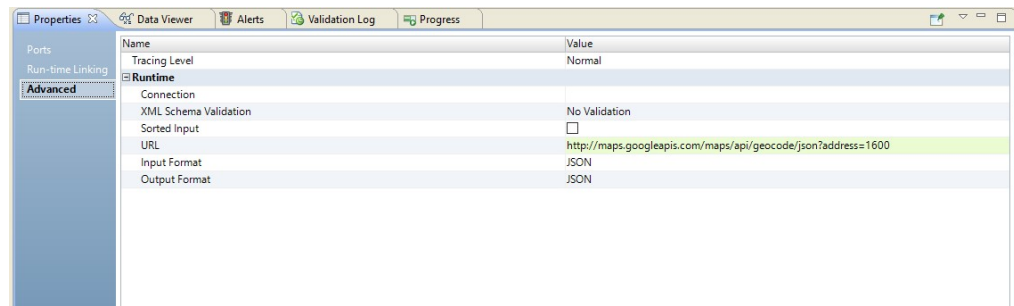
12. **포트** 탭의 오른쪽 위에서 **새로 만들기**를 선택하고 **다른 포트**를 선택합니다. **출력 XML** 옆의 확인란을 선택하고 **확인**을 클릭합니다.

다음 이미지에 표시된 것과 같이 출력 XML 포트가 출력 그룹 아래 중첩됩니다.



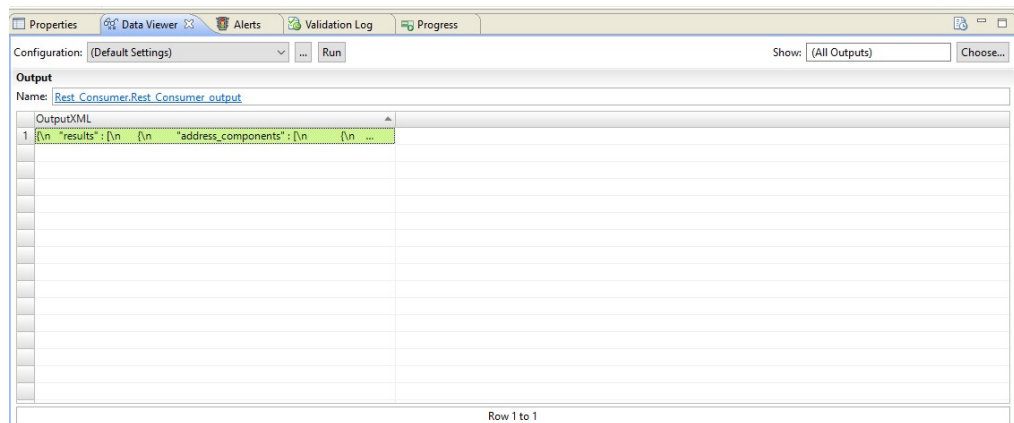
13. 고급 탭으로 이동합니다.
14. 웹 서비스에 연결하려면 기본 URL로 URL 속성을 구성합니다.
15. HTTP 연결을 사용하려면 **연결** 속성에 연결 이름을 입력합니다. HTTP 연결은 URL 속성에 구성된 기본 URL을 재정의합니다.
16. 요청 및 응답 메시지의 형식을 입력합니다. **입력 형식** 및 **출력 형식**을 XML 또는 JSON으로 구성합니다.

다음 이미지는 고급 속성을 구성하는 위치를 보여 줍니다.



17. 데이터 뷰어를 실행하거나 매핑을 실행합니다.

다음 이미지는 출력 XML 포트에 표시되는 응답의 예를 보여 줍니다.



웹 서비스에서 구문 분석된 데이터를 수신하는 방법

웹 서비스에서 수신하는 메시지를 구문 분석할 수 있습니다. 메시지를 구문 분석하면 REST 웹 서비스 소비자 변환이 메서드 출력 정의의 응답 메시지에 정의된 구조에 따라 데이터를 수신합니다.

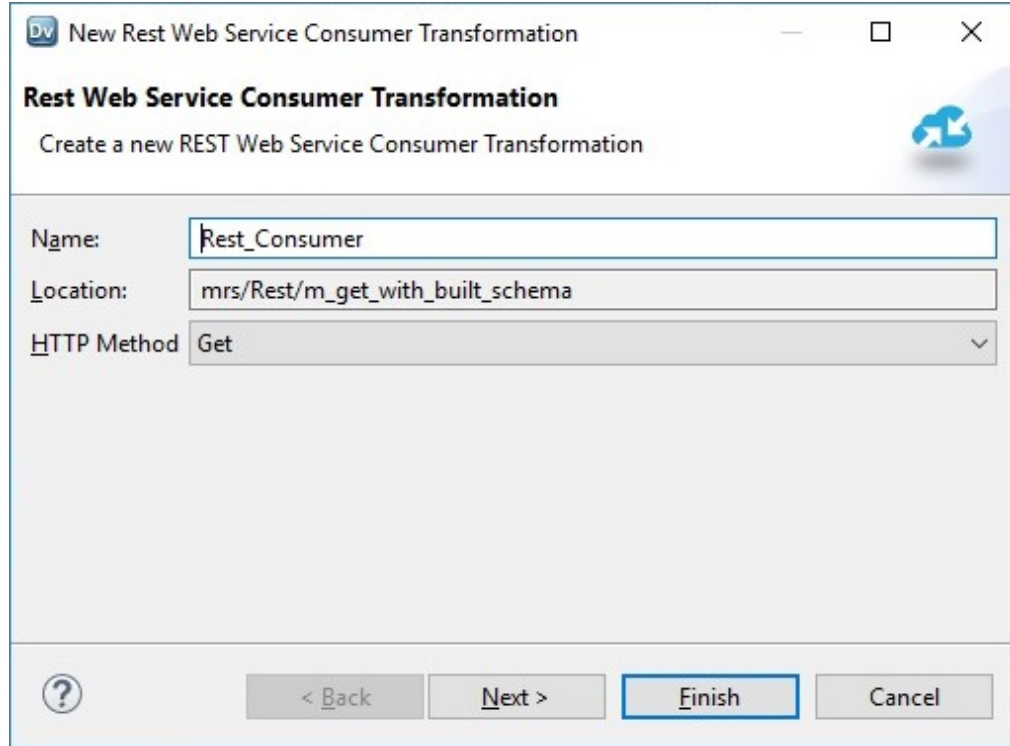
예를 들어 다음 JSON 응답 메시지를 구문 분석할 수 있습니다.

```
{"name": "infa", "message": "hello"}
```

이 응답 메시지를 구문 분석하려면 변환을 구성할 때 다음 단계를 완료합니다.

1. Developer tool 매핑 편집기에서 매핑 팔레트를 아래로 스크롤하여 REST 웹 서비스 소비자 변환을 찾고 매핑으로 끌어서 놓습니다.

새 REST 웹 서비스 소비자 변환 마법사가 나타납니다.



The image shows a dialog box titled "New Rest Web Service Consumer Transformation". It has a subtitle "Rest Web Service Consumer Transformation" and a description "Create a new REST Web Service Consumer Transformation". The dialog box contains three input fields: "Name:" with the value "Rest_Consumer", "Location:" with the value "mrs/Rest/m_get_with_built_schema", and "HTTP Method:" with a dropdown menu showing "Get". At the bottom, there are four buttons: a question mark icon, "< Back", "Next >", and "Finish" (which is highlighted with a blue border), and a "Cancel" button.

2. 변환 이름을 입력합니다.
3. HTTP 메서드 옆에서 Get을 선택합니다.
Get 메서드를 사용하는 경우 입력 매핑을 구성하지 않습니다.
4. 다음을 클릭합니다.
마법사에 Get 메서드를 사용하는 경우 입력이 적용되지 않는 메시지가 표시됩니다.
5. 다음을 클릭합니다.
6. 빈 항목으로 생성을 선택합니다.
7. 표시 옆에서 출력 매핑을 선택합니다.
8. 수신하려는 응답 메시지에 따라 메서드 출력 정의를 설계합니다.

다음 이미지는 메서드 출력 정의가 나타나는 방식을 보여 줍니다.

☐ Ports
 ☐ Method input
 ☒ Method output

Show: ☒ Method output definition
 ☐ Output mapping

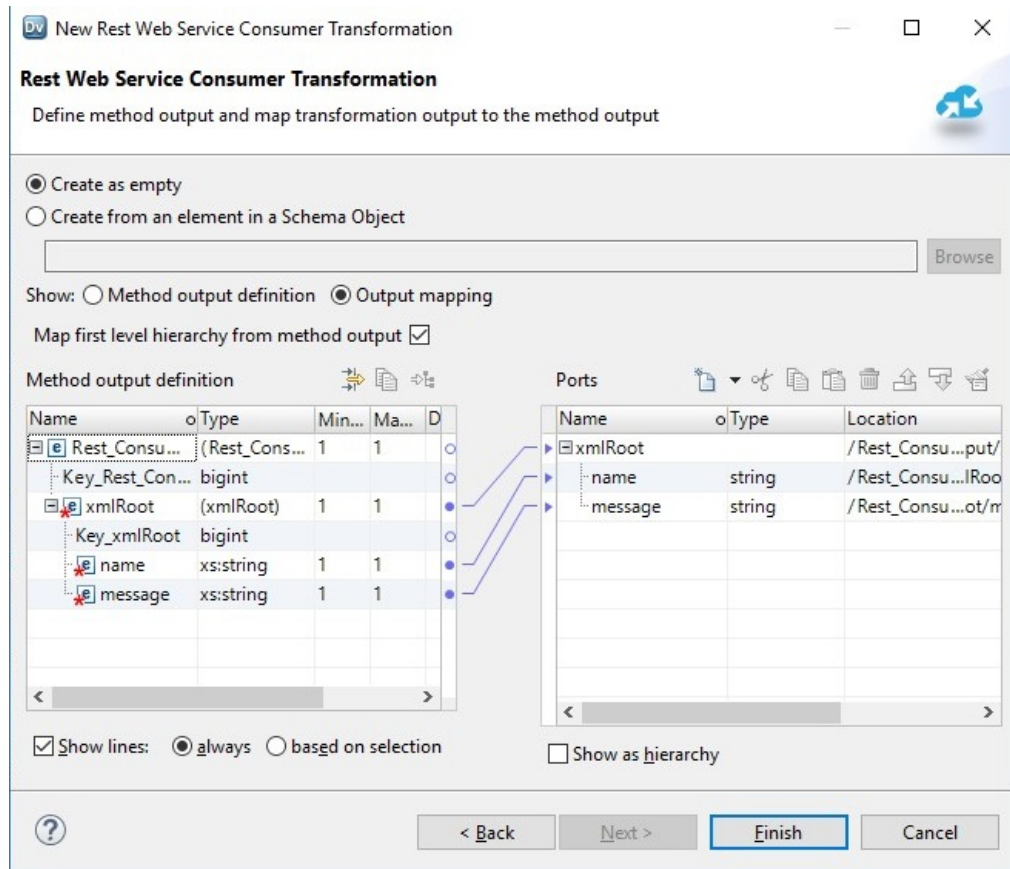
Method output definition

Name	Type	Min...	Ma...	Description	>>
Rest_Consumer_output	(Rest_Cons...				
xmlRoot	(xmlRoot)	1	1		
name	xs:string	1	1		
message	xs:string	1	1		

변환은 웹 서비스에서 JSON 응답을 수신하지만 데이터 통합 서비스는 xmlRoot라는 루트 요소를 JSON 응답에 자동으로 추가합니다. JSON 응답에 대한 메서드 출력 정의를 구성하는 경우 동일한 루트 요소인 xmlRoot를 구성합니다.

9. **출력 매핑**을 선택합니다.
10. **메서드 출력 정의** 영역의 포트를 **포트** 영역으로 끌어 출력 매핑을 구성합니다.

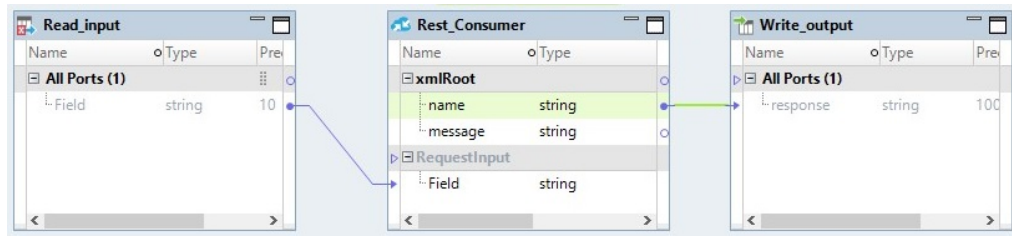
다음 이미지는 출력 매핑의 예제를 보여 줍니다.



11. 마침을 클릭합니다.
12. 변환을 선택하고 고급 탭으로 이동합니다.

13. 출력 형식을 JSON으로 구성합니다.

다음 이미지는 변환이 매핑에서 통합될 때 나타나는 방식을 보여 줍니다.



다음 이미지는 매핑을 실행할 때 나타날 수 있는 응답을 보여 줍니다.

Properties Data Viewer Alerts Validation Log			
Configuration: (Default Settings) ... Run ⓘ All rows are selected for preview.			
Output			
Name: Rest_Consumer.xmlRoot			
	name	message	
1	infa	hello	

스키마를 사용하여 REST 웹 서비스 소비자 변환을 생성하는 방법

변환에 수동으로 구성하기가 어려운 요소가 다수 포함되는 경우 스키마를 사용하여 REST 웹 서비스 소비자 변환을 생성합니다. 변환의 요청 및 응답 메시지에 포함하려는 모든 요소가 포함된 스키마를 가져옵니다.

XML 형식의 요청 및 응답 메시지를 처리하는 방법

REST 웹 서비스 소비자 변환에서 웹 서비스로 전송하는 요청 메시지를 생성할 때 사용할 요소 계층이 포함된 XML 파일이 있습니다. 이 XML 파일에는 조직의 개인을 식별할 때 사용할 수 있는 데이터(예: ID, 이름, 이메일 및 국가)가 포함되어 있습니다. 이러한 요소는 XML 파일에서 다음과 같이 나타낼 수 있습니다.

```
<Person><ID>sample</ID><Name>sample</Name><Email>sample</Email><Country>sample</Country></Person>
```

다른 XML 파일에는 응답 메시지를 생성할 때 사용할 요소 계층이 포함되어 있습니다. 이 XML 파일에는 각 개인을 대할 때 사용할 인사말 유형에 대한 데이터가 포함되어 있습니다. 인사말에는 개인의 ID 및 인사말 콘텐츠에 대한 데이터가 포함됩니다. 이러한 요소는 XML 파일에서 다음과 같이 나타낼 수 있습니다.

