# Identify Columns and Mappings

Identify the data that needs to be replicated to Snowflake. Identify the following

* Source table and column names
* Source column data types
* Destination table and column names
* Destination data types
* Are transformations needed. This should be accounted for in the table created in Snowflake.
  + NULLS present? How should they be handled?
  + Calculated columns (for example time stamp)
  + Concatenated columns
  + Trimmed columns

This example will use the following:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Source Database** | SysproCompany100 | SysproCompany100 | SysproCompany100 | SysproCompany100 | SysproCompany100 |
| **Source Schema** | dbo | dbo | dbo | dbo | dbo |
| **Source Table** | LctRoute | LctRoute | LctRoute | LctRoute | LctRoute |
| **Source Column** | Route | Description | DaysBeforeEta | DemurrageDays | DaysAfterEta |
| **Source Data Type** | VARCHAR(8) | VARCHAR(50) | DECIMAL(5,0) | DECIMAL(5,0) | DECIMAL(5,0) |
| **Destination Database** | SQLREPORTING | SQLREPORTING | SQLREPORTING | SQLREPORTING | SQLREPORTING |
| **Destination Schema** | SYSPROCOMPANY100 | SYSPROCOMPANY100 | SYSPROCOMPANY100 | SYSPROCOMPANY100 | SYSPROCOMPANY100 |
| **Destination Table** | ROUTE | ROUTE | ROUTE | ROUTE | ROUTE |
| **Destination Column** | Route | Description | DaysBeforeEta | DemurrageDays | DaysAfterEta |
| **Destination Data Type** | VARCHAR(8) | VARCHAR(50) | DECIMAL(5,0) | DECIMAL(5,0) | DECIMAL(5,0) |

The only thing that is different is the source and destination table name.

## Create Table Definition for Snowflake

CREATE TABLE ROUTE

(

Route VARCHAR(8) NULL,

Description VARCHAR(50) NULL,

DaysBeforeEta DECIMAL(5,0) NULL,

DemurrageDays DECIMAL(5,0) NULL,

DaysAfterEta DECIMAL(5,0) NULL

)

## Create Query for Data Retrieval

SELECT Route,

Description,

DaysBeforeEta,

DemurrageDays,

DaysAfterEta

FROM LctRoute;

Note: When the query is entered into Talend it must be surrounded by double quotes like

“

SELECT Route,

Description,

DaysBeforeEta,

DemurrageDays,

DaysAfterEta

FROM LctRoute;

”

# Create Table in Snowflake

## Login to Snowflake

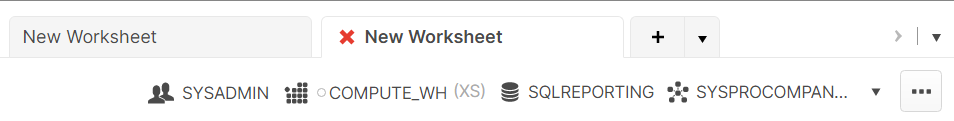
Use the following to log into Snowflake:

Snowflake URL: <https://vs10472.central-us.azure.snowflakecomputing.com>

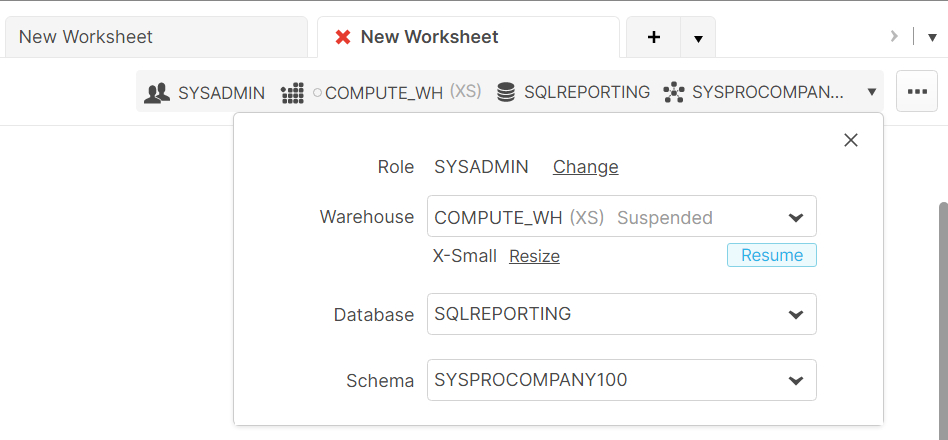
User Name: SF\_Talend

Password: V!6!i932t#FE@KZaA2fq

1. Paste table create code into the query window.
2. Verify you are in the correct database and schema context. In the top right of the screen you can verify the Role (SYSADMIN), Warehouse (COMPUTE\_WH), Database (SQLREPORTING) and Schema (SYSPROCOMPANY100).

