



Informatica® Cloud Data Validation  
July 2025

# Test Cases

Informatica Cloud Data Validation Test Cases  
July 2025

© Copyright Informatica LLC 2023, 2025

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

Informatica, Informatica Cloud, Informatica Intelligent Cloud Services, PowerCenter, PowerExchange, 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.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at [infa\\_documentation@informatica.com](mailto:infa_documentation@informatica.com).

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2025-07-10

# Table of Contents

<b>Preface .....</b>	<b>4</b>
Informatica Resources. ....	4
Informatica Documentation. ....	4
Informatica Intelligent Cloud Services web site. ....	4
Informatica Intelligent Cloud Services Communities. ....	4
Informatica Intelligent Cloud Services Marketplace. ....	4
Data Integration connector documentation. ....	5
Informatica Knowledge Base. ....	5
Informatica Intelligent Cloud Services Trust Center. ....	5
Informatica Global Customer Support. ....	5
 <b>Chapter 1: Test cases.....</b>	<b>6</b>
Comparison methods. ....	7
Null handling. ....	7
Parameterization for WHERE clause. ....	8
Parameter files. ....	8
Parameter file requirements. ....	8
Parameter scope. ....	9
Creating a test case. ....	9
Step 1. Define test case details. ....	9
Step 2. Select connections and data sources. ....	10
Step 3. Select table columns and primary keys. ....	12
Step 4. Map unmapped columns. ....	12
Step 5. Configure test case properties (optional). ....	13
Rules and guidelines for data sampling. ....	14
Viewing test case details. ....	15
Running a test case. ....	15
Deleting a test case job. ....	15
 <b>Index.....</b>	<b>16</b>

# Preface

Read *Test Cases* to learn how to create repeatable tests that validate the accuracy and completeness of data migration.

## Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

### Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit <https://docs.informatica.com>.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at [infa\\_documentation@informatica.com](mailto:infa_documentation@informatica.com).

### Informatica Intelligent Cloud Services web site

You can access the Informatica Intelligent Cloud Services web site at <http://www.informatica.com/cloud>. This site contains information about Informatica Cloud integration services.

### Informatica Intelligent Cloud Services Communities

Use the Informatica Intelligent Cloud Services Community to discuss and resolve technical issues. You can also find technical tips, documentation updates, and answers to frequently asked questions.

Access the Informatica Intelligent Cloud Services Community at:

<https://network.informatica.com/community/informatica-network/products/cloud-integration>

Developers can learn more and share tips at the Cloud Developer community:

<https://network.informatica.com/community/informatica-network/products/cloud-integration/cloud-developers>

### Informatica Intelligent Cloud Services Marketplace

Visit the Informatica Marketplace to try and buy Data Integration Connectors, templates, and mapplets:

<https://marketplace.informatica.com/>

## Data Integration connector documentation

You can access documentation for Data Integration Connectors at the Documentation Portal. To explore the Documentation Portal, visit <https://docs.informatica.com>.

## Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit <https://search.informatica.com>. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at [KB\\_Feedback@informatica.com](mailto:KB_Feedback@informatica.com).

## Informatica Intelligent Cloud Services Trust Center

The Informatica Intelligent Cloud Services Trust Center provides information about Informatica security policies and real-time system availability.

You can access the trust center at <https://www.informatica.com/trust-center.html>.

Subscribe to the Informatica Intelligent Cloud Services Trust Center to receive upgrade, maintenance, and incident notifications. The [Informatica Intelligent Cloud Services Status](#) page displays the production status of all the Informatica cloud products. All maintenance updates are posted to this page, and during an outage, it will have the most current information. To ensure you are notified of updates and outages, you can subscribe to receive updates for a single component or all Informatica Intelligent Cloud Services components. Subscribing to all components is the best way to be certain you never miss an update.

To subscribe, on the [Informatica Intelligent Cloud Services Status](#) page, click **SUBSCRIBE TO UPDATES**. You can choose to receive notifications sent as emails, SMS text messages, webhooks, RSS feeds, or any combination of the four.

## Informatica Global Customer Support

You can contact a Global Support Center through the Informatica Network or by telephone.