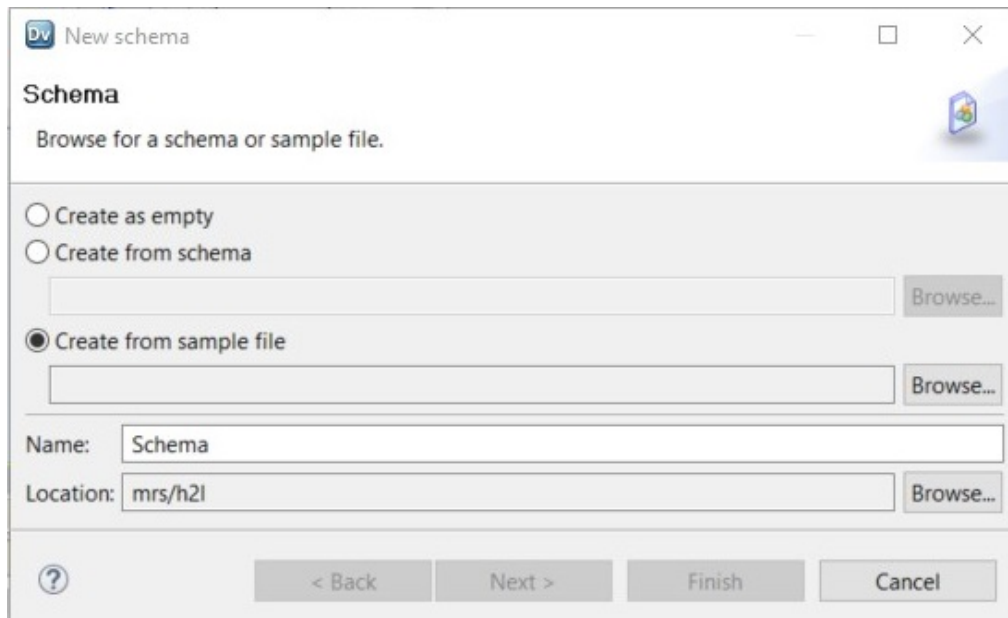
```
<Greeting><ID>sample</ID><Content>sample</Content></Person>
```

REST 웹 서비스 소비자 변환에서 이러한 XML 파일의 요소 계층을 사용하려면 Developer tool에서 XML 파일을 스키마 개체로 가져온 다음 REST 웹 서비스 소비자 변환을 생성할 때 스키마 개체를 사용하여 메서드 입력 및 메서드 출력을 정의합니다.

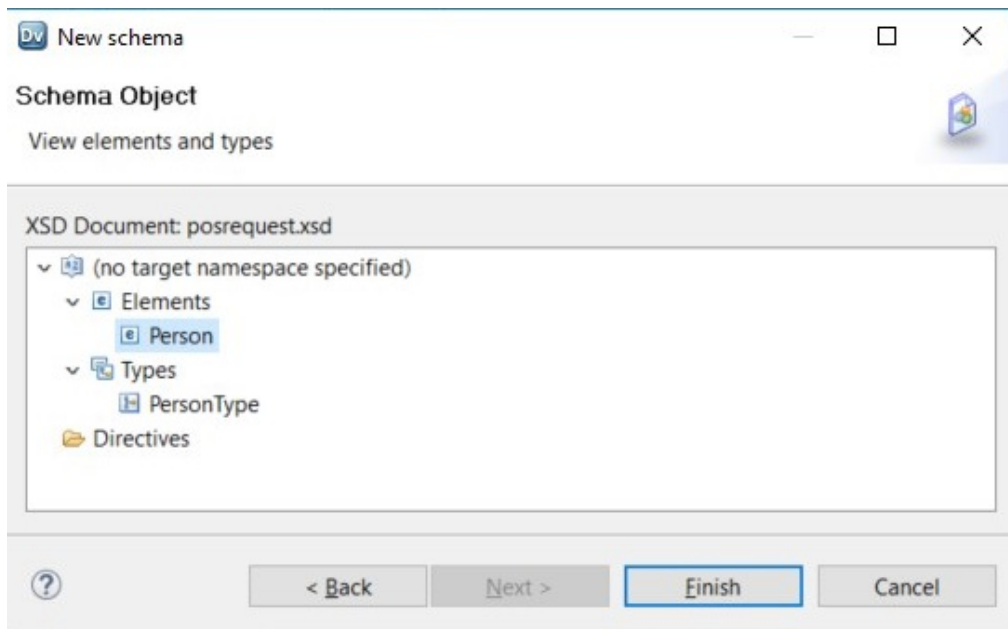
1단계. XML 파일에서 스키마를 생성합니다.

다음 단계를 완료하여 메서드 입력에 대한 스키마 개체를 생성합니다. 이러한 단계를 반복하여 메서드 출력에 대한 스키마 개체를 생성합니다.

1. Developer tool에서 **파일 > 새로 만들기 > 스키마**를 클릭하여 스키마 마법사를 엽니다.
2. **샘플 파일에서 생성**을 선택합니다.
3. **찾아보기**를 클릭합니다.
4. XML 파일로 이동하고 **열기**를 클릭합니다.
5. 스키마의 이름을 입력하고 프로젝트 위치를 확인합니다. 필요한 경우 새 위치를 찾아봅니다.



6. **다음**을 클릭하여 스키마 요소를 표시합니다.

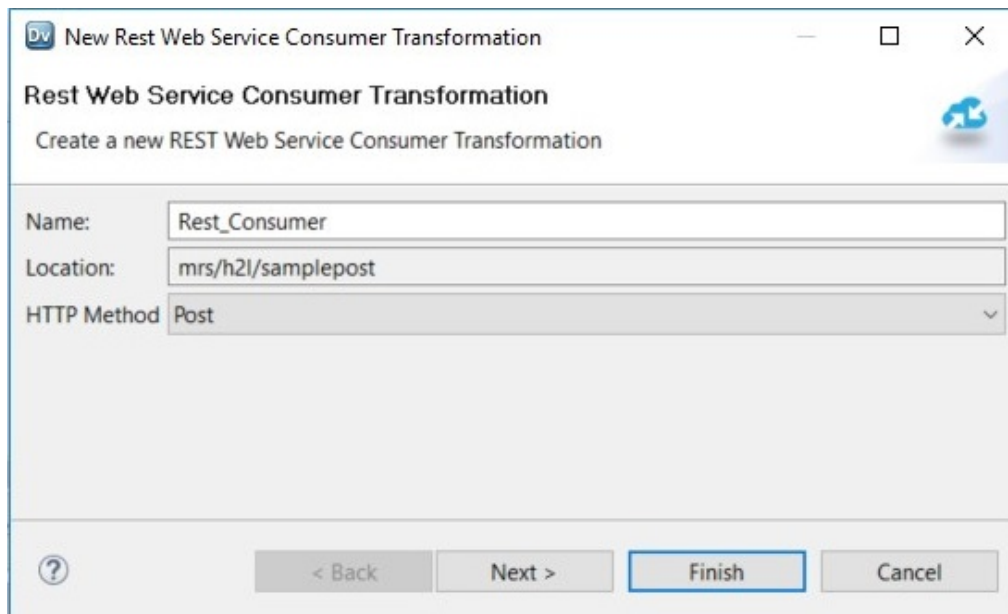


7. 마침을 클릭합니다.
이전에 생성한 스키마 개체를 사용하여 변환의 메서드 입력을 정의할 수 있습니다.
8. 1~7 단계를 반복하여 메서드 출력에 대한 스키마 개체를 생성합니다.

2단계. REST 웹 서비스 소비자 변환을 생성합니다.

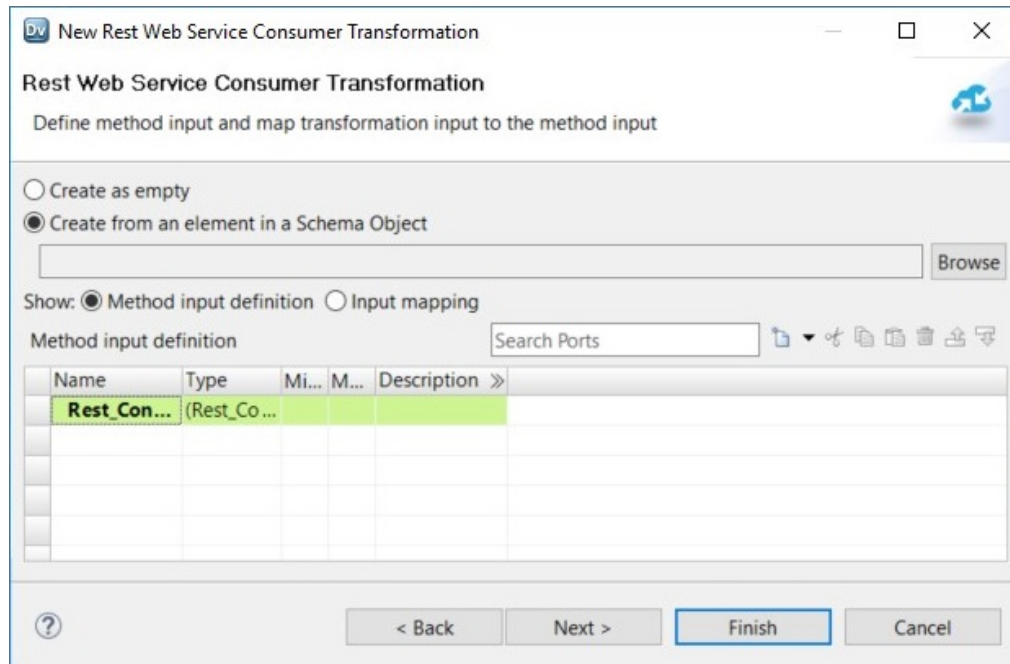
변환을 생성합니다. 스키마를 사용하여 메서드 입력 및 메서드 출력을 정의합니다.

1. Developer tool 매핑 편집기에서 매핑 팔레트를 아래로 스크롤하여 REST 웹 서비스 소비자 변환을 찾고 매핑으로 끌어서 놓습니다.
2. 변환 이름을 입력합니다.
3. HTTP 메서드를 선택합니다. 이 예에서는 HTTP 메서드 **Post**를 선택합니다.

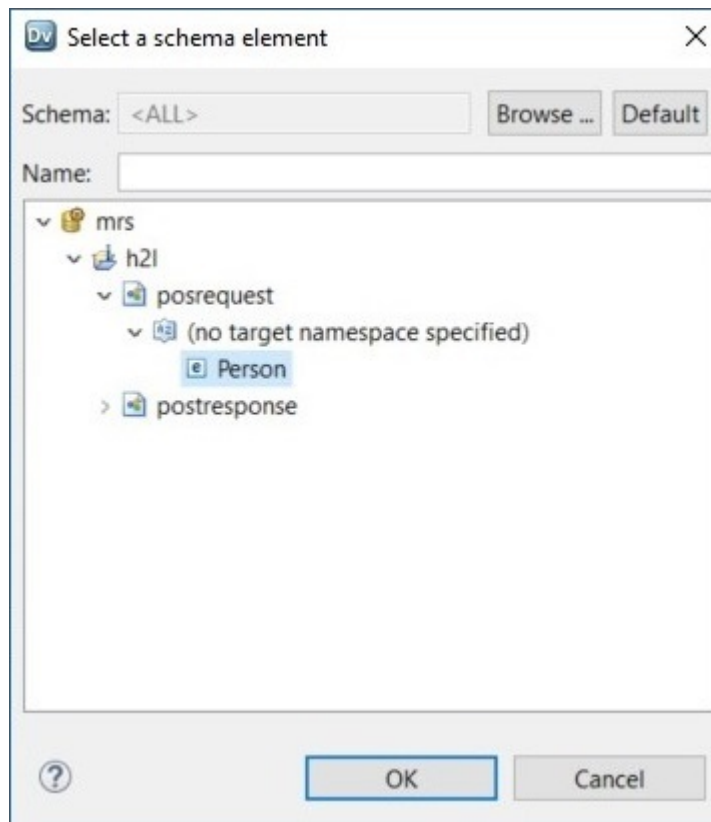


4. 다음을 클릭하여 메서드 입력을 정의합니다.

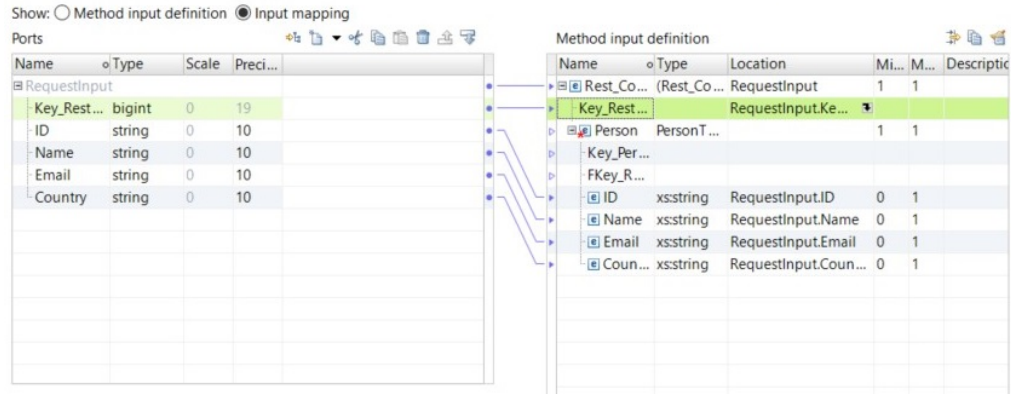
- 스키마 개체의 요소에서 생성을 선택합니다.



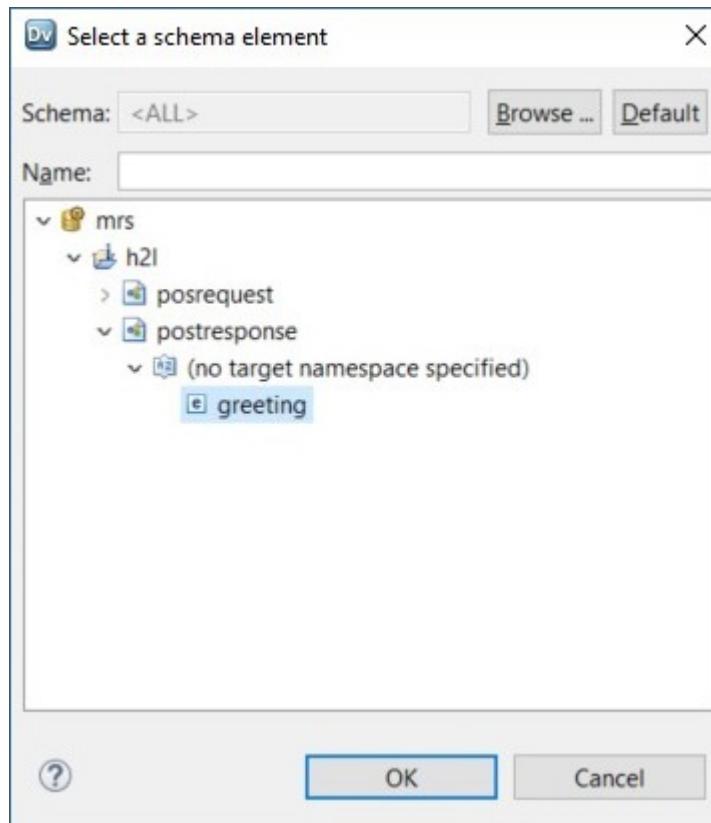
- 찾아보기를 클릭합니다.
- 스키마의 상위 요소를 선택합니다. 이 예에서는 메서드 입력에 사용된 스키마에 상위 요소인 **Person**이 포함되어 있습니다.