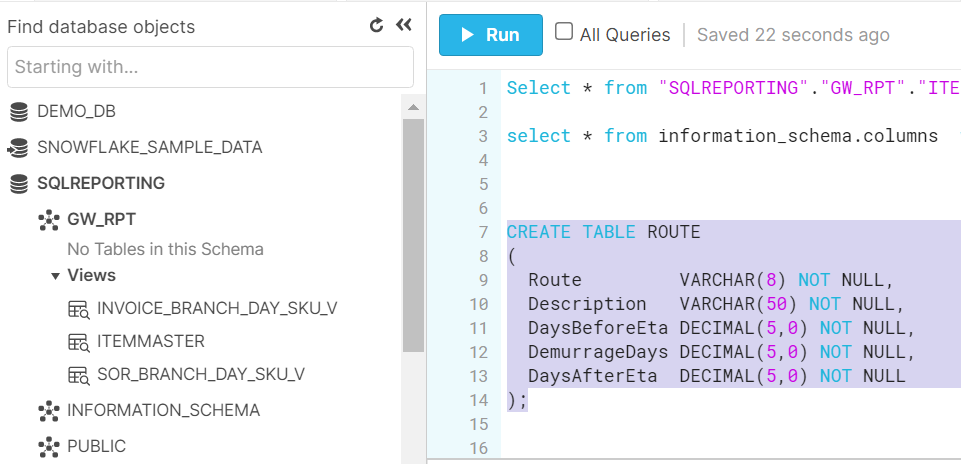


If you need to make changes click the down arrow next to the ellipse.



These settings are correct for this example.

1. If other code is in the query window highlight just the code you want to run.
2. Click the Run button to create the table.



Note: The screen shot shows the columns configured to not accept NULLS. However, all columns in SnowFlake should be configured to accept NULLs.