To find online support resources on the Informatica Network, click **Contact Support** in the Informatica Intelligent Cloud Services Help menu to go to the **Cloud Support** page. The **Cloud Support** page includes system status information and community discussions. Log in to Informatica Network and click **Need Help** to find additional resources and to contact Informatica Global Customer Support through email.

The telephone numbers for Informatica Global Customer Support are available from the Informatica web site at <https://www.informatica.com/services-and-training/support-services/contact-us.html>.

# CHAPTER 1

## Test cases

Data validation testing is a process to check whether a set of data is valid and complete. Data Validation test cases test the accuracy and validity of the data by selecting and comparing two data sets.

When you create a test case, you select the connections that contain the data to compare, the data sources to compare, and the columns to compare. The data sources can be database tables, views, or saved SQL queries. Data Validation maps the columns in the two data sources. If any column remains unmapped, you can map it manually.

You can also choose to compare the entire table based on the actual data values or use aggregation functions to compare the table.

Optionally, you can configure the following options:

### **Bad record limit**

You can define the maximum number of unmatched, extra, and missing records to show in the detailed test results.

### **String validation**

You can specify whether you want to ignore casing differences in the data and trim leading and trailing white spaces in string values.

### **Data sampling level**

You can configure data sampling for Data Validation to compare between source and target data. The less data that Data Validation samples, the faster the test runs. This setting applies to all the columns you select to compare, in both tables.

### **Schedule**

You can select a schedule to run the test case. Create schedules in Administrator.

### **Email notifications**

You can specify whether you want to send email notifications when the test case completes successfully or fails. You can either use the default email notification options that are configured in Administrator or define custom email notifications. When you use custom email notifications, you can specify the email addresses that Data Validation must use for the success and failure email notifications. Separate multiple email addresses with a comma.

### **Logging level**

You can configure a test case to use one of the following logging levels:

- Standard. The log includes error messages.
- Verbose. The log includes messages of all logging levels. This option is useful for debugging.

Before you run test cases, verify that the Secure Agent that runs them has a minimum of 32 GB memory.

When you run the test case, Data Validation compares each column in the source with the column in the target that is mapped to it and reports whether or not the two data sets match. For unmatched data sets, the report lists unmatched, missing, and extra records.

## Comparison methods

When you compare data between two connections, you can use one of the following comparison methods:

### **Value Test - Compare Entire Table**

If you choose the **Value Test - Compare Entire Table** method, Data Validation compares the entire table based on the actual data values. This is the default method.

### **Aggregation Functions Test**

If you choose the **Aggregation Functions Test** method, you can specify the aggregation functions that Data Validation must use to compare the table.

An aggregation functions test retrieves summarized information about the data contained in the data sources. Use aggregation to verify whether all records were moved or to identify incorrect logic in WHERE clauses.

Based on the data type of the columns, you can select one or more of the following aggregation functions:

- COUNT. Counts the number of rows that contain non-null values for a string or numeric column.
- COUNT\_ROWS. Counts the number of rows for a string or numeric column. Includes rows that contain nulls.
- MIN. Calculates the minimum value for a numeric column.
- MAX. Calculates the maximum value for a numeric column.
- AVG. Calculates the average value of a numeric column.
- SUM. Calculates the total value of a numeric column.

**Note:** If either of the tables or columns that you compare doesn't contain data, the **Value of Comparison** column shows the value as **NIL**. If both the tables or columns that you compare don't contain data, the **Test Result** column shows the value as **No Comparable Data**.

## Null handling

Data Validation treats null values as nulls.

However, if you use a Netezza connection, you must set the `syncNetezzaNullWithPCNull` and `treatNoDataAsEmptyString` custom properties to configure Data Validation to treat null values as nulls. For more information about these custom properties, see the [Informatica Knowledge Base article 531184](#).

If you use an Amazon Redshift v2 connection, Data Validation treats empty strings also as nulls.

When you use a Salesforce connection to compare two tables with identical column names, one table might display null data incorrectly even though it contains data. You can duplicate the Salesforce connection and use a different connection for each source to make sure the data is not mistaken for null. You must avoid using the same Salesforce connection for both tables or sources.