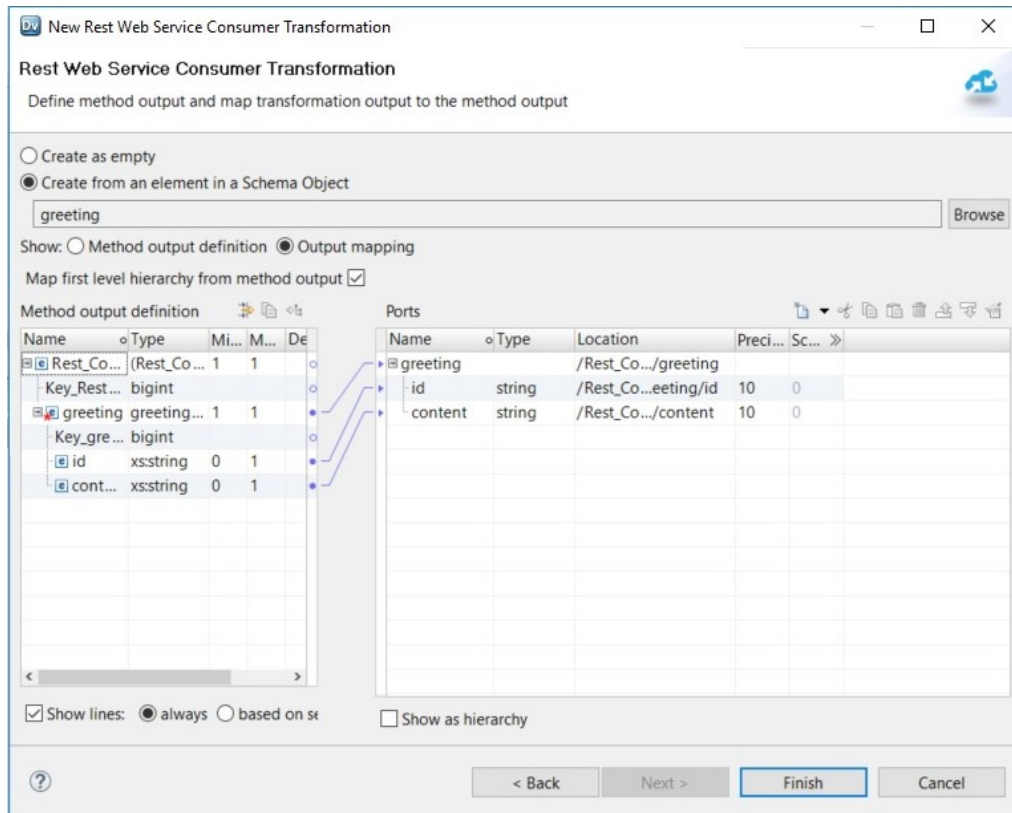
8. **확인**을 클릭합니다.
9. **입력 매핑**을 선택합니다.
10. 포트 영역의 **RequestInput** 아래 요소를 메서드 입력 정의 영역의 **Rest_Consumer_Input**에 매핑합니다. 포트 영역의 키를 메서드 입력 정의 영역에 매핑해야 합니다.



11. 메서드 출력을 정의하려면 변환 마법사에서 **다음**을 클릭합니다.
12. **스키마 개체의 요소에서 생성**을 선택합니다.
13. 스키마의 상위 요소를 선택합니다. 이 예에서는 메서드 출력에 사용되는 스키마에 상위 요소인 **greeting**이 포함되어 있습니다.

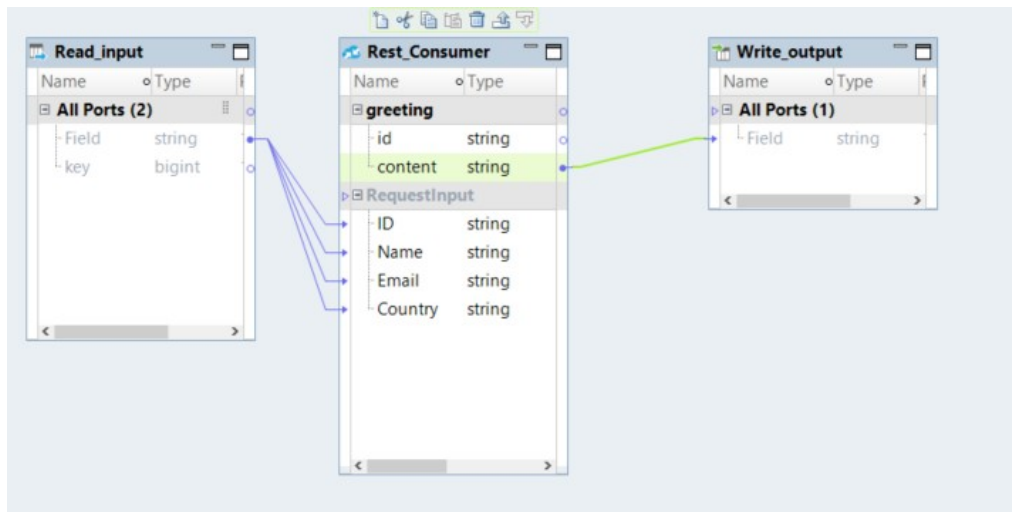


14. 메서드 출력 정의 영역의 **Rest_Consumer_Output** 아래 요소를 포트 영역에 매핑합니다. 키는 매핑하지 않아도 됩니다.



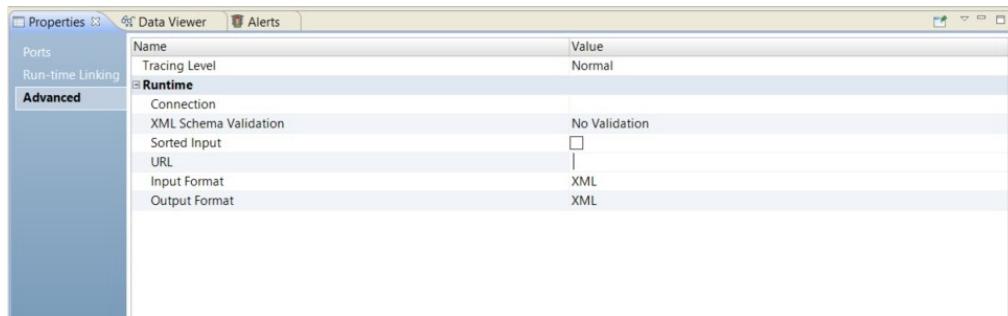
15. 마침을 클릭합니다.

다음 이미지는 매핑 워크플로우에서 변환이 어떻게 나타나는지를 보여 줍니다.



16. 매핑 편집기에서 REST 웹 서비스 소비자 변환을 선택합니다.
17. 속성 보기로 이동하고 고급 탭을 선택합니다.

18. 고급 속성을 구성합니다. 기본 URL을 입력하려면 **URL** 속성을 구성하거나 **연결** 속성에서 HTTP 연결을 입력합니다.



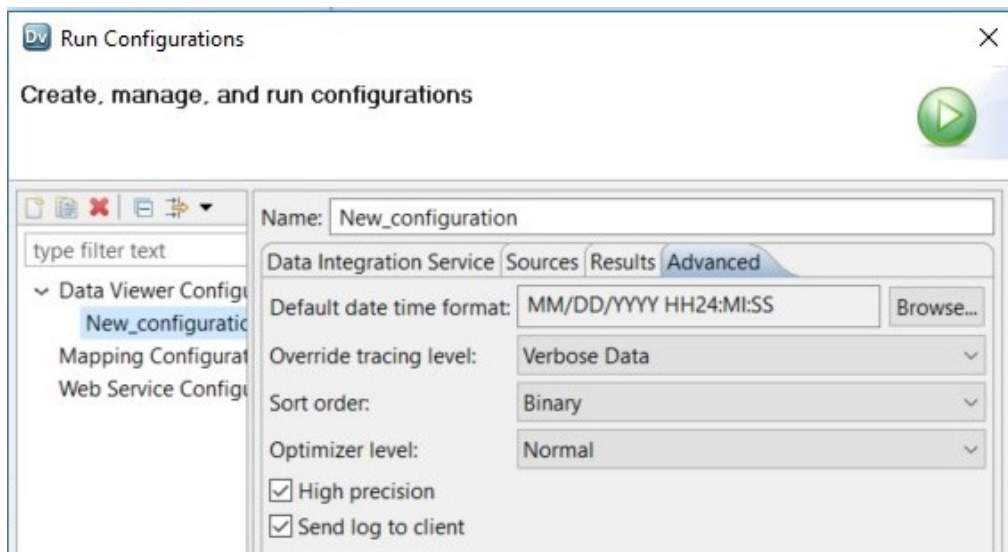
JSON 형식의 요청 및 응답 메시지를 구문 분석하는 방법

JSON 응답을 구문 분석하도록 REST 웹 서비스 소비자 변환을 설정하려면 Developer tool 기본 설정에서 로그 수준을 자세한 정보 표시 데이터로 설정합니다. 자세한 정보 표시 데이터 로그를 열고 응답을 XML 파일에 복사합니다. XML 파일을 스키마 개체로 가져오고 스키마를 사용하여 변환을 생성합니다. 변환을 생성한 후 JSON 응답을 수신하도록 변환을 구성합니다.

1단계. 로그 수준을 자세한 정보 표시 데이터로 설정합니다.

로그 수준을 자세한 정보 표시 데이터로 설정하려면 다음 단계를 완료합니다.

1. Developer tool에서 **창 > 기본 설정**을 클릭합니다.
2. **기본 설정** 창의 왼쪽에서 **구성 실행 > 매핑**을 클릭합니다.
3. 오른쪽에서 **고급** 탭을 선택합니다.
4. **추적 수준 재정의** 속성에 대해 **자세한 정보 표시 데이터**를 선택합니다.



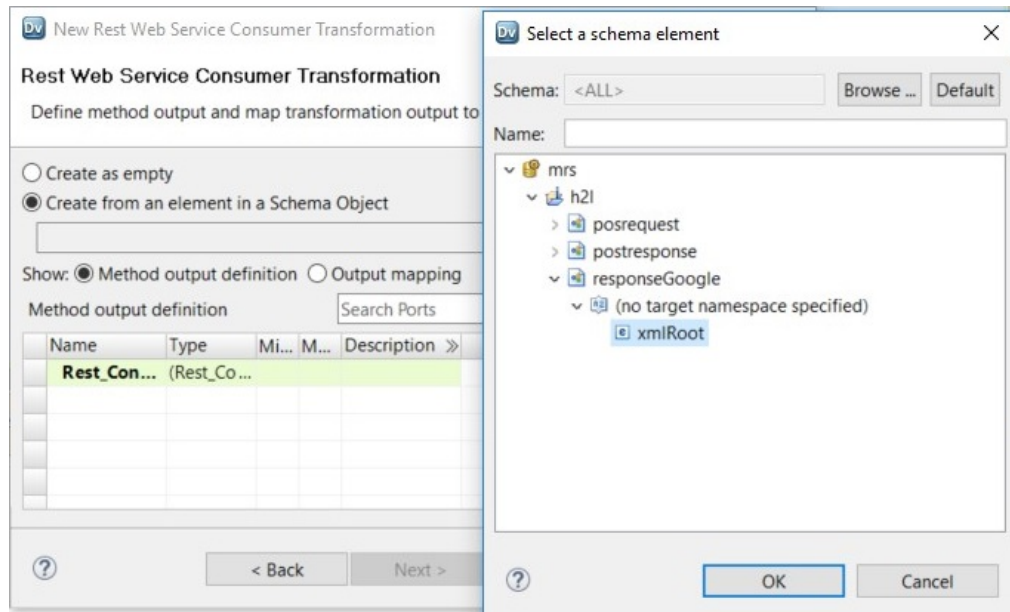
5. 자세한 정보 표시 데이터 로그를 엽니다.
6. **HTTP Response BODY: <?xml**을 검색합니다.
7. **<xmlRoot>**와 **</xmlRoot>** 사이의 데이터를 복사하고 데이터를 XML 파일로 저장합니다.
8. Developer tool에서 XML 파일을 스키마 개체로 가져옵니다.

2단계. 스키마를 사용하여 REST 웹 서비스 소비자 변환을 생성합니다.

스키마를 사용하여 REST 웹 서비스 소비자 변환이 수신하는 데이터를 구문 분석하려면 다음 단계를 완료합니다.

1. Developer tool 매핑 편집기에서 매핑 팔레트를 아래로 스크롤하여 REST 웹 서비스 소비자 변환을 찾고 매핑으로 끌어서 놓습니다.
2. 변환 이름을 입력합니다.
3. HTTP 메서드를 선택합니다.
4. 다음을 클릭합니다.
5. 메서드 입력을 정의하려면 **빈 항목으로 생성** 또는 **스키마 개체의 요소에서 생성**을 선택합니다. 메서드 입력을 정의하는 경우 메서드 입력 정의 및 입력 매핑을 정의합니다.
6. 다음을 클릭합니다.
7. 메서드 출력을 정의하려면 **스키마 개체의 요소에서 생성**을 선택합니다.
8. 찾아보기를 클릭합니다.
9. **xmlRoot** 요소를 선택합니다.

다음 이미지는 xmlRoot 요소가 나타나는 방식을 보여 줍니다.



10. 출력 매핑을 구성합니다. 상위 요소에 하위 요소가 포함되는 경우 하위 요소를 수동으로 매핑해야 합니다. 메서드 출력 정의 영역의 상위 및 하위 요소 그룹을 포트 영역에 끌어다 놓습니다.

Ports

Run-time Linking
Advanced

Show: ☐ Method output definition ☒ Output mapping

Method output definition

Name	Type	M...	Description
Rest.Co..._Rest.Co...	1	1	
Key_Rest...			
xmRoot	xmRoot...	1	1
Key_xm...	bigint		
results	resultsty...	0	U...
Key_re...	bigint		
for...	xsstring	0	1
types	xsstring	0	1
geo...	geometr...	0	1
Key...	bigint		
view...	viewpor...	0	1
Key...	bigint		
s...	southwe...	0	1
Ke...	bigint		
In xsstring		0	1
In xsstring		0	1
n...	northea...	0	1
Ke...	bigint		
In xsstring		0	1
In xsstring		0	1
bo...	boundT...	0	1
Key...	bigint		
s...	southwe...	0	1
Ke...	bigint		
In xsstring		0	1
In xsstring		0	1
n...	northea...	0	1
Ke...	bigint		
In xsstring		0	1
loc...	location...	0	1
Key...	bigint		
In xsstring		0	1
lat	xsstring	0	1
loc...	xsstring	0	1
add...	address...	0	U...
Key...	bigint		
qty...	xsstring	0	U...
Key...	bigint		
sh...	xsstring	0	1
lo...	xsstring	0	1

☒ Show lines: ☒ always ☐ based on selection

Map first level hierarchy ☐ Input ☐ Output

Ports

Name	Type	Location
Rest.Consumer_output		/Rest.Co..._out...
xmRoot	bigint	/Rest.Co..._xmRo...
status	string	/Rest.Co..._stati...
results		/Rest.Co..._resu...
Key_results	bigint	/Rest.Co..._resu...
formatted_address	string	/Rest.Co..._addr...
types	string	/Rest.Co..._ntstyp...
place_id	string	/Rest.Co..._place...
address_components		/Rest.Co..._ponen...
Key_results	bigint	/Rest.Co..._resu...
Key_address_components	bigint	/Rest.Co..._ponen...
short_name	string	/Rest.Co..._ort_nar...
long_name	string	/Rest.Co..._ng_nam...
geometry		/Rest.Co..._eomet...
Key_results	bigint	/Rest.Co..._st/resu...
Key_geometry	bigint	/Rest.Co..._geom...
location_type	string	/Rest.Co..._tion_ty...
location		/Rest.Co..._locat...
Key_geometry	bigint	/Rest.Co..._geom...
Key_location	bigint	/Rest.Co..._locat...
lng	string	/Rest.Co..._ation/li...
lat	string	/Rest.Co..._cation/l...
viewport		/Rest.Co..._viewpo...
Key_geometry	bigint	/Rest.Co..._geom...
Key_viewport	bigint	/Rest.Co..._viewpo...
southwest		/Rest.Co..._southw...
Key_viewport	bigint	/Rest.Co..._viewpo...
Key_southwest	bigint	/Rest.Co..._southw...
lng	string	/Rest.Co..._west/in...
lat	string	/Rest.Co..._hwest/i...
northeast		/Rest.Co..._northe...
Key_viewport	bigint	/Rest.Co..._viewpo...
Key_northeast	bigint	/Rest.Co..._northe...
lng	string	/Rest.Co..._heat/I...
lat	string	/Rest.Co..._theast/I...
types		/Rest.Co..._nts/typ...
Key_address_components	bigint	/Rest.Co..._ponen...
Key_types	bigint	/Rest.Co..._nts/typ...
types	string	/Rest.Co..._nts/typ...

☐ Show as hierarchy

12. 변환을 실행합니다.

다음 이미지는 수신할 수 있는 응답의 예를 보여 줍니다.