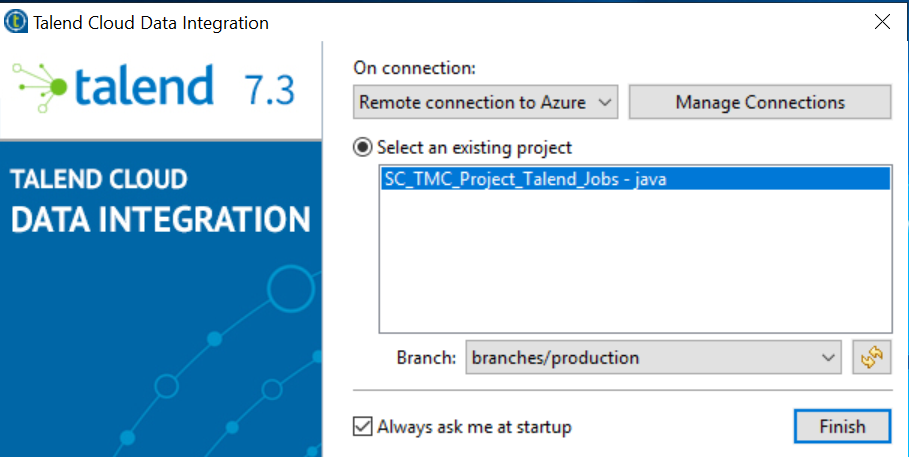
# Create Talend Job

## Login to Talend (RDP)

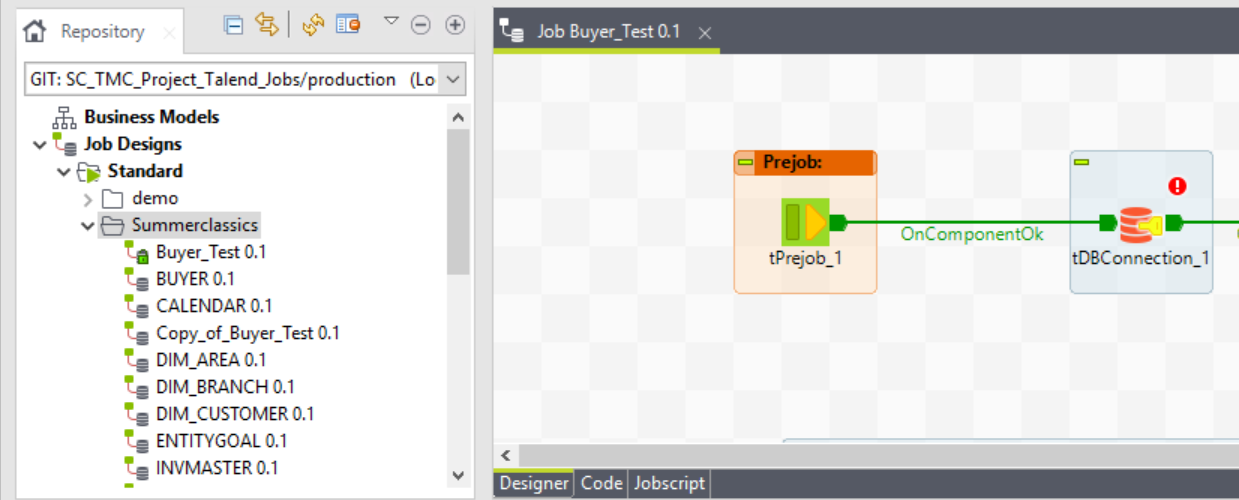
Login to server Talend using RDP and your adm\_ Active Directory account.

## Run the Talend application

Run the Talend Cloud Data Integration application located at C:\Program Files (x86)\Talend-Studio\studio\Talend-Studio-win-x86\_64.exe as Administrator. Select branches/production for Branch.

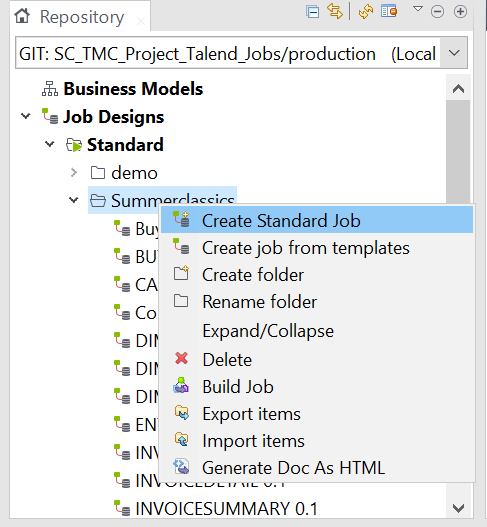


Close the welcome screen.

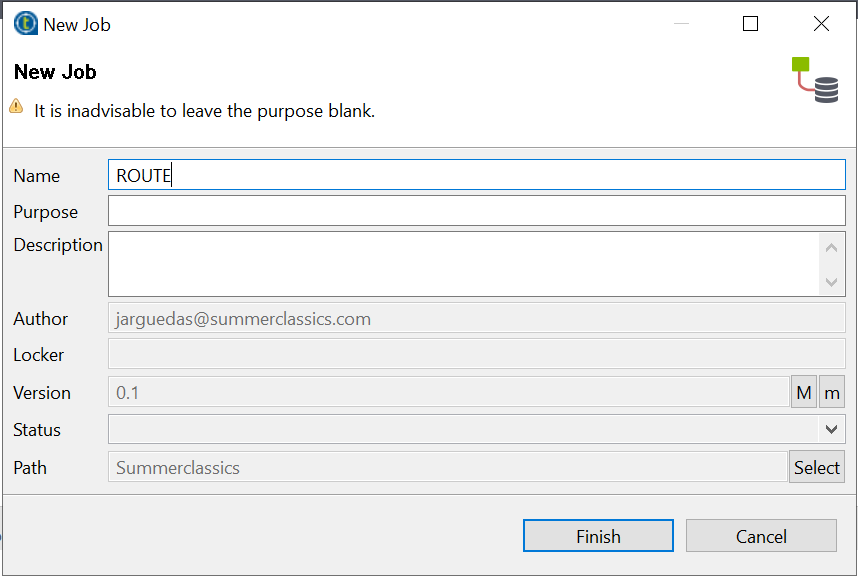
Expand Job Designs -> Standard -> Summerclassics in the left pane. 