# Parameterization for WHERE clause

When you use a WHERE clause to enable data sampling for a Data Validation test case, you can parameterize the WHERE clause to provide different values at run time.

You can download a sample parameter file from the test case creation wizard. Then, define the parameter names and values in the file. Prefix the parameter name with the \$ character. Save the file in a directory relative to your Secure Agent. Specify the parameter file directory and parameter file name when you create the test case.

The parameter file must be in the UTF-8 format and the file size can't exceed 1 MB.

## Parameter files

A parameter file is a list of user-defined parameters and their associated values.

Use a parameter file to define values that you want to update without having to edit the test case. You update the values in the parameter file instead of updating values in a test case. The parameter values are applied when the test case runs.

You enter the parameter file directory and parameter file name when you configure the test case.

## Parameter file requirements

You can reuse parameter files across test cases. To reuse a parameter file, define local and global parameters within a parameter file.

You group parameters in different sections of the parameter file. Each section is preceded by a heading that identifies the project, folder, and test case to which you want to apply the parameter values. You define parameters directly below the heading, entering each parameter on a new line.

The following table describes the headings that define each section in the parameter file and the scope of the parameters that you define in each section:

Heading	Description
#USE_SECTIONS	Tells Data Validation that the parameter file contains asset-specific parameters. Use this heading as the first line of a parameter file that contains sections.
[Global]	Defines parameters that can be reused across test cases.
[Default].[test case name]	Defines parameters for the named test case only. If a parameter is defined in a default section and in a global section, the value in the default section overrides the global value.

If the parameter file does not contain sections, Data Validation reads all parameters as global.

Precede the parameter name with a dollar sign as follows: \$<parameter>. Define parameter values as follows:

```
$<parameter1>=value1  
$<parameter2>=value2
```

For example, you have the parameters SalesQuota and Region. In the parameter file, you define each parameter in the following format:

```
$SalesQuota=1000  
$Region=NW
```



The parameter value includes all characters after the equals sign (=), including leading or trailing spaces. Parameter values are treated as String values.

## Parameter scope

When you define values for the same parameter in multiple sections in a parameter file, the parameter with the smallest scope takes precedence over parameters with a larger scope.

In this case, Data Validation gives precedence to parameter values in the following order:

1. Values defined in a test case section.
2. Values defined in the #USE\_SECTIONS section.
3. Values defined in a global section.

## Creating a test case

Use the test case wizard to create a test case that validates the data between the source and the target.

To create a test case, you perform the following tasks:

1. Define the test case details.
2. Select the two connections that contain the data to compare. For each connection, select the data source to compare. The data source can be a database table, view, or saved SQL query.
3. Select the columns to compare and select primary keys for the data sources to identify where the records are unique.
4. Data Validation maps the columns in the two data sources according to the keys. If any column remains unmapped, map it manually.
5. Select a comparison method.
6. Optionally, configure other test case properties related to bad record limit, string validation, data sampling, schedule, email notification, and logging.

**Note:** Verify that the runtime environment that runs the test contains sufficient disk space to run the test. The size of the tables or views that the test compares, the number of columns, the size of the data in the columns, and the sampling level determine the disk space that Data Validation requires to run the test.

### Step 1. Define test case details

Define the general details of the test case on the **General** tab of the test case wizard.

1. On the navigation bar, click **New > Test Case**.

The test case wizard appears.

2. On the **General** tab, enter the test case name.

The name can contain ASCII, Chinese, Hebrew, and Japanese characters, digits, spaces, and the following characters:

, \_ -

The name can't contain any of the following characters:

` " ' ! " # \$ % & ( ) \* + ' . / : ; < > = ? @ [ ] \ ^ ~ { } |

3. Use the default project or folder location of the test case or select a custom location. To navigate to a different test case location, click **Browse**. Select a new location and click **Select**.
4. Optionally, enter a description of the test case.
5. From the **Runtime Environment** list, select the runtime environment to run the test in. At least one Secure Agent must be installed on the runtime environment.
6. From the **Report Location** list, select a flat file connection to the location where Data Validation stores test reports.  
**Note:** The code page of the connection must be UTF-8.
7. Optionally, to parameterize the WHERE clause for data sampling, perform the following steps:
  - a. Click **Download** to download a sample parameter file.  
 By default, the sample parameter file is downloaded to the following directory:  
`<Secure Agent installation directory>/apps/DVProcessor/data/userparameters`
  - b. Define the parameter names and values in the file. Prefix the parameter names with the \$ character.
  - c. Save the file in a directory relative to your Secure Agent.  
 Default is `<Secure Agent installation directory>/apps/DVProcessor/data/userparameters`.
  - d. In the **Parameterization** section, enter the parameter file directory and parameter file name. For more information, see [“Parameterization for WHERE clause” on page 8](#).
8. Click **Next**.  
 The **Connections** tab appears.

## Step. 2 Select connections and data sources

On the **Connections** tab of the test case wizard, select the connections and data sources that the test case compares.

1. From the **Connection 1** list, select the connection that contains the source.  
 If you use an Amazon Redshift v2 connection, enter a valid path for Connection 1 in the **Path** field and enter a valid S3 bucket name in the **S3 Bucket Name** field.  
 If you use an Amazon S3 v2 or Microsoft Azure Data Lake Storage Gen2 connection, in the **Path** field, optionally, enter a relative path to the location where the file is stored for Connection 1. If you do not enter a path, Data Validation lists all the files in the folder path specified in the connection.  
 If you use a Google BigQuery V2 connection, enter a schema name to get a list of objects.  
 The details of the connection appear below the selected connection.
2. From the **Connection 1** list, select the data source that you want to compare. The data source can be a database table, view, or saved SQL query.  
 If you use an Amazon S3 v2 connection, select a Parquet or flat file.  
 To use a saved SQL query, under the **Object or Saved SQL Query Details** section, click **Saved SQL Query**. Then, from the **Saved SQL Query** list, select a saved SQL query. The query appears in the **SQL Query** box. Click **Validate** to test whether the query is valid or not. If the query is not valid, fix the query and validate it again. If you change the connection or saved SQL query, you must validate the query again.
3. From the **Connection 2** list, select the connection that contains the target.  
 If you use an Amazon Redshift v2 connection, enter a path for Connection 2 in the **Path** field and enter a valid S3 bucket name in the **S3 Bucket Name** field.

If you use an Amazon S3 v2 or Microsoft Azure Data Lake Storage Gen2 connection, in the **Path** field, optionally, enter a relative path to the location where the file is stored for Connection 2. If you do not enter a path, Data Validation lists all the files in the folder path specified in the connection.

If you use a Google BigQuery V2 connection, enter a schema name to get a list of objects.

The details of the connection appear below the selected connection.

4. From the **Connection 2** list, select the data source that you want to compare. The data source can be a database table, view, or saved SQL query.

If you use an Amazon S3 v2 connection, select a Parquet or flat file.

To use a saved SQL query, under the **Object or Saved SQL Query Details** section, click **Saved SQL Query**. Then, from the **Saved SQL Query** list, select a saved SQL query. The query appears in the **SQL Query** box. Click **Validate** to test whether the query is valid or not. If the query is not valid, fix the query and validate it again. If you change the connection or saved SQL query, you must validate the query again.

5. If the connections that contain the data to compare point to flat files, specify the delimited flat file formatting options.

The following table describes the flat file formatting parameters:

Parameters	Description
Delimiter	Indicates the boundary between two columns of data. Select one of the following options: <ul style="list-style-type: none"><li>- Comma</li><li>- Tab</li><li>- Colon</li><li>- Semicolon</li><li>- Other. Select this option and specify the character to use as a delimiter.</li></ul> If you choose an escape character or a quote character as a delimiter, or if you use the same character as consecutive delimiter and qualifier, you might receive unexpected results. Default is comma.
Text Qualifier	Character that defines the boundaries of text strings. If you select a quote character, Data Validation ignores delimiters within quotes. Default is double quote (").
Escape Character	Character that immediately precedes a column delimiter character embedded in an unquoted string, or immediately precedes the quote character in a quoted string. When you specify an escape character, Data Validation reads the delimiter character as a regular character.
First Data Row	Number of the row that Data Validation starts reading the file from during import. For example, if you enter <b>2</b> , Data Validation skips the first row. <b>Note:</b> If you set <b>Field Labels</b> to <b>Import from Row</b> , Data Validation sets the first data row automatically. For example, if you set the field labels row number to 10, Data Validation sets the first data row to 11.