Key_Rest...	
1	1

Row 1 to 1

status	
1	OK

Row 1 to 1

Key_results	formatted...	types	place_id
1	1	4000 SE 8...	street_add... ChIJ_ZYPcj...
2	2	La Jota Hal...	premise ChIJyQVpn...
3	3	KY-1600, K...	route ChIJKabb6...
4	4	NC-1600,...	route ChIJ7Z6kO...
5	5	NC-1600	route ChIJX2Av

Row 1 to 5

Key_results	Key_addre...	short_name	long_name
1	1	1	1600 1600
2	1	2	4000 4000
3	1	3	SE 82nd Ave Southeast...
4	1	4	Lents Lents
5	1	5	Portland Portland

Row 1 to 30

Key_addre...	Key_types	types
1	1	1 subpremise
2	2	2 street_nu...
3	3	3 route
4	4	4 neighborh...
5	4	5 political

Row 1 to 48

Key_results	Key_geom...	location_ty...
1	1	1 ROOFTOP
2	2	2 ROOFTOP

응답의 데이터는 구문 분석 후 상위 요소 및 중첩된 하위 요소로 구성된 그룹을 형성합니다.

REST 웹 서비스 소비자 변환에서 사용자 지정 URL을 생성하는 방법

REST 웹 서비스 소비자 변환은 기본 URL을 사용하여 웹 서비스에 연결합니다. URL 포트 또는 인수 포트를 정의하여 기본 URL을 사용자 지정할 수 있습니다.

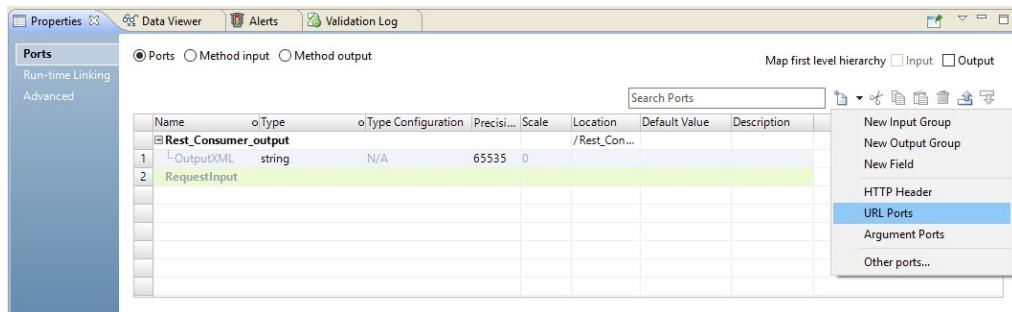
URL 포트를 구성하여 동적 URL을 생성하거나 인수 포트를 구성하여 웹 서비스에서 액세스하는 리소스를 URL로 필터링할 수 있습니다. 데이터 통합 서비스는 URL 또는 인수 포트의 값을 기본 URL에 추가합니다.

URL 포트를 사용하여 동적 URL을 작성하는 방법

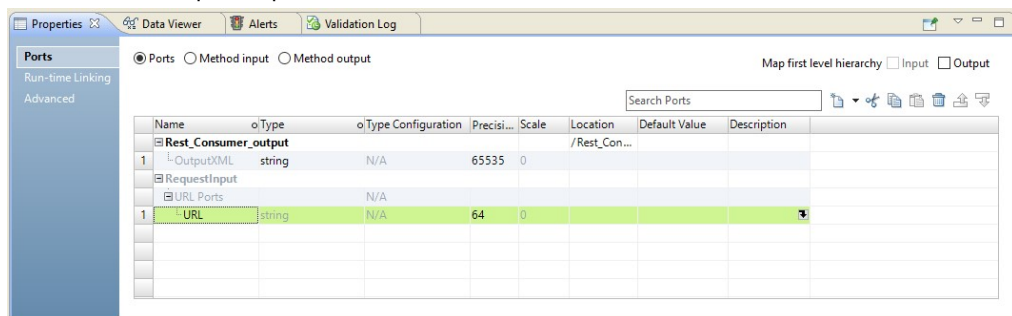
URL 포트를 사용하여 REST 웹 서비스 소비자 변환의 동적 URL을 작성할 수 있습니다. 변환에서 URL 포트를 정의하여 `https://sample.com/1/posts` 또는 `https://sample.com/2/posts` 같은 동적 URL을 작성할 수 있습니다.

REST 웹 서비스 소비자 변환을 생성한 후 동적 URL을 구성하려면 다음 단계를 완료합니다.

1. REST 웹 서비스 소비자 변환을 선택합니다.
2. 속성 보기로 이동하고 고급 탭을 선택합니다.
3. 기본 URL을 구성하려면 다음 단계 중 하나를 완료합니다.
 - URL 속성을 구성하려면 기본 URL을 입력합니다.
 - 연결 속성을 구성하려면 HTTP 연결을 선택합니다.
4. 포트 탭으로 이동합니다.
5. RequestInput 그룹을 강조 표시하고 새로 만들기를 선택합니다. 드롭다운 메뉴에서 URL 포트를 선택합니다.



다음 이미지는 RequestInput 아래 중첩된 URL 포트를 보여 줍니다.



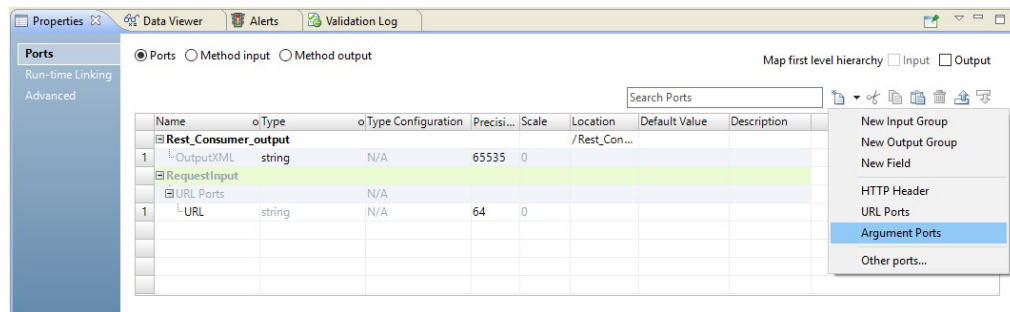
6. URL 포트의 이름을 편집합니다.
이 예에서는 URL 포트 이름을 number로 변경합니다.
7. 게시에 사용할 다른 URL 포트를 생성합니다. 포트 이름을 posts로 변경합니다.
8. `https://sample.com/1/posts` 또는 `https://sample.com/2/posts` 같은 동적 URL을 작성하려면 업스트림 매핑 개체의 포트를 REST 웹 서비스 소비자 변환의 URL 포트에 연결합니다. 숫자 포트를 number URL 포트에 연결하고 문자열 포트를 posts URL 포트에 연결합니다.

인수 포트를 사용하여 URL 매개 변수를 생성하는 방법

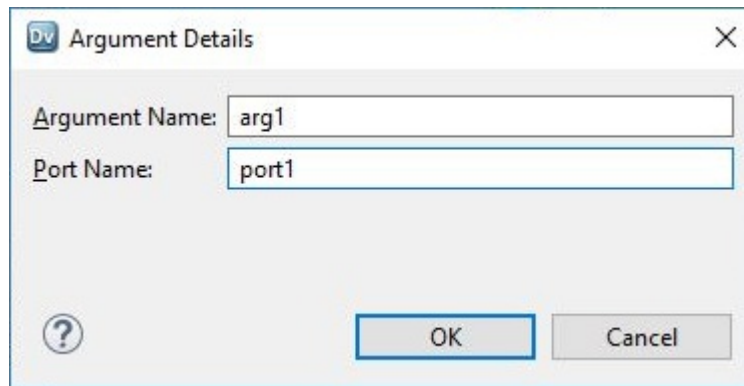
인수 포트 URL의 매개 변수를 생성하여 웹 서비스에서 요청하는 데이터를 필터링할 수 있습니다. 예를 들어 Get 메서드를 사용하는 경우 메서드 입력 정의 또는 입력 매핑을 구성할 수 없지만 Get 요청에서 필터를 사용할 수 있습니다. 인수 포트를 사용하여 요청 메시지에 필터를 생성할 수 있습니다. 인수 포트 URL 매개 변수를 생성하여 `https://sample.com?arg1=val1&arg2=val2` 같은 URL을 작성합니다.

REST 웹 서비스 소비자 변환을 생성한 후 인수 포트를 구성하려면 다음 단계를 완료합니다.

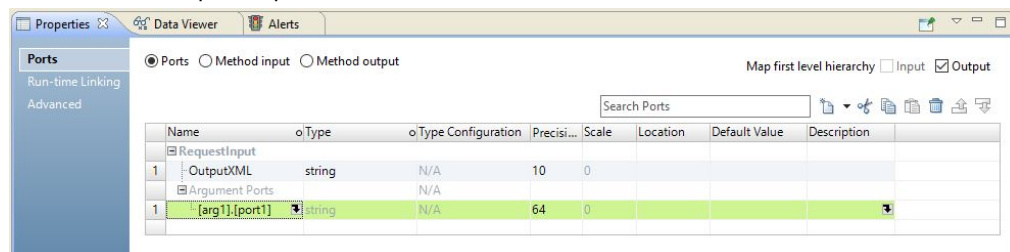
1. REST 웹 서비스 소비자 변환을 선택합니다.
2. 속성 보기로 이동하고 **고급** 탭을 선택합니다.
3. 기본 URL을 구성하려면 다음 단계 중 하나를 완료합니다.
 - URL 속성을 구성하려면 기본 URL을 입력합니다.
 - 연결 속성을 구성하려면 HTTP 연결을 선택합니다.
4. **포트** 탭으로 이동합니다.
5. **RequestInput** 그룹을 강조 표시하고 **새로 만들기**를 선택합니다. 드롭다운 메뉴에서 **인수 포트**를 선택합니다.



6. 인수 이름 및 포트 이름에 대한 값을 입력합니다.
포트 이름에 데이터 통합 서비스가 처리할 수 없는 문자가 포함되는 경우 포트 이름을 변경합니다.

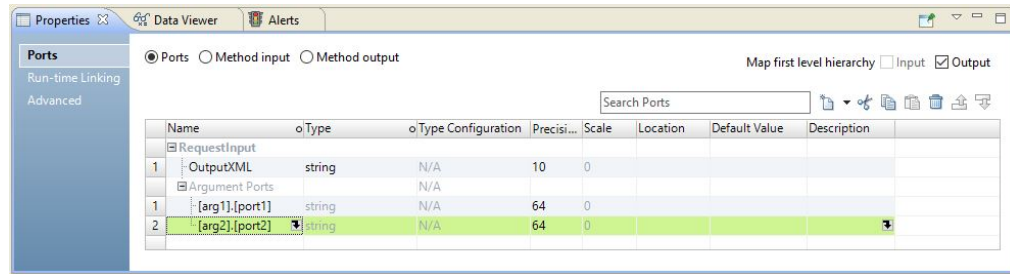


다음 이미지는 RequestInput 아래 중첩된 인수 포트를 보여 줍니다.



7. 3 및 4단계를 반복하여 두 번째 인수 포트를 추가합니다.

첫 번째 인수 포트 아래 두 번째 인수 포트가 나타납니다.



인수 포트 값이 `http://sample.com?arg1=val1&arg2=val2` 형식으로 URL에 추가됩니다. 포트 이름은 URL에 사용되지 않습니다. 데이터 통합 서비스는 포트 이름을 사용하여 인수 포트를 식별합니다.

8. `arg1` 및 `arg2` 인수 포트를 기본 URL의 매개 변수로 사용하려면 업스트림 매핑 개체의 포트를 REST 웹 서비스 소비자 변환의 인수 포트에 연결합니다.

업스트림 개체에서 연결된 포트는 URL `https://sample.com?arg1=val1&arg2=val2`의 `val1` 및 `val2`를 정의합니다.

제 14 장

REST 및 SOAP 웹 서비스 관리

이 장에 포함된 항목:

- [웹 서비스 관리 개요, 172](#)
- [웹 서비스 속성 구성, 172](#)
- [웹 서비스 보안 관리, 176](#)
- [그리드의 웹 서비스, 180](#)
- [웹 서비스 로그, 181](#)
- [웹 서비스 모니터링, 181](#)

웹 서비스 관리 개요

Administrator 도구에서 REST 또는 SOAP 웹 서비스를 관리할 수 있습니다. 웹 서비스 보안 구성, 웹 서비스 구성, 웹 서비스 로그 보기 및 웹 서비스 요청 모니터링을 수행할 수 있습니다. 이러한 태스크는 적절한 사용 권한이 있어야만 수행할 수 있습니다.

웹 서비스를 데이터 통합 서비스에 배포한 후에는 다음과 같은 태스크를 수행할 수 있습니다.

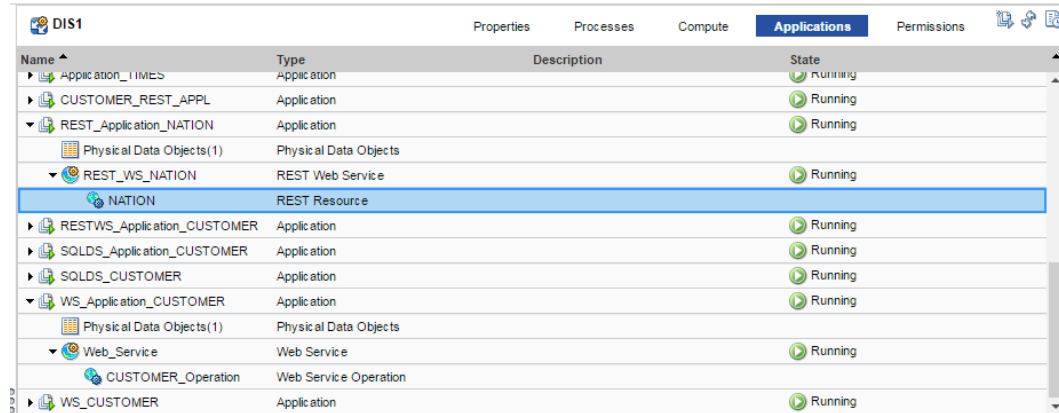
- 웹 서비스 보안 구성. 웹 서비스 보안을 활성화하고 웹 서비스에 대한 사용 권한을 할당합니다.
- 웹 서비스 속성 및 웹 서비스 작업 속성을 구성합니다.
- 웹 서비스 작업을 그리드에서 실행하도록 데이터 통합 서비스를 구성합니다.
- 웹 서비스 로그 보기. 웹 서비스에 대한 데이터 통합 서비스 로그를 봅니다. 웹 서비스 런타임 로그 디렉터리에 있는 웹 서비스 런타임 로그를 봅니다.
- 웹 서비스 모니터링. Administrator 도구나 Monitoring 도구를 사용하여 웹 서비스 요청을 모니터링합니다.

웹 서비스 속성 구성

Administrator 도구에서는 데이터 통합 서비스에 배포하는 각 웹 서비스의 웹 서비스 속성을 구성할 수 있습니다.

Administrator 도구에서 데이터 통합 서비스의 **응용 프로그램** 보기에서 웹 서비스의 속성을 편집합니다. 응용 프로그램 이름을 확장하고 웹 서비스 또는 REST 웹 서비스를 선택합니다. **속성** 보기에 속성이 표시됩니다.

다음 이미지는 데이터 통합 서비스의 **응용 프로그램** 보기를 보여 줍니다.



SOAP 웹 서비스 작업 또는 REST 웹 서비스 리소스의 속성도 편집할 수 있습니다. 속성을 보려면 작업 이름 또는 리소스 이름을 선택합니다.

웹 서비스 속성

REST 웹 서비스 및 SOAP 웹 서비스 속성에는 데이터 통합 서비스가 웹 서비스를 실행할 때 사용하는 속성 및 읽기 전용 일반 속성이 포함되어 있습니다.