# Create a new job

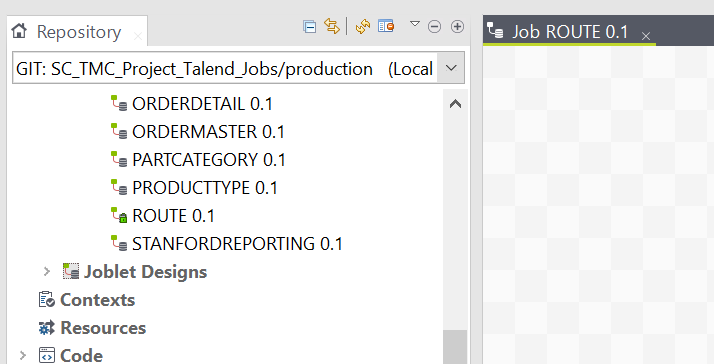
Right-click Summerclassics and select Create Standard Job.



Enter the job name and click Finish.



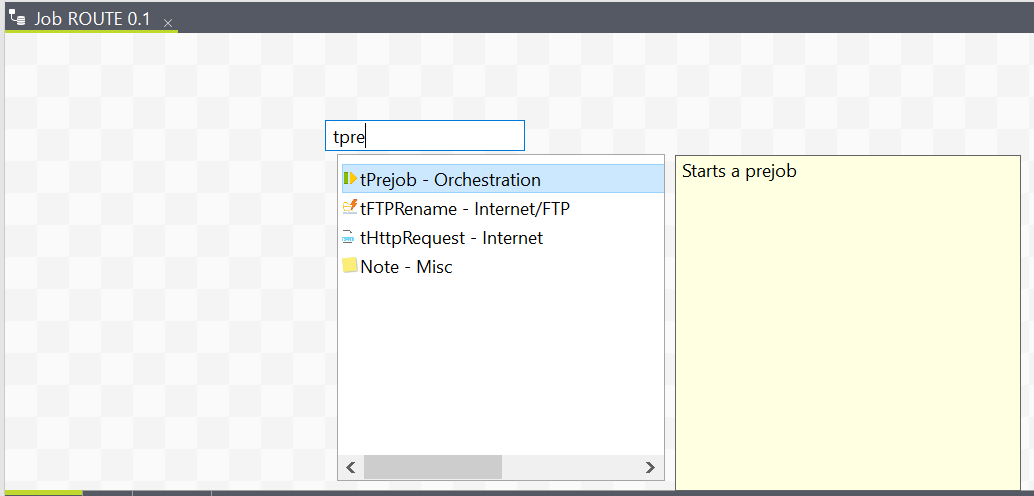
The version is appended to the job name. The job ROUTE 0.1 has been created and is opened for editing.