Parameters	Description
Field Labels	Determines how Data Validation displays column names in profile results. Select one of the following options: <ul style="list-style-type: none"> <li>- Auto-generate. Data Validation generates the column names. This option doesn't apply to flat files within an Amazon S3 v2 connection.</li> <li>- Import from Row. Data Validation imports the column name from the specified row number.</li> </ul>
Row Number	Row that contains the column name. Applies when you choose <b>Import From Row</b> in the <b>Field Labels</b> option.

- Click **Next**.

The **Configuration** tab appears.

## Step 3. Select table columns and primary keys

Select the columns to compare and the primary keys of the source and target on the **Configuration** tab of the test case wizard.

Based on the database requirements, you can select a single primary key, or select multiple keys to create a composite key.

You must select at least one primary key for each connection. The primary key must be a unique key in the database that can identify a row uniquely. If the primary key column has duplicates, the report might show inaccurate results.

The length of a composite key must not exceed 1,000 characters.

- From the **Available Fields** list in the Connection 1 area, select the fields to compare and use the right arrow to move them to the **Selected Fields** list, or use the double right arrow to move all available fields.
- From the **Primary key for the table in Connection 1** list, select primary keys for the source.
- Optionally, select **Data preview**.

A preview of the Connection 1 table columns you selected appears. Optionally, modify the selection as needed.

- From the **Available Fields** list in the Connection 2 area, select the fields to compare and use the right arrow to move them to the **Selected Fields** list, or use the double right arrow to move all available fields.
- From the **Primary key for the table in Connection 2** list, select primary keys for the target.
- Optionally, select **Data preview**.

A preview of the Connection 2 table columns you selected appears. Optionally, modify the selection as needed.

- Click **Next**.

The **Mapping** tab appears.

## Step 4. Map unmapped columns

Data Validation maps the columns in the two tables or views based on the column names. The **Mapping** tab of the test case wizard shows the mapping status of each column including the data type, precision, and scale.

The **Mapping** tab shows if the data type matches in each pair of columns. If any columns remain unmapped, map them manually. You must map all unmapped columns.

If Data Validation mapped all the columns successfully, go on to the next step.

You can also choose a comparison method. For more information, see [“Comparison methods” on page 7](#).

1. Select one of the following comparison methods:
  - Value Test - Compare Entire Table. Data Validation compares the entire table based on the actual data values. This is the default method.
  - Aggregation Functions Test. Select the aggregation functions that Data Validation must use to compare the table.
2. In the **Map Columns** table, if the value of **Column Mapped?** for a column is **No**, from the **Table Column in Connection 2** list, select a column to map to the Connection 1 column.
3. Repeat Step 1 to map all unmapped columns.
4. Optionally, select **Data preview**.

A preview of the Connection 1 table columns and Connection 2 table columns you selected appears. Optionally, modify the selection as needed.
5. Click **Next**.

The **Settings** tab appears. The **Save** and **Run** buttons are active.
6. Click **Save** to save the test case, or perform the next optional steps.

## Step 5. Configure test case properties (optional)

Configure optional properties on the **Settings** tab of the test case wizard.

1. In the **Bad Record Limit** section, select the maximum number of unmatched, extra, and missing records to show in the detailed test results.

You can select one of the following values:

- 100
- 500
- 1000

2. In the **String Validation** section, perform the following steps:
  - Select the **Ignore case** option to ignore casing differences in the data.
  - Select the **Trim white spaces in a string value** option to trim leading and trailing white spaces in string values.
3. Configure data sampling for Data Validation to compare data between the source and the target. The less data that Data Validation samples, the faster the test runs.

This setting applies to all the columns you select to compare, in both tables.

In the **Data Sampling** section, select one of the following values:

- WHERE clause. Data Validation selects those rows for sampling that meet the condition specified in the WHERE clause. To create a WHERE clause, click **Add Clause**. Enter the WHERE clause for the connection. Optionally, click **Validate**. If there are no validation errors, click **Continue**. If there are validation errors, fix the clause and try again.