응용 프로그램 보기의 맨 위 패널에서 웹 서비스 또는 REST 웹 서비스를 확장할 경우 웹 서비스 작업 또는 웹 서비스 리소스에 액세스할 수 있습니다.

응용 프로그램 보기에는 웹 서비스, 웹 서비스 작업 또는 웹 서비스 리소스에 대한 읽기 전용 일반 속성이 표시됩니다. 보기에 나타나는 속성은 개체 유형에 따라 다릅니다.

다음 테이블에는 각 유형의 웹 서비스 및 웹 서비스 작업 또는 리소스에 대한 읽기 전용 일반 속성이 설명되어 있습니다.

속성	설명
이름	선택된 개체의 이름입니다. 모든 개체에 대해 표시됩니다.
설명	선택된 개체에 대한 간략한 설명입니다. 모든 개체에 대해 표시됩니다.
유형	선택된 개체의 유형입니다. 모든 개체 유형에 대해 표시됩니다.
위치	선택된 개체의 위치입니다. 도메인 및 데이터 통합 서비스 이름이 포함됩니다. 모든 개체에 대해 표시됩니다.
URL	웹 서비스에 연결하는 데 사용되는 URL입니다. 웹 서비스에 대해 표시됩니다.

다음 테이블에는 웹 서비스에 대해 구성 가능한 웹 서비스 속성이 설명되어 있습니다.

속성	설명
시작 유형	응용 프로그램을 시작할 때 또는 웹 서비스를 시작할 때 웹 서비스를 활성화하고 실행할지 결정합니다.
추적 수준	런타임 웹 서비스 로그에 기록되는 오류 메시지의 수준입니다. 다음 메시지 수준 중 하나를 선택합니다. <ul style="list-style-type: none"> - OFF. DTM 프로세스에서 웹 서비스 런타임 로그에 메시지를 기록하지 않습니다. - SEVERE. SEVERE 메시지에는 웹 서비스가 실행을 중지할 수도 있는 오류가 포함됩니다. - WARNING. WARNING 메시지에는 복구 가능한 실패 또는 경고가 포함됩니다. DTM 프로세스에서 웹 서비스 런타임 로그에 WARNING 및 SEVERE 메시지를 기록합니다. - INFO. INFO 메시지에는 웹 서비스 상태 메시지가 포함됩니다. DTM 프로세스에서 웹 서비스 런타임 로그에 INFO, WARNING 및 SEVERE 메시지를 기록합니다. - FINE. FINE 메시지에는 웹 서비스 요청에 대한 데이터 처리 오류가 포함됩니다. DTM 프로세스에서 웹 서비스 런타임 로그에 FINE, INFO, WARNING 및 SEVERE 메시지를 기록합니다. - FINEST. FINEST 메시지는 디버깅에 사용됩니다. DTM 프로세스에서 웹 서비스 런타임 로그에 FINEST, FINE, INFO, WARNING 및 SEVERE 메시지를 기록합니다. - ALL. DTM 프로세스에서 웹 서비스 런타임 로그에 FINEST, FINE, INFO, WARNING 및 SEVERE 메시지를 기록합니다. 기본값은 INFO입니다.
요청 제한 시간	웹 서비스 요청 시간이 초과되기 전에 데이터 통합 서비스에서 작업 매핑을 실행하는 최대 시간(밀리초)입니다. 기본값은 3,600,000입니다.
최대 동시 요청 수	웹 서비스에서 한번에 처리할 수 있는 최대 요청 수입니다. 기본값은 10입니다.
정렬 순서	데이터 통합 서비스가 유니코드 모드에서 실행될 때 데이터를 정렬하고 비교하기 위해 사용하는 정렬 순서입니다.
TLS(Transport Layer Security) 활성화	웹 서비스에서 HTTPS를 사용해야 함을 나타냅니다. 데이터 통합 서비스가 HTTPS를 사용하도록 구성되지 않은 경우에는 웹 서비스가 시작되지 않습니다.

다음 테이블에는 REST 웹 서비스에 고유한 속성이 포함되어 있습니다.

속성	설명
인증 필요	REST 웹 서비스에 대한 기본 인증을 활성화합니다. 기본 인증에서는 웹 서비스 요청의 사용자 이름 및 암호가 필요합니다. 기본값은 비활성화됩니다.
입력 전체 자릿수	데이터 통합 서비스가 요청 메시지에서 구문 분석하는 최대 문자 수입니다. 요청 메시지가 입력 전체 자릿수를 초과할 경우 웹 서비스 요청이 실패합니다. 기본값은 10,000입니다.
출력 전체 자릿수	데이터 통합 서비스가 응답 메시지에 대해 생성하는 최대 문자 수입니다. 응답 메시지가 출력 전체 자릿수를 초과할 경우 데이터 통합 서비스에서 응답 메시지를 자릅니다. 기본값은 3,000입니다.

다음 테이블에는 SOAP 웹 서비스에 고유한 속성이 포함되어 있습니다.

속성	설명
WS-보안 활성화	데이터 통합 서비스가 사용자 자격 증명의 유효성을 검사하고 사용자에게 각 웹 서비스 작업을 실행할 수 있는 권한이 있는지 확인할 수 있도록 합니다. SOAP 웹 서비스 전용입니다.
최적화 수준	데이터 통합 서비스에서 개체에 적용하는 최적화 수준입니다. 구성할 최적화 수준에 연결된 숫자 값을 입력합니다. 다음 숫자 값 중 하나를 입력할 수 있습니다. <ul style="list-style-type: none"> - 0. 데이터 통합 서비스가 최적화를 적용하지 않습니다. - 1. 데이터 통합 서비스가 초기 예측 최적화 방법을 적용합니다. - 2. 데이터 통합 서비스가 초기 예측, 초기 선택, 푸시인 및 조건자 최적화 방법을 적용합니다. - 3. 데이터 통합 서비스가 비용 기반, 초기 예측, 초기 선택, 푸시인, 조건자 및 반 조인 최적화 방법을 적용합니다.
DTM 활성 상태 유지 시간	마지막 요청을 완료한 후 DTM 인스턴스가 열려 있는 시간(밀리초)입니다. 동일한 작업에 대해 발행된 웹 서비스 요청은 열려 있는 인스턴스를 재사용할 수 있습니다. 요청을 처리하는 데 필요한 시간이 DTM 인스턴스에 대한 초기화 시간에 비해 작을 경우 활성 상태 유지 시간을 사용하면 성능을 향상시킬 수 있습니다. 요청이 실패할 경우 DTM 인스턴스가 종료됩니다. 정수여야 합니다. 음의 정수 값은 데이터 통합 서비스에 대해 DTM 활성 상태 유지 시간이 사용되고 있음을 의미합니다. 0은 데이터 통합 서비스가 메모리에서 DTM 인스턴스를 유지하지 않음을 의미합니다. 기본값은 -1입니다.
SOAP 출력 전체 자릿수	데이터 통합 서비스가 응답 메시지에 대해 생성하는 최대 문자 수입니다. 응답 메시지가 SOAP 출력 전체 자릿수를 초과할 경우 데이터 통합 서비스에서 응답 메시지를 자릅니다. 기본값은 200,000입니다.
SOAP 입력 전체 자릿수	데이터 통합 서비스가 요청 메시지에서 구문 분석하는 최대 문자 수입니다. 요청 메시지가 SOAP 입력 전체 자릿수를 초과할 경우 웹 서비스 요청이 실패합니다. 기본값은 200,000입니다.

웹 서비스 작업 및 리소스 속성

데이터 통합 서비스가 웹 서비스 작업 또는 웹 서비스 리소스를 실행할 때 사용하는 설정을 구성합니다.

다음 테이블에는 SOAP 웹 서비스 작업 또는 REST 웹 서비스 리소스에 대해 구성 가능한 속성이 설명되어 있습니다.

속성	설명
결과 집합 캐시 만료 기간	결과 집합 캐시를 사용할 수 있는 기간(밀리초)입니다. -1로 설정하면 캐시가 만료되지 않습니다. 0으로 설정하면 결과 집합 캐시가 비활성화됩니다. 만료 기간 변경은 기존 캐시에 적용되지 않습니다. 모든 캐시가 동일한 만료 기간을 사용하게 하려면 만료 기간을 변경한 후 결과 집합 캐시를 제거합니다. 기본값은 0입니다.

웹 서비스 결과 집합 캐싱

결과 집합 캐싱을 사용하면 데이터 통합 서비스가 웹 서비스 요청에 대해 캐싱된 결과를 사용할 수 있습니다. 짧은 시간 내에 동일한 쿼리를 실행하는 사용자가 있다면 결과 집합 캐싱을 사용하여 동일한 쿼리의 런타임을 줄일 수 있습니다.

결과 집합 캐싱을 구성하면 데이터 통합 서비스가 각 웹 서비스 요청과 연결된 DTM 프로세스의 결과를 캐싱합니다. 데이터 통합 서비스는 사용자가 구성한 만료 기간 동안 결과를 캐싱합니다. 캐시가 만료되기 전에 외부 클라이언트가 동일한 쿼리 또는 요청을 하면 데이터 통합 서비스가 캐싱된 결과를 반환합니다. 캐시가 없거나 만료된 경우 데이터 통합 서비스는 DTM 인스턴스를 시작하여 요청을 처리합니다.

캐시에 포함된 데이터의 양이 최대 캐시 메모리 크기를 초과하면 데이터 통합 서비스는 결과 집합을 <Informatica_install_dir>/tomcat/bin/disTemp/<Service_Name>/<Node_Name>/에 있는 암호화된 캐시 파일에 저장합니다.

웹 서비스에서 **WS-Security**를 사용할 경우 데이터 통합 서비스는 사용자의 웹 서비스에 대한 결과 집합 캐시를 저장합니다. 데이터 통합 서비스는 웹 서비스 요청의 사용자 이름 토큰에 제공된 사용자 이름을 사용하여 캐시를 저장합니다. 데이터 통합 서비스는 사용자별 결과를 캐싱한 경우 웹 서비스 요청을 전송한 사용자에게 캐싱된 결과만 반환합니다.

Administrator 도구에서 결과 집합 캐싱을 구성하려면 다음 단계를 사용합니다.

1. 데이터 통합 서비스 프로세스 속성에서 결과 집합 캐시 속성을 구성합니다.

다음 테이블에는 결과 집합 캐시 속성이 설명되어 있습니다.

속성	설명
최대 총 디스크 크기	총 결과 집합 캐시 저장소에 대해 허용되는 최대 바이트 수입니다. 기본값은 0입니다.
캐시 메모리 크기당 최대	메모리에서 단일 결과 집합 캐시 인스턴스에 대해 할당할 최대 바이트 수입니다. 기본값은 0입니다.
최대 총 메모리 크기	메모리에서 총 결과 집합 캐시 저장소에 대해 할당되는 최대 바이트 수입니다. 기본값은 0입니다.
최대 캐시 수	데이터 통합 서비스에 대해 허용되는 최대 결과 집합 캐시 인스턴스 수입니다. 기본값은 0입니다.

2. **SOAP** 웹 서비스 작업 속성 또는 **REST** 웹 서비스 리소스 속성에 캐시 만료 기간을 구성합니다.
결과 집합 캐시 만료 기간은 결과 집합을 사용할 수 있는 밀리초 수입니다. -1로 설정하면 캐시가 만료되지 않습니다. 0으로 설정하면 결과 집합 캐싱이 비활성화됩니다. 만료 기간 변경은 기존 캐시에는 적용되지 않습니다. 모든 캐시에 동일한 만료 기간을 사용하려면 만료 기간을 변경한 후 결과 집합 캐시를 제거합니다. 기본값은 0입니다.
3. 데이터 통합 서비스가 사용자별로 결과를 캐싱하도록 하려면 웹 서비스 속성에서 **WS-Security**를 활성화합니다.

웹 서비스 작업이 결과 집합을 캐싱하도록 구성되어 있을 때 웹 서비스 요청에 대한 결과 집합 캐싱을 비활성화하려면 **SOAP** 요청의 **HTTP** 헤더에 다음 구문을 포함시킵니다. `WebServiceOptions.disableResultSetCache=true`

웹 서비스 보안 관리

HTTP 클라이언트 필터, **TLS(Transport Layer Security)** 및 메시지 레이어 보안은 웹 서비스에 대해 허용된 데이터 액세스 및 보안 데이터 전송을 제공할 수 있습니다. 메시지 레이어 보안을 구성하는 경우 데이터 통합 서비스는 연결에 자격 증명을 전달할 수 있습니다.

REST 웹 서비스에 대해 다음 보안 옵션을 구성할 수 있습니다.

인증 필요

REST 웹 서비스에 대한 기본 인증을 활성화합니다. 기본 인증에서는 각 웹 서비스 요청에 도메인에 대한 사용자 이름 및 암호가 포함되어 있어야 합니다. **Administrator** 도구의 데이터 통합 서비스에서 속성을 활성화합니다. **응용 프로그램 > ApplicationName REST 웹 서비스 > isAuthenticationRequired**를 클릭합니다. 인증이 필요하도록 설정한 경우 각 **GET** 요청에 사용자 이름 및 암호를 제공해야 **REST** 웹 서비스가 응답을 반환합니다. 기본값은 비활성화됩니다.

SOAP 웹 서비스에 대해 다음 보안 옵션을 구성할 수 있습니다.

HTTP 클라이언트 필터

데이터 통합 서비스가 웹 서비스 클라이언트의 호스트 이름 또는 IP 주소를 기반으로 요청을 수락하게 하려는 경우 **Administrator** 도구를 사용하여 **HTTP 클라이언트 필터**를 구성하십시오. 시스템에서 실행되는 웹 서비스 클라이언트는 기본적으로 요청을 보낼 수 있습니다.

메시지 레이어 보안

데이터 통합 서비스가 **SOAP** 요청의 사용자 자격 증명을 인증하게 하려는 경우 **Administrator** 도구를 사용하여 **WS-Security**를 활성화하고 웹 서비스 사용 권한을 구성하십시오. 데이터 통합 서비스는 **SOAP** 요청에서 사용자 이름 토큰으로 제공되는 사용자 자격 증명의 유효성을 검사할 수 있습니다. 사용자 이름 토큰이 유효하지 않은 경우 데이터 통합 서비스는 해당 요청을 거부하고 시스템 정의 오류를 웹 서비스 클라이언트에 보냅니다. 사용자가 웹 서비스 작업을 실행할 권한을 가지고 있지 않은 경우 데이터 통합 서비스는 해당 요청을 거부하고 시스템 정의 오류를 웹 서비스 클라이언트에 보냅니다.

TLS(Transport Layer Security)

웹 서비스 및 웹 서비스 클라이언트가 **HTTPS URL**을 사용하여 통신하게 하려는 경우 **Administrator** 도구를 사용하여 웹 서비스에 대한 **TLS(Transport Layer Security)**를 활성화하십시오. 또한 웹 서비스가 실행되는 데이터 통합 서비스가 **HTTPS** 프로토콜을 사용해야 합니다. **HTTPS URL**은 **SSL**을 사용하여 웹 서비스와 웹 서비스 클라이언트 간 데이터 전송을 위한 보안 연결을 제공합니다.

통과 보안

작업 매핑에 연결 자격 증명이 필요한 경우 데이터 통합 서비스는 **SOAP** 요청의 사용자 이름 토큰에서 연결로 자격 증명을 전달할 수 있습니다. 연결로 자격 증명을 전달하도록 데이터 통합 서비스를 구성하려면 **Administrator** 도구를 사용하여 연결에 대한 통과 보안을 사용하도록 데이터 통합 서비스를 구성하고 웹 서비스에 대한 **WS-Security**를 활성화하십시오.

참고: 사용자 이름 토큰에 해시된 암호나 요약 암호가 포함된 경우 통과 보안을 사용할 수 없습니다.