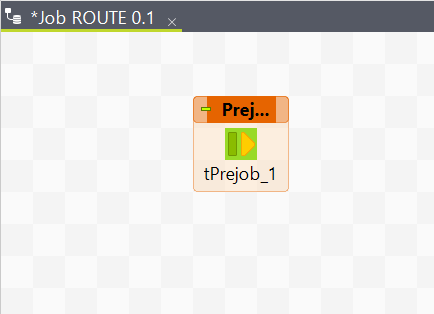
# Create Connections

## Create Pre-Job Connections

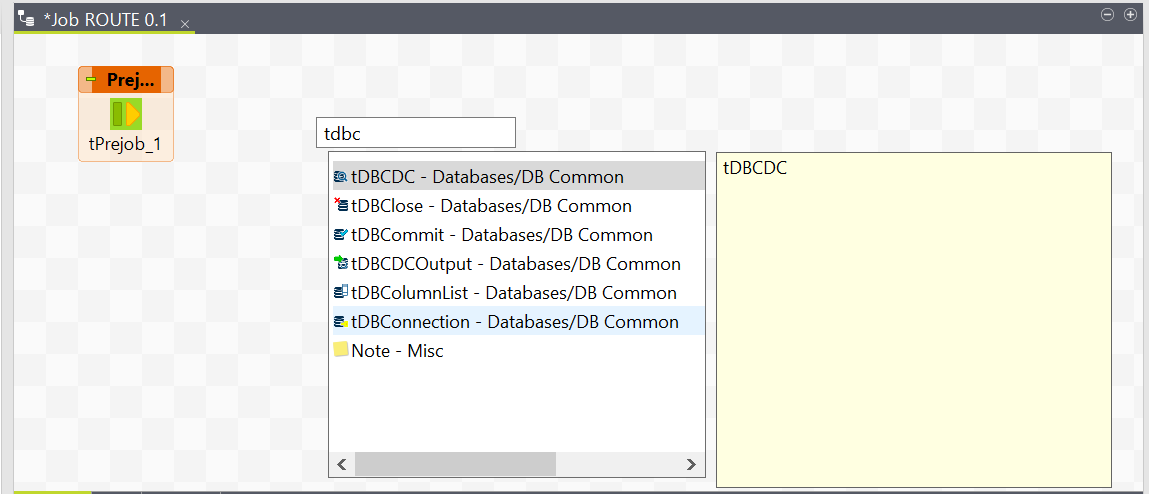
Select the canvas and begin typing tPrejob – Orchestration. When tPrejob – Orchestration appears select it.



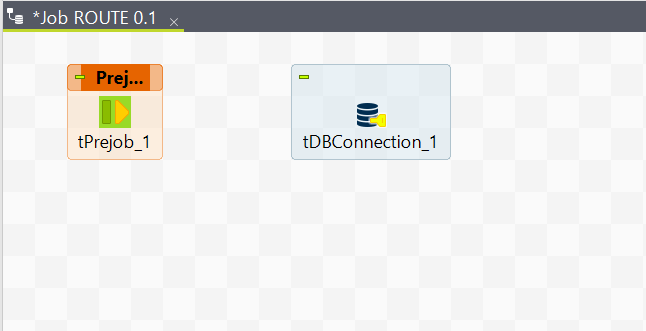
A tPrejob – Orchestration component named tPrejob\_1 is added to the canvas.



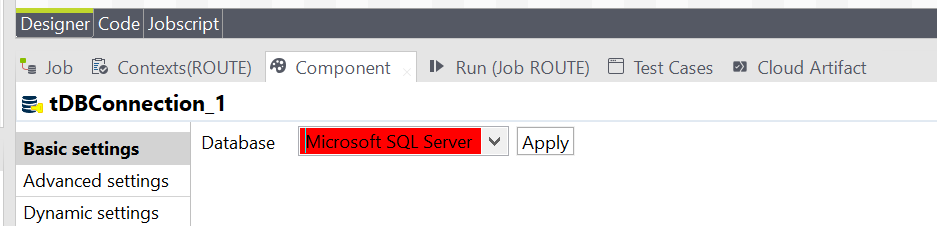
Create a database connection for the prejob. Select the canvas and begin typing tDBConnection – Databases/DB Common. When tDBConnection – Databases/DB Common appears select it.



A new tDBConnection – Databases/DB Common component named tDBConnection\_1 is added to the canvas.



Double click the tDBConnection\_1 component so that its properties can be edited in the bottom window. Select Microsoft SQL Server in the Database drop-down box and click Apply.



Enter the following for the tDBConnection\_1 properties:

JDBC Provider: Open source JTDS

Host: "sqlreporting.summerclassics.msft"