You can also parameterize the WHERE clause to provide different values at run time. Prefix the parameter name with the \$ character. For more information, see [“Parameterization for WHERE clause” on page 8](#).
- Sample the first. Select the number of rows at the beginning of the table that the test compares.
- Sample the last. Select the number of rows at the end of the table that the test compares.

- **Percentage.** Select a percentage of the rows that the test compares. The test report shows records that are missing or unmatched in the Connection 2 table, but doesn't show extra records if any exist.

For information about the rules and guidelines that you must consider while configuring data sampling, see [“Rules and guidelines for data sampling” on page 14](#).

4. In the **Schedule Test Case Run** section, select a schedule to run the test case.  
You can create schedules in Administrator.
5. In the **Email Notification Options** section, specify whether you want to send email notifications for the test case run.  
Select one of the following values:
  - Do not set email notifications. Data Validation doesn't send email notifications.
  - Use the default email notification options for my organization. Data Validation uses the default email notification options that are configured in Administrator.
  - Use custom email notifications. Specify the email addresses that Data Validation must use for the success and failure email notifications. Separate multiple email addresses with a comma.
6. In the **Logging** section, select one of the following logging levels:
  - **Standard.** The log includes error messages.
  - **Verbose.** The log includes messages of all logging levels. This option is useful for debugging.
7. To save the Data Integration mappings and tasks that Data Validation creates when it runs the test case, select **Save Data Integration assets after the test case run**.
8. Click **Save**.

The test case is ready to be run.

## Rules and guidelines for data sampling

Consider the following rules and guidelines when you configure data sampling:

- If you use an aggregation functions test, you can select only the WHERE clause option for data sampling. The other sampling options are not available.
- If you select the **Sample the first** or **Sample the last** option, in addition to the mismatched records in the sampled data, Data Validation reports the missing rows and extra rows in the sampled data set.
- Data Validation uses string-based sorting for the data. Therefore, if the primary key column contains numeric values and you select the **Sample the first** or **Sample the last** option, you might see different results in the report. For example, the following table shows the values before and after sorting:

Before Sorting	After Sorting
1	1
5	10
10	12
34	34
12	5

# Viewing test case details

You can view details about a test case, including test, connection, and source details.

1. On the **Explore** page, navigate to the test case.
2. In the row that contains the test case, click **Actions** and select **View**.

On the **Test Case Details** page, you can click **Edit** to modify the test case or **Run** to run the test case.

# Running a test case

You can run a test case from the **Explore** page, from the test case details page, or from the test case wizard.

- To run a test case from the **Explore** page, navigate to the test case. In the row that contains the test case, click **Actions** and select **Run**.
- To run a test case from the test case details page, on the **Explore** page, navigate to the test case. In the row that contains the test case, click **Actions** and select **View**, and then, on the test case details page, click **Run**.
- To run a test case from the test case wizard, after you save the test case, click **Run**.

After you run a test case, review the reports to assess whether the data you compared matches between test case connections. For more information about reports, see *Monitor*.

# Deleting a test case job

You can delete a test case job on the **Test Case Details** page. You can't delete a job instance of a test case that you add to a test suite.

1. On the **Explore** page, navigate to the test case.
2. In the row that contains the job of the test case to delete, click **Actions** and select **View**.

The **Test Case Details** page opens.

3. Click the **Jobs** tab and navigate to the job to delete.
4. In the row that contains the job, click the Delete icon.

Data Validation deletes the job and any report or log associated with the job.

# INDEX

## C

Cloud Application Integration community  
URL [4](#)  
Cloud Developer community  
URL [4](#)

## D

Data Integration community  
URL [4](#)

## I

Informatica Global Customer Support  
contact information [5](#)  
Informatica Intelligent Cloud Services  
web site [4](#)

## M

maintenance outages [5](#)

## S

status  
Informatica Intelligent Cloud Services [5](#)  
system status [5](#)

## T

trust site  
description [5](#)

## U

upgrade notifications [5](#)

## W

web site [4](#)