웹 서비스 사용 권한

웹 서비스에 사용자 인증이 필요한 경우 사용 권한은 **REST** 웹 서비스 또는 **SOAP** 웹 서비스에 대해 사용자가 갖고 있는 액세스 권한 수준을 제어합니다. **Administrator** 도구를 사용하면 **REST** 웹 서비스 또는 **SOAP** 웹 서비스에 대한 사용 권한을 구성할 수 있습니다. 리소스나 작업에 대한 사용 권한도 설정할 수 있습니다.

사용 권한을 할당하려면 데이터 통합 서비스에 대한 **응용 프로그램** 보기에서 웹 서비스, 리소스 또는 작업을 선택합니다. **사용자 사용 권한** 또는 **그룹 사용 권한**을 클릭합니다.

관리자는 다음과 같은 유형의 사용자 및 그룹에 웹 서비스 사용 권한을 할당합니다.

- 웹 서비스 소비자. 웹 서비스에 요청을 보내고 웹 서비스로부터 응답을 수신하는 원시 도메인 사용자입니다. 사용자에게는 웹 서비스에 대한 실행 권한이 있어야 합니다.
- 웹 서비스 관리자. **Administrator** 도구에 로그인하고, 웹 서비스 속성을 편집하며, 다른 사용자에게 사용 권한을 부여할 수 있는 사용자입니다.
- 웹 서비스 작업자. **Administrator** 도구에 로그인하고, 웹 서비스를 모니터링하며, 웹 서비스를 시작 또는 중지할 수 있는 사용자입니다.

다음 사용 권한을 사용자 및 그룹에 할당할 수 있습니다.

- 부여 사용 권한. 사용자는 **Administrator** 도구 또는 **infacmd** 명령줄 프로그램을 사용하여 웹 서비스 개체에 대한 사용 권한을 관리할 수 있습니다.
- 실행 사용 권한. 사용자가 웹 서비스 요청을 전송하고 웹 서비스 응답을 수신할 수 있습니다.

다음 테이블에는 각 SOAP 웹 서비스 개체에 대한 사용 권한이 설명되어 있습니다.

개체	부여 사용 권한	실행 사용 권한
SOAP 웹 서비스	웹 서비스 및 웹 서비스 내 모든 웹 서비스 작업에 대한 사용 권한을 부여하고 취소합니다.	웹 서비스 요청을 전송하고 웹 서비스 내 모든 웹 서비스 작업에서 웹 서비스 응답을 수신합니다.
SOAP 웹 서비스 작업	웹 서비스 작업에 대한 사용 권한을 부여, 취소 및 거부합니다.	웹 서비스 요청을 전송하고 웹 서비스 작업에서 웹 서비스 응답을 수신합니다.

다음 테이블에는 각 REST 웹 서비스 개체에 대한 사용 권한이 설명되어 있습니다.

개체	부여 사용 권한	실행 사용 권한
REST 웹 서비스	REST 웹 서비스 및 해당 웹 서비스 내 모든 웹 서비스 리소스에 대한 사용 권한을 부여하고 취소합니다.	웹 서비스 요청을 보내고 REST 웹 서비스에 포함된 모든 웹 서비스 리소스로부터 웹 서비스 응답을 수신합니다.
REST 리소스	REST 웹 서비스 리소스에 대한 사용 권한을 부여, 취소 및 거부합니다.	웹 서비스 요청을 보내고 REST 웹 서비스 리소스로부터 웹 서비스 응답을 수신합니다.

SOAP 요청의 UserName 토큰

웹 서비스에 사용자 인증이 필요한 경우 웹 서비스 클라이언트의 SOAP 요청에 사용자 이름 토큰 헤더가 포함되어야 합니다. 웹 서비스에 사용자 인증이 필요하지 않은 경우 데이터 통합 서비스가 SOAP 요청에서 제공된 사용자 이름 토큰 헤더를 무시합니다.

SOAP 요청의 사용자 이름 토큰 요소에 다음 암호 유형 중 하나가 있을 수 있습니다.

- 일반 텍스트
- 해시됨
- 요약

참고: 사용자 이름 토큰에 해시된 암호 또는 요약 암호가 포함된 경우 LDAP 인증을 사용할 수 없습니다.

UsernameToken 요소의 암호 요소에 사용자 암호를 포함하십시오. 암호 요소에는 사용된 암호 보안 유형을 표시하는 유형 특성이 있습니다.

일반 텍스트 암호

사용자 암호를 암호화하지 않아도 되는 경우 SOAP 요청의 사용자 이름 토큰 헤더에 일반 텍스트 암호를 포함합니다. 데이터 통합 서비스에서는 UsernameToken 요소의 일반 텍스트 암호를 처리할 수 있습니다.

암호가 일반 텍스트인 경우 UsernameToken 요소에 다음 하위 요소가 포함됩니다.

사용자 이름 요소

원시 보안 도메인 또는 LDAP 보안 도메인의 사용자 이름을 포함합니다. 기본 보안 도메인은 원시 보안 도메인입니다. 사용자 이름이 원시 보안 도메인에 속하는 경우 Username 요소에는 보안 도메인의 이름이 필요하지 않습니다. 사용자 이름이 LDAP 보안 도메인에 속하는 경우 사용자 이름은 보안 도메인의 이름과 슬래시(/) 뒤에 와야 합니다.

암호 요소

일반 텍스트의 암호를 포함합니다. 암호 요소의 유형 특성을 "PasswordText"로 설정합니다.

다음 샘플 SOAP 헤더는 일반 텍스트 암호가 포함된 UsernameToken 요소의 예를 보여 줍니다.

```
<soap:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
    <wsse:UsernameToken wsu:Id="UsernameToken-14" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
      <wsse:Username>Administrator</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">Administrator</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</soap:Header>
```

해시된 암호

사용자 암호를 암호화해야 할 경우 SOAP 요청의 사용자 이름 토큰 헤더에 해시된 암호를 포함하십시오. 데이터 통합 서비스는 UsernameToken 요소의 해시된 암호를 처리할 수 있습니다.

해시된 암호를 사용할 경우 UsernameToken 요소에 다음 하위 요소가 포함됩니다.

사용자 이름 요소

원시 보안 도메인에 사용자 이름을 포함합니다.

암호 요소

해시된 암호를 포함합니다. 암호를 MD5 또는 SHA-1 해시 함수로 해시화하고 Base64로 인코딩해야 합니다. 암호 요소의 유형 특성을 "PasswordText"로 설정합니다.

다음 샘플 SOAP 헤더에서 해시된 암호가 포함된 UsernameToken 요소 예제를 보여 줍니다.

```
<soap:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
    <wsse:UsernameToken wsu:Id="UsernameToken-14" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
      <wsse:Username>Administrator</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">Ntm58Cxf7SB0QAz30lsTq1nv-D7</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</soap:Header>
```

요약 암호

사용자 암호가 임시 값 및 타임스탬프로 해시된 암호화된 암호인 경우 SOAP 요청의 사용자 이름 토큰 헤더에 요약 암호를 포함하십시오. 데이터 통합 서비스는 UsernameToken 요소의 요약 암호를 처리할 수 있습니다.

요약 암호를 사용할 경우 UsernameToken 요소에 다음 하위 요소가 포함됩니다.

사용자 이름 요소

원시 보안 도메인에 있는 사용자 이름을 포함합니다.

암호 요소

요약 암호를 포함합니다. 암호는 임시 값 요소의 임시 값 및 작성 날짜 요소의 타임스탬프와 연결된 암호를 해시하여 작성된 값입니다. 암호를 SHA-1 해시 함수로 해시화하고 Base64로 인코딩해야 합니다. 요약 암호 보안의 경우 암호 요소의 유형 특성을 "PasswordDigest"로 설정합니다.

임시 값 요소

한 번만 사용할 수 있는 임의의 값인 임시 값을 포함합니다. 기본적으로 작성 날짜 요소의 값으로 표시되는 요청 작성 시간으로부터 300초 동안 유효합니다. 클라이언트 응용 프로그램은 임시 값이 유효한 시간 내에 요청을 전송해야 합니다. 예를 들어, 작성 날짜 값이 요청이 오전 10시에 작성되었음을 표시합니다. 요청은

오전 10시에서 오전 10시 5분까지 유효합니다. 클라이언트 응용 프로그램이 오전 10시 전 또는 오전 10시 5분 후에 웹 서비스에 요청을 전송할 경우, 요청 및 임시 값이 유효하지 않게 되고 요청이 실패합니다.

작성 날짜 요소

요청이 작성된 시간을 나타내는 타임스탬프 값을 포함합니다. 타임스탬프에서는 UTC 형식 yyyy-MM-dd'T'HH:mm:ss.SSS'Z'를 사용합니다. 예: 2008-08-11T18:06:32.425Z.

요약 암호는 표준 OASIS 암호 요약 알고리즘을 사용합니다.

Password_Digest = Base64 (SHA-1 (nonce + created + password))

임의 도구를 사용하여 임시 값, 타임스탬프 및 요약 암호를 생성할 수 있습니다.

다음 샘플 SOAP 헤더에서는 요약 암호가 포함된 UsernameToken 요소 예제를 보여 줍니다.

```
<soap:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
    <wsse:UsernameToken wsu:Id="UsernameToken-14" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
      <wsse:Username>Administrator</wsse:Username>
      <wsse:Password_Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordDigest">Ntm58Cxf7SBOQAz30lsTqInv-D7</wsse:Password>
      <wsse:Nonce_EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">zWELHdoAzNjQQ9xzLIwFZA==</wsse:Nonce>
      <wsu:Created>2010-10-15T20:56:18.633Z</wsu:Created>
    </wsse:UsernameToken>
  </wsse:Security>
</soap:Header>
```

그리드의 웹 서비스

데이터 통합 서비스는 웹 서비스를 그리드에서 실행할 수 있습니다.

데이터 통합 서비스 그리드에서 웹 서비스 요청이 실행되는 경우 서비스는 데이터 통합 서비스 프로세스에서 작업을 실행합니다. 그리드의 모든 노드에 서비스와 계산 역할이 둘 다 있어야 합니다.

그리드에서 실행되는 데이터 통합 서비스를 활성화하면 하나의 서비스 프로세스가 그리드에서 서비스 역할을 가진 각 노드에서 시작됩니다. 데이터 통합 서비스는 서비스 프로세스 하나를 마스터 서비스 프로세스로 지정하고 나머지 서비스 프로세스를 작업자 서비스 프로세스로 지정합니다. 작업자 서비스 프로세스가 시작되면 마스터 서비스 프로세스에 등록되므로 마스터가 작업자를 인식하게 됩니다.

작업자 서비스 프로세스에서는 웹 서비스 작업이 실행됩니다. 마스터 서비스 프로세스는 작업자 서비스 프로세스 역할도 하며 작업을 완료합니다.

웹 서비스 요청을 제출하면 데이터 통합 서비스는 외부 HTTP 로드 균형 조정기를 사용하여 요청을 작업자 서비스 프로세스에 배포합니다. 웹 서비스에 대해 요청을 여러 개 제출하는 경우 데이터 통합 서비스는 라운드 로빈을 사용하여 각 쿼리를 작업자 서비스 프로세스에 디스패치합니다.

서비스 프로세스에서 작업을 실행하는 그리드 구성

데이터 통합 서비스 그리드에서 웹 서비스 요청을 실행하는 경우 관리자는 서비스 프로세스에서 작업을 실행하도록 데이터 통합 서비스를 구성합니다. 관리자는 Administrator 도구에서 데이터 통합 서비스를 구성할 수 있습니다.

관리자는 웹 서비스 작업에 대한 그리드를 생성하고 데이터 통합 서비스를 그리드에 할당합니다. 관리자는 서비스 프로세스에서 작업을 실행하도록 데이터 통합 서비스를 구성하고 로드 균형 조정을 구성합니다. 또한 관리자는 각 노드의 데이터 통합 프로세스 속성과 실행 DTM(Data Transformation 관리자)의 계산 속성을 구성할 수 있습니다.

웹 서비스 로그

데이터 통합 서비스 로그 및 웹 서비스 런타임 로그에서 웹 서비스 로그를 볼 수 있습니다.

데이터 통합 서비스 로그에는 웹 서비스 상태 변경 또는 웹 서비스 실행을 방해하는 오류 같은 웹 서비스 메시지가 포함될 수 있습니다. **Administrator** 도구의 **로그** 탭에서 데이터 통합 서비스 로그를 확인하십시오.

데이터 통합 서비스의 **DTM** 프로세스에서 웹 서비스 작업 매핑을 실행할 경우 웹 서비스 작업 매핑에 대한 메시지가 웹 서비스 런타임 로그에 표시됩니다. 데이터 통합 서비스의 **DTM** 프로세스에서 각 웹 서비스 요청에 대한 로그 파일을 작성합니다. 데이터 통합 서비스 프로세스 로그 위치 내의 **ws** 디렉터리에서 웹 서비스 런타임 로그를 확인하십시오. 기본적으로 다음 디렉터리에서 웹 서비스 런타임 로그를 확인하십시오.

<InformaticaInstallationDir>/tomcat/bin/disLogs/ws

웹 서비스 추적 수준

Administrator 도구에서 **DTM** 프로세스가 웹 서비스 런타임 로그에 기록하는 메시지의 추적 수준을 표시하는 **추적 수준** 속성을 구성하십시오. 웹 서비스에 설정된 추적 수준을 재정의하기 위해 웹 서비스 요청에 **HTTP** 헤더를 구성할 수 있습니다.

웹 서비스 요청은 웹 서비스 추적 수준 구성을 기반으로 런타임 로그를 생성합니다. 웹 서비스 요청에 대한 웹 서비스 추적 수준을 재정의하려면 웹 서비스 **SOAP** 요청의 **HTTP** 헤더에 다음 항목을 포함시키십시오.

WebServiceOptions.traceLevel= <trace_level>. 예를 들어, **HTTP** 헤더를 사용하여 추적 수준을 **FINE**으로 설정하려면 다음 텍스트를 사용하십시오. **WebServiceOptions.traceLevel= fine**

웹 서비스 모니터링

REST 웹 서비스 또는 **SOAP** 웹 서비스를 모니터링하려면 해당 속성, 런타임 통계, 런타임 보고서 및 각 웹 서비스 요청에 대한 정보를 확인합니다.

다음 위치에서 웹 서비스를 모니터링할 수 있습니다.

- **Monitoring** 도구. **Developer tool**에서 **진행률** 보기의 **메뉴** 단추를 클릭하고 **작업 모니터링**을 선택합니다. 웹 서비스를 실행하는 데이터 통합 서비스를 선택하고 **확인**을 클릭합니다. **Monitoring** 도구가 열립니다.
- **Administrator** 도구. **Administrator** 도구에서 웹 서비스를 모니터링하려면 **모니터링** 탭을 클릭합니다.

웹 서비스를 모니터링하면 서비스에 대한 요약 통계 또는 실행 통계를 확인할 수 있습니다.

요약 통계 보기에는 웹 서비스 분포 및 상태에 대한 그래픽 정보가 표시됩니다. 시간 범위를 선택하고 **요청 및 연결** 패널을 확장하면 해당 시간 범위의 웹 서비스 요청이 요약 통계에 표시됩니다. 요청 세부 정보 또는 요청에 대한 요약을 볼 수 있습니다. 그래픽 분포 또는 테이블 형식의 요약을 봅니다. 모델 리포지토리에 저장된 데이터를 사용하여 요약 통계 보기에 통계가 표시됩니다. 요약 통계를 보려면 모니터링 구성에서 모델 리포지토리를 먼저 구성해야 합니다.

실행 통계 보기에는 응용 프로그램에 배포된 특정 웹 서비스에 대한 정보가 표시됩니다. **SOAP** 웹 서비스 또는 **REST** 웹 서비스에 대해 다음과 같은 통계를 볼 수 있습니다.

- 총 요청 수.
- 완료된 요청.
- 중단된 요청. 중단 모드에서 데이터 통합 서비스가 재사용 또는 비활성화되었을 때 중단된 요청입니다.
- 실패한 요청.

웹 서비스의 속성을 보려면 탐색기에서 응용 프로그램을 확장한 후 **웹 서비스** 폴더 또는 **REST 서비스** 폴더를 선택합니다. 웹 서비스 목록이 콘텐츠 패널에 나타납니다. 콘텐츠 패널에는 이름, 설명, 상태 등 각 웹 서비스에 대한 속성이 표시됩니다.

웹 서비스를 선택하면 세부 정보 패널에 다음 보기가 표시됩니다.

- **속성 보기.** 속성 보기에는 웹 서비스에 대한 일반 속성과 런타임 통계가 표시됩니다.
- **보고서 보기.** 보고서 보기에는 선택한 웹 서비스에 대한 모니터링 보고서가 표시됩니다.
- **작업 보기.** 작업 보기에는 웹 서비스에 포함된 작업에 대한 이름과 설명이 표시됩니다. 각 작업에 대한 속성, 요청 및 보고서도 보기에 표시됩니다.
- **요청 보기.** 요청 보기에는 요청 ID, 사용자 이름, 상태, 시작 시간, 경과 시간 및 종료 시간 같은 웹 서비스에 대한 속성이 표시됩니다. 요청 목록을 필터링할 수 있습니다.

웹 서비스의 속성 보기

속성 보기에는 웹 서비스에 대한 일반 속성 및 런타임 통계가 표시됩니다.

속성 보기의 콘텐츠 패널에서 웹 서비스를 선택하면 일반 속성 및 모니터링 통계가 표시됩니다.

웹 서비스의 일반 속성

이름 및 개체 유형 등 웹 서비스에 대한 일반 속성을 볼 수 있습니다.

웹 서비스의 통계

특정 기간 동안의 웹 서비스 요청에 대한 런타임 통계를 볼 수 있습니다. **통계** 섹션에는 웹 서비스 요청의 완료, 실패 및 전체 수가 표시됩니다.

웹 서비스의 보고서 보기

보고서 보기에는 선택한 웹 서비스에 대한 모니터링 보고서가 표시됩니다.

웹 서비스를 모니터링할 때 **보고서** 보기에는 웹 서비스에 대한 보고서가 표시됩니다. 예를 들어 가장 활발한 WebService 클라이언트 IP 보고서를 보면 특정 기간 동안 가장 많은 수의 웹 서비스 요청을 수신한 IP 주소를 확인할 수 있습니다.

REST 또는 SOAP 웹 서비스의 작업 보기

작업 보기에는 웹 서비스에 포함된 각 작업 또는 리소스의 이름 및 설명이 표시됩니다. 각 작업에 대한 속성, 요청 및 보고서도 보기에 표시됩니다.

콘텐츠 패널에서 웹 서비스 작업을 선택하면 세부 정보 패널에 **속성** 보기, **요청** 보기 및 **보고서** 보기가 표시됩니다.

속성 보기

속성 보기에는 선택한 웹 서비스 작업 또는 리소스에 대한 일반 속성 및 통계가 표시됩니다. 일반 속성에는 작업 또는 리소스 이름 및 개체 유형이 포함됩니다. 이 보기에는 특정 기간 동안의 웹 서비스 작업에 대한 통계도 표시됩니다. 통계에는 웹 서비스 요청의 완료, 실패 및 전체 수가 포함됩니다.

요청 보기

요청 보기에는 요청 ID, 사용자 이름, 상태, 시작 시간, 경과된 시간 및 종료 시간 등 각 웹 서비스 작업에 대한 속성이 표시됩니다. 사용자는 요청 목록을 필터링할 수 있습니다. 또한 선택한 웹 서비스 요청에 대한 로그를 볼 수도 있습니다.

SOAP 웹 서비스의 보고서 보기

보고서 보기에는 SOAP 웹 서비스 작업에 대한 보고서가 표시됩니다.

웹 서비스의 요청 보기

요청 보기에는 요청 ID, 사용자 이름, 상태, 시작 시간, 경과된 시간, 종료 시간 등 각 웹 서비스 요청에 대한 속성이 표시됩니다. 사용자는 요청 목록을 필터링할 수 있습니다.

콘텐츠 패널에서 웹 서비스 요청을 선택하면 요청에 대한 로그가 세부 정보 패널에 표시됩니다. 세부 정보 패널에는 선택한 웹 서비스 요청에 대한 일반 속성 및 통계가 표시됩니다. 통계에는 웹 서비스 요청의 완료, 실패 및 전체 수가 포함됩니다.

요청 보기에서 웹 서비스 요청을 중단할 수도 있습니다. 웹 서비스 요청을 중단하려면 콘텐츠 패널에서 워크플로우 요청을 선택하고 **작업 > 선택한 요청 중단**을 클릭합니다.

부록 A

데이터 유형 호환성

이 부록에 포함된 항목:

- [데이터 유형 참조 개요, 184](#)
- [XML 및 변환 데이터 유형, 185](#)
- [10진수, 186](#)

데이터 유형 참조 개요

매핑을 작성할 때 데이터 통합 서비스가 소스의 데이터를 읽고, 해당 데이터를 변환하여, 대상에 데이터를 쓰는 지침 집합을 작성합니다. 데이터 통합 서비스는 매핑의 첫 번째 변환에서 시작하여 매핑의 데이터 흐름 및 매핑의 각 포트에 할당된 데이터 유형을 기반으로 데이터를 변환합니다.

Developer tool에는 다음 두 가지 유형의 데이터 유형이 표시됩니다.

원시 데이터 유형

원시 데이터 유형은 실제 데이터 개체로 사용되는 관계형 테이블 또는 플랫폼 파일과 관련됩니다. 원시 데이터 유형은 실제 데이터 개체의 열 속성에 표시됩니다.

변환 데이터 유형

변환 데이터 유형은 변환에 표시되는 데이터 유형의 집합입니다. ANSI SQL-92 일반 데이터 유형에 기반하는 내부 데이터 유형으로, 데이터 통합 서비스는 이 데이터 유형을 사용하여 여러 플랫폼에서 데이터를 이동합니다. 변환 데이터 유형은 매핑의 모든 변환에 표시됩니다.

변환 데이터 유형에는 다음 데이터 유형이 포함됩니다.

- 기본 데이터 유형. 단일 열 위치에 있는 단일 데이터 값을 나타냅니다.
- 복합 데이터 유형. 단일 열 위치에 있는 여러 데이터 값을 나타냅니다. 복합 파일의 계층 데이터를 처리하려면 Spark 엔진에서 실행되는 매핑에 복합 데이터 유형을 사용합니다.

데이터 통합 서비스는 소스 데이터를 읽을 때 원시 데이터 유형을 비교 가능한 변환 데이터 유형으로 변환한 후 데이터를 변환합니다. 데이터 통합 서비스가 대상에 쓸 때는 변환 데이터 유형을 비교 가능한 원시 데이터 유형으로 변환합니다.

다중 바이트 문자 집합을 지정하는 경우 데이터 유형은 최대 3바이트의 문자를 저장할 수 있는 추가 공간을 데이터베이스에 할당합니다.

XML 및 변환 데이터 유형

XML 데이터 유형은 데이터 통합 서비스가 플랫폼에서 데이터를 이동하기 위해 사용하는 변환 데이터 유형에 매핑됩니다.

데이터 통합 서비스는 W3C 권장 사항(2001년 5월 2일)에 지정된 모든 XML 데이터 유형을 지원합니다. 그러나 데이터 통합 서비스에서 전체 XML 값 범위를 지원하지 않을 수 있습니다. XML 데이터 유형에 대한 자세한 정보는 다음 위치에서 XML 데이터 유형에 대한 W3C 사양을 참조하십시오. <http://www.w3.org/TR/xmlschema-2>.

다음 테이블에서는 XML 데이터 유형을 변환 데이터 유형과 비교합니다.

데이터 유형	변환	범위
anyURI	문자열	1 ~ 104,857,600자
base64Binary	2진	1~104,857,600바이트
부울	문자열	1 ~ 104,857,600자
바이트	정수	-2,147,483,648 ~ 2,147,483,647
날짜	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
날짜/시간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
10진수	10진수	1 ~ 28 전체 자릿수, 배율 0 ~ 28
배정밀도	배정밀도	전체 자릿수 15자리
기간	문자열	1 ~ 104,857,600자
ENTITIES	문자열	1 ~ 104,857,600자
ENTITY	문자열	1 ~ 104,857,600자
부동 소수점 수	배정밀도	전체 자릿수 15자리
gDay	문자열	1 ~ 104,857,600자
gMonth	문자열	1 ~ 104,857,600자
gMonthDay	문자열	1 ~ 104,857,600자
gYear	문자열	1 ~ 104,857,600자
gYearMonth	문자열	1 ~ 104,857,600자
hexBinary	2진	1~104,857,600바이트
ID	문자열	1 ~ 104,857,600자
IDREF	문자열	1 ~ 104,857,600자
IDREFS	문자열	1 ~ 104,857,600자
int	정수	-2,147,483,648 ~ 2,147,483,647

데이터 유형	변환	범위
정수	정수	-2,147,483,648 ~ 2,147,483,647
언어	문자열	1 ~ 104,857,600자
긴 정수	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
이름	문자열	1 ~ 104,857,600자
NCName	문자열	1 ~ 104,857,600자
negativeInteger	정수	-2,147,483,648 ~ 2,147,483,647
NMTOKEN	문자열	1 ~ 104,857,600자
NMTOKENS	문자열	1 ~ 104,857,600자
nonNegativeInteger	정수	-2,147,483,648 ~ 2,147,483,647
nonPositiveInteger	정수	-2,147,483,648 ~ 2,147,483,647
normalizedString	문자열	1 ~ 104,857,600자
NOTATION	문자열	1 ~ 104,857,600자
positiveInteger	정수	-2,147,483,648 ~ 2,147,483,647
QName	문자열	1 ~ 104,857,600자
짧은 정수	정수	-2,147,483,648 ~ 2,147,483,647
문자열	문자열	1 ~ 104,857,600자
시간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
토큰	문자열	1 ~ 104,857,600자
unsignedByte	정수	-2,147,483,648 ~ 2,147,483,647
unsignedInt	정수	-2,147,483,648 ~ 2,147,483,647
unsignedLong	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
unsignedShort	정수	-2,147,483,648 ~ 2,147,483,647

10진수

웹 서비스 작업 매핑에 정밀도가 28자리보다 높은 10진수 데이터 유형의 입력 또는 출력 변환이 포함된 경우 데이터 통합 서비스가 10진수 데이터 유형을 배정밀도 데이터 유형으로 전환합니다.

인덱스

A

anyAttribute 특성
웹 서비스 소비자 변환 [86](#), [89](#)
anyType
포트 매핑 [79](#)
anyType 요소
구문 분석 [66](#)
웹 서비스 소비자 변환 [86](#), [89](#)
attributeFormDefault
스키마 개체 [31](#)

B

패턴 속성
스키마 개체 [27](#)
포트
SOAP 메시지에 매핑 [74](#)
비정규화된 웹 서비스 입력 [77](#)
포트 탭
결함 변환 [55](#)
입력 변환 [50](#)
출력 변환 [52](#)
푸시인 최적화
웹 서비스 소비자 변환 [96](#)
피벗된 데이터
SOAP 메시지 [76](#)
피벗된 출력
SOAP 메시지 구문 분석 [65](#)
필터 포트
웹 서비스 소비자 변환 [96](#)
합집합 요소
SOAP 메시지 구문 분석 [69](#)
설명 [81](#)
헤더
HTTP POST [47](#)
생성 [47](#)
설명 [15](#)

E

elementFormDefault
스키마 개체 [31](#)

G

GZip
SOAP 메시지 압축 [95](#)

H

HTTP POST
헤더 [47](#)
HTTP 연결
REST 웹 서비스 [148](#)
HTTP 오류 출력
웹 서비스 소비자 변환에서 활성화 [92](#)
HTTP 응답 코드
REST 웹 서비스 소비자 변환 [144](#)
HTTP 헤더
REST 웹 서비스 소비자 변환에 추가 [143](#)
웹 서비스 소비자 변환에 추가 [86](#)

I

isAuthenticationRequired
REST 웹 서비스 [176](#)

N

nillible 속성
스키마 개체 [27](#)
네임스페이스
생성된 접두사 변경 [25](#)
스키마 개체 [26](#)
노드
웹 서비스 결함 처리 [57](#)
다양성 속성
스키마 개체 [28](#)
단순 유형
스키마 개체 [29](#)
단순 유형 보기
스키마 개체 [28](#)
대체 그룹
SOAP 메시지 구문 분석 [68](#)
스키마 개체 [28](#)
웹 서비스 [79](#)
웹 서비스 소비자 변환 [86](#), [89](#)
데이터 액세스 방법
REST 웹 서비스 미리보기 [114](#)
데이터 유형
XML [185](#)
개요 [184](#)
데이터 통합 서비스 그리드
웹 서비스 [180](#)
동시 웹 서비스 요청 메시지
웹 서비스 소비자 변환에서 활성화 [92](#)
동적 URL
웹 서비스 소비자 변환 [86](#)
리소스
REST 웹 서비스 [103](#), [105](#)

- 리소스 동기화
 - REST 웹 서비스 [107](#)
- 리소스 매핑
 - REST 웹 서비스 [108](#)
 - 고급 탭 [113](#)
 - 기본값 [108](#)
 - 사용자 지정 [109](#)
 - 설명 [103](#)
- 리소스 식별자
 - 데이터 미리보기에서 필터링 [114](#)
- 리소스 키
 - REST 웹 서비스 [107](#)
 - 검색 기준 [112](#)
 - 요청 URI에서 참조 [112](#)
- 멤버 유형
 - 스키마 개체 [28](#)
- 모든 그룹
 - REST 웹 서비스 소비자 변환에서 보기 [147](#)
 - 웹 서비스 소비자 변환에서 보기 [88, 91](#)
- 모든 요소
 - 웹 서비스 소비자 변환 [86, 89](#)
- 목록 요소
 - SOAP 메시지 구문 분석 [69](#)
 - 설명 [80](#)
- 바인딩
 - WSDL 파일 요소 [83](#)
- 방법
 - 데이터 개체를 REST 웹 서비스로 배포 [133](#)
 - 데이터 개체에서 REST 웹 서비스 생성 [130](#)
- 보기 옵션 사용자 지정
 - 설명 [61](#)
- 보안
 - UsernameToken [178](#)
 - 웹 서비스 보안 [176](#)
- 복합 유형
 - 고급 속성 [29](#)
- 복합 키
 - REST 웹 서비스 소비자 변환 [145](#)
 - 웹 서비스 소비자 변환 [86](#)
- 부작용
 - 웹 서비스 소비자 변환 [96](#)
- 블록 속성
 - 스키마 개체 [28](#)
- 비정규화된 입력
 - 웹 서비스 포트 [77](#)
- 비정규화된 출력
 - SOAP 메시지 구문 분석 [65](#)
- 사용자 지정 리소스 매핑
 - 설명 [109](#)
- 사전 정의된 결함
 - 생성 [47](#)
 - 웹 서비스 결함 처리 [59](#)
- 생성
 - 결함 변환 [56](#)
- 생성된 접두사
 - 네임스페이스 변경 [25](#)
- 생성된 키
 - 웹 서비스 출력 그룹 [64](#)
- 서비스
 - WSDL 파일 요소 [83](#)
- 선택 요소
 - REST 웹 서비스 소비자 변환에서 보기 [147](#)
 - SOAP 메시지 구문 분석 [68](#)
 - 설명 [80](#)
 - 웹 서비스 소비자 변환에서 보기 [88, 91](#)
- 세부 정보 요소
 - 웹 서비스 결함 처리 [57](#)

- 속성에서 상속
 - 스키마 개체 [29](#)
- 스키마 개체
 - 요소의 고급 속성 [28](#)
 - attributeFormDefault [31](#)
 - elementFormDefault [31](#)
 - 가져오기 [31](#)
 - 개요 [24](#)
 - 개요 보기 [24](#)
 - 기본 편집기 설정 [33](#)
 - 네임스페이스 [26](#)
 - 단순 유형 [28](#)
 - 대체 그룹 [28](#)
 - 동기화 [32](#)
 - 복합 요소 [29](#)
 - 복합 요소의 고급 속성 [29](#)
 - 블록 속성 [28](#)
 - 속성에서 상속 [29](#)
 - 스키마 보기 [25](#)
 - 스키마 파일 [25](#)
 - 스키마 파일 편집 [32](#)
 - 요소 속성 [27](#)
 - 추상 속성 [27](#)
 - 특성 속성 [30](#)
 - 파일 위치 [31](#)
- 스키마 보기
 - REST 웹 서비스 [107](#)
 - 단순 유형 고급 속성 [29](#)
 - 스키마 개체 [25](#)
- 스키마 파일
 - 기본 편집기 설정 [33](#)
 - 스키마 개체에 추가 [25](#)
 - 스키마 개체에서 제거 [25](#)
 - 편집 [34](#)
- 시스템 정의된 결함
 - 웹 서비스 결함 처리 [59](#)
- 시퀀스 그룹
 - REST 웹 서비스 소비자 변환에서 보기 [147](#)
- 암호
 - 일반 텍스트 [178](#)
 - 해시됨 [179](#)
- 역할
 - 웹 서비스 결함 처리 [57](#)
- 연결
 - REST 웹 서비스 [148](#)
 - 웹 서비스 [92](#)
- 열거 속성
 - 스키마 개체 [27](#)
- 요소
 - 생성 [46](#)
 - 합집합 [81](#)
- 요청 메시지
 - REST 웹 서비스 [111](#)
- 웹 서비스
 - anyType에 포트 매핑 [79](#)
 - SOAP 예제 [17](#)
 - WSDL [16](#)
 - WSDL URL [16](#)
 - WSDL 보기 [40](#)
 - WSDL 없이 생성 [42](#)
 - WSDL 연관 [40](#)
 - WSDL에서 작성 [40](#)
 - 개요 [12](#)
 - 개요 보기 [38](#)
 - 결과 집합 캐싱 구성 [175](#)
 - 결함 변환 구성 [56](#)
 - 결함 처리 [57](#)

웹 서비스 (계속)

- 구성 요소 [15](#)
- 대체 그룹 [79](#)
- 데이터 통합 서비스 그리드 [180](#)
- 로그 [181](#)
- 리소스 속성 [175](#)
- 모니터링 [181](#)
- 배포 [17](#)
- 보안 [176](#)
- 사용 권한 [177](#)
- 사용 권한 유형 [177](#)
- 사전 정의된 결함 [59](#)
- 사전 정의된 결함 작성 [47](#)
- 속성 [173](#)
- 속성 구성 [172](#)
- 수동으로 생성 [42](#)
- 시스템 정의된 결함 [59](#)
- 요소 작성 [46](#)
- 일반 결함 [60](#)
- 입력 변환 [49](#)
- 입력 변환 구성 [50](#)
- 작업 [15](#)
- 작업 속성 [175](#)
- 추적 수준 [181](#)
- 출력 변환 구성 [53](#)
- 파생된 유형 [78](#)
- 프로세스 [13](#)
- 헤더 작성 [47](#)

웹 서비스 변환

- 위치 열 [72](#)

웹 서비스 보안

- HTTP 클라이언트 필터 [176](#)
- HTTPS [176](#)
- isAuthenticationRequired [176](#)
- TLS(Transport Layer Security) [176](#)
- 권한 부여 [176](#)
- 메시지 레이어 보안 [176](#)
- 사용 권한 [176](#)
- 인증 [176](#)
- 통과 보안 [176](#)

웹 서비스 소비자

- 프로세스 [14](#), [104](#)

웹 서비스 소비자 변환

- HTTP 오류 출력 활성화 [92](#)
- HTTP 헤더 추가 [86](#)
- SOAP 메시지 [82](#)
- SOAP 압축 [95](#)
- TLS(Transport Layer Security) [84](#)
- 개요 [82](#)
- 고급 속성 [92](#)
- 끝점 URL [86](#)
- 동시 웹 서비스 요청 메시지 [92](#)
- 동적 WS-Security 이름 [86](#)
- 동적 웹 서비스 URL [86](#)
- 보안 [84](#)
- 오류 처리 [94](#)
- 일반 SOAP 결함 [94](#)
- 일반 결함 출력 활성화 [92](#)
- 입력 매핑 [86](#)
- 입력 포트 매핑 [86](#)
- 작성 [98](#)
- 작업 [83](#)
- 초기 선택 최적화 [96](#)
- 출력 노드 매핑 [89](#)
- 출력 매핑 [89](#)
- 쿠키 인증 [86](#)
- 키 보기 [88](#), [91](#)

웹 서비스 소비자 변환 (계속)

- 푸시인 최적화 [96](#)
- 푸시인 최적화 활성화 [98](#)
- 필터 최적화 [96](#)

웹 서비스 연결

- 개요 [92](#)

위치 열

- 웹 서비스 변환 [72](#)

응답 메시지

- REST 웹 서비스 [113](#)

응답 코드

- REST 웹 서비스 소비자 변환 [144](#)

이유

- 웹 서비스 결함 처리 [57](#)

인증

- UsernameToken [178](#)

인증서

- 인증서 관리 [23](#), [35](#)
- 인증서 속성 [23](#), [36](#)
- 트러스트되지 않은 인증서 [23](#), [35](#)
- 트러스트되지 않은 인증서 추가 [23](#), [36](#)

일반 SOAP 결함

- 웹 서비스 소비자 변환 [94](#)

일반 결함

- 웹 서비스 결함 처리 [60](#)

일반 결함 출력

- 웹 서비스 소비자 변환에서 활성화 [92](#)

입력 매핑

- REST 웹 서비스 소비자 변환 [144](#)
- 웹 서비스 소비자 변환 [86](#)

입력 변환

- XML로 입력에 매핑 [49](#)
- 계층의 첫 번째 수준 매핑 [50](#)
- 매핑 규칙 및 지침 [50](#)
- 보기 옵션 사용자 지정 [61](#)
- 설명 [49](#)
- 웹 서비스에서 구성 [50](#)
- 포트 탭 [50](#)

입력 포트 영역

- SOAP 메시지 생성 [71](#)

작업

- WSDL 파일 요소 [83](#)
- 설명 [15](#)

작업 결함

- 설명 [15](#)

작업 매핑

- 개요 [48](#)
- 결함 변환 [54](#)
- 입력 변환 [49](#)
- 출력 변환 [52](#)
- 테스트 [61](#)

작업 영역

- 웹 서비스 변환 [72](#)

작업 입력

- 설명 [15](#)

작업 입력 영역

- 웹 서비스 소비자 변환 사용자 지정 [88](#)

작업 출력

- 설명 [15](#)

작업 출력 영역

- 웹 서비스 소비자 변환 사용자 지정 [91](#)

첫 번째 수준 계층 매핑

- 설명 [109](#)

초기 선택 최적화

- 웹 서비스 소비자 변환 [96](#)

최대 길이

- 스키마 개체 [27](#)

- 최대 발생
 - 스키마 개체 [27](#)
- 최소 길이
 - 스키마 개체 [27](#)
- 최소 발생
 - 스키마 개체 [27](#)
- 추상 속성
 - 스키마 개체 [27](#)
- 추적 수준
 - 웹 서비스 [181](#)
- 출력 매핑
 - REST 웹 서비스 [109](#)
 - REST 웹 서비스 소비자 변환 [146](#)
 - 여러 그룹 [110](#)
 - 웹 서비스 소비자 변환 [89](#)
- 출력 변환
 - XML로 출력에 매핑 [52](#)
 - 계층의 첫 번째 수준 매핑 [52](#)
 - 고급 탭 [52](#)
 - 매핑 규칙 및 지침 [52](#)
 - 보기 옵션 사용자 지정 [61](#)
 - 설명 [52](#)
 - 웹 서비스에서 구성 [53](#)
 - 포트 탭 [52](#)
- 코드
 - 웹 서비스 결함 처리 [57](#)
- 쿠키 인증
 - REST 웹 서비스 소비자 변환 [144](#)
 - 웹 서비스 소비자 변환 [86](#)
- 키
 - SOAP 메시지 계층 [73](#)
- 특성 속성
 - 스키마 개체 [30](#)
- 파생된 유형
 - SOAP 메시지 구문 분석 [67](#)
 - 웹 서비스 [78](#)
- 파생된 유형 요소
 - 웹 서비스 소비자 변환 [86, 89](#)

Q

- Qname 요소
 - SOAP 메시지 구문 분석 [68](#)

R

- REST 및 SOAP
 - 비교 [13](#)
- REST 웹 서비스
 - 개요 [103, 112](#)
 - 기본 매핑 [108](#)
 - 데이터 뷰어 보기 [114](#)
 - 리소스 동기화 [107](#)
 - 리소스 매핑 [108](#)
 - 리소스 키 [107](#)
 - 리소스 키를 기준으로 쿼리 [112](#)
 - 사용자 지정 매핑 [109](#)
 - 스키마 보기 [107](#)
 - 여러 번 발생하는 데이터 [110](#)
 - 요청 메시지 형식 [111](#)
 - 웹 서비스 생성 [116](#)
 - 응답 메시지 형식 [113](#)
 - 출력 변환 [109](#)
 - 프로세스 [104](#)

- REST 웹 서비스 생성
 - 데이터 개체 배포 [133](#)
 - 데이터 개체에서 생성 [130](#)
- REST 웹 서비스 소비자 변환
 - Delete 메서드 [141](#)
 - Get 메서드 [139](#)
 - HTTP 메서드 [139](#)
 - HTTP 헤더 포트 [143](#)
 - JSON 스키마 [165](#)
 - Post 메서드 [140](#)
 - Put 메서드 [141](#)
 - RequestInput 포트 [142](#)
 - TLS(Transport Layer Security) [84](#)
 - URL 포트 [143, 168](#)
 - XML 스키마 [159](#)
 - XML 스키마 유효성 검사 [148](#)
 - 개요 [136](#)
 - 고급 속성 [148](#)
 - 구문 분석되지 않은 데이터 [153](#)
 - 구문 분석되지 않은 [153](#)
 - 구문 분석된 데이터 [155](#)
 - 구문 분석됨 [153](#)
 - 구성 [138](#)
 - 기본 URL 설정 [148](#)
 - 리소스 식별 [138](#)
 - 매핑 입력 [136](#)
 - 매핑 출력 [136](#)
 - 메시지 구성 [138](#)
 - 보안 [84](#)
 - 사용자 지정 URL [168](#)
 - 스키마 [159, 165](#)
 - 연결 속성 [148](#)
 - 응답 코드 포트 [144](#)
 - 인수 포트 [143, 168](#)
 - 인터넷 미디어 유형 [148](#)
 - 입력 매핑 [144](#)
 - 입력 매핑 규칙 [145](#)
 - 입력 포트 [142](#)
 - 입력 포트 매핑 [145](#)
 - 작성 [149](#)
 - 재사용 가능 [149](#)
 - 재사용 불가능 [149](#)
 - 정렬된 입력 [148](#)
 - 추적 수준 [148](#)
 - 출력 XML 포트 [144](#)
 - 출력 매핑 [146](#)
 - 출력 매핑 규칙 [147](#)
 - 출력 매핑 보기 사용자 지정 [147](#)
 - 출력 포트 [142](#)
 - 출력 포트 매핑 [148](#)
 - 쿠키 포트 [144](#)
 - 통과 포트 [143](#)
 - 포트 [142](#)
 - 포트에 요소 매핑 [142](#)
 - 프로세스 [137](#)
 - 프록시 서버 지원 [136](#)

S

- SOAP 1.1
 - 결함 요소 [57](#)
- SOAP 1.2
 - 결함 요소 [57](#)
- SOAP 계층
 - 입력 포트에 대한 관계 [72](#)

SOAP 메시지

anyType 요소 구문 분석 [66](#)

개요 [82](#)

다중 발생 노드 매핑 [63](#)

대체 그룹 구문 분석 [68](#)

데이터 피벗 [76](#)

목록 요소 구문 분석 [69](#)

목록 요소 매핑 [80](#)

선택 요소 구문 분석 [68](#)

선택 요소 매핑 [80](#)

여러 입력 포트 매핑 [75](#)

키 [73](#)

포트 매핑 [74](#)

포트를 합집합 요소에 매핑 [81](#)

SOAP 메시지 구문 분석

Qname 요소 [68](#)

비정규화된 출력 [65](#)

설명 [62](#)

정규화된 출력 [64](#)

파생된 유형 [67](#)

피벗된 출력 [65](#)

합집합 요소 [69](#)

SOAP 및 REST

비교 [13](#)

SOAP 압축

웹 서비스 소비자 변환 [95](#)

SOAP 웹 서비스

개발 [17](#)

생성 개요 [37](#)

SOAP 작업

웹 서비스 소비자 변환에서 재정의 [92](#)

SOAP 작업 재정의

웹 서비스 소비자 변환 [92](#)

T

TLS(Transport Layer Security)

REST 웹 서비스 소비자 변환 [84](#)

웹 서비스 소비자 변환 [84](#)

U

UsernameToken

암호 요약 [179](#)

암호 유형 [178](#)

일반 텍스트 암호 [178](#)

해시된 암호 [179](#)

W

WS-Security 사용자 이름

동적 포트 [86](#)

WSDL

WSDL에서 웹 서비스 작성 [40](#)

설명 [16](#)

웹 서비스와 연관 [40](#)

WSDL URL

설명 [16](#)

WSDL 데이터 개체

가져오기 [19](#)

개요 보기 [20](#)

고급 보기 [21](#)

동기화 [22](#)

스키마 보기 [19](#)

WSDL 데이터 개체 (계속)

작성 [21](#)

WSDL 파일

바인딩 요소 [83](#)

서비스 요소 [83](#)

작업 요소 [83](#)

포트 요소 [83](#)

X

XML로 입력에 매핑

입력 변환 [49](#)

XML로 출력에 매핑

결함 변환 [54](#)

출력 변환 [52](#)

결과

결과 집합 개성

구성 [175](#)

웹 서비스 작업 속성 [175](#)

결함 문자열

웹 서비스 결함 처리 [57](#)

결함 변환

XML로 출력에 매핑 [54](#)

계층의 첫 번째 수준 매핑 [55](#)

매핑 규칙 및 지침 [55](#)

보기 옵션 사용자 지정 [61](#)

사전 정의된 결함 [56](#)

생성 [56](#)

설명 [54](#)

웹 서비스에서 구성 [56](#)

일반 결함 [56](#)

포트 탭 [55](#)

결함 요소

SOAP 1.1 [57](#)

SOAP 1.2 [57](#)

결함 작업자

웹 서비스 결함 처리 [57](#)

결함 코드

웹 서비스 결함 처리 [57](#)

결함을 오류로 처리

웹 서비스 소비자 변환에서 활성화 [92](#)

계층의 첫 번째 수준 매핑

결함 변환 [55](#)

입력 변환 [50](#)

출력 변환 [52](#)

고급 속성

REST 웹 서비스 소비자 변환 [148](#)

웹 서비스 소비자 변환 [92](#)

고급 탭

출력 변환 [52](#)

고정값 속성

스키마 개체 [27](#)

공백 축소 속성

스키마 개체 [28](#)

기본 속성

스키마 개체 [28](#)

기본 인증

REST 웹 서비스 [176](#)



끝점 URL

REST 웹 서비스 소비자 변환 [143](#)

끝점 URL (계속)

웹 서비스 소비자 변환 [86](#)