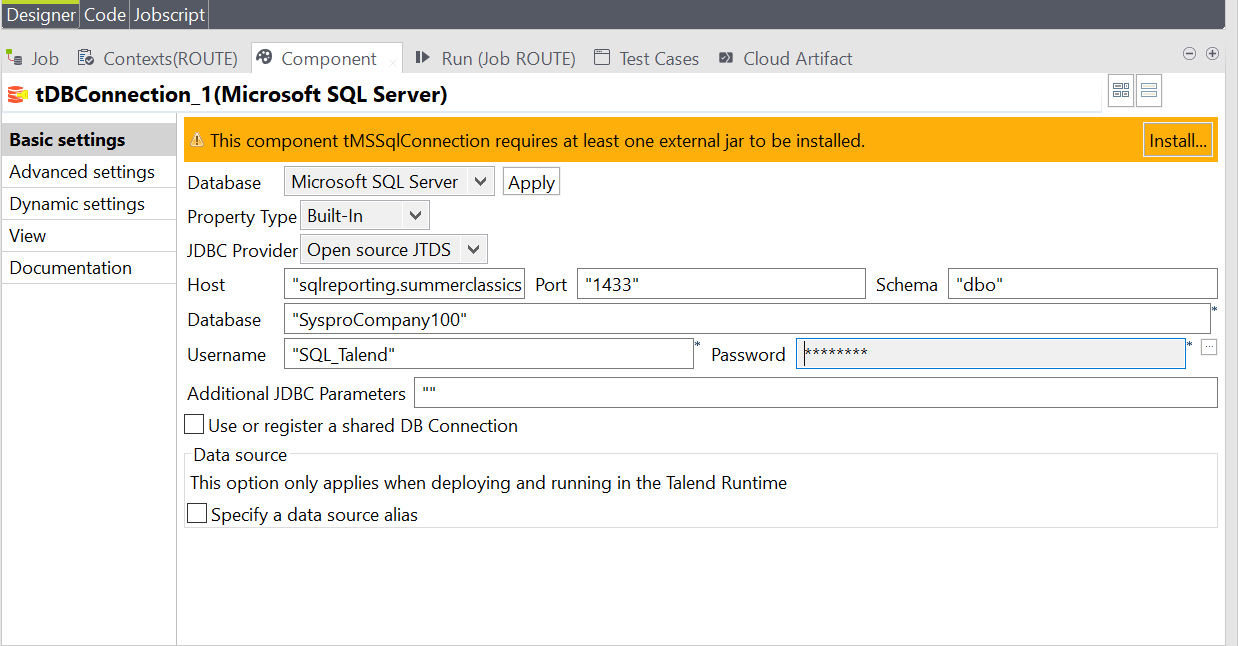
Port: "1443"

Schema "dbo" (from the table mapping)

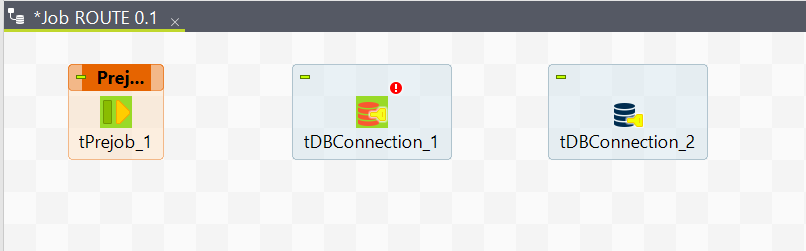
Database: "SysproCompany100" (from the table mapping)

Username: "SQL\_Talend"

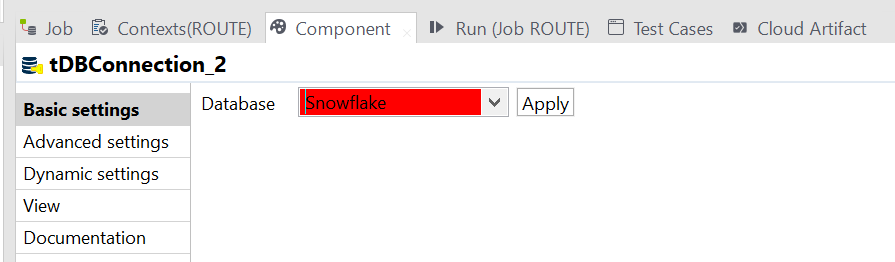
Password: "PYT8$De!%%66BbQftm9D" (to enter the password click the ellipse to the right of the password field)



Create a second tDBConnection – Databases/DB Common component using the same steps used to create the first one. It will be added to the canvas with the name tDBConnection\_2.



Double click it to edit its properties. Select Snowflake in the Database drop-down box and click Apply.



Configure it with the following:

Account: "vs10472" (account is taken from Snowflake URL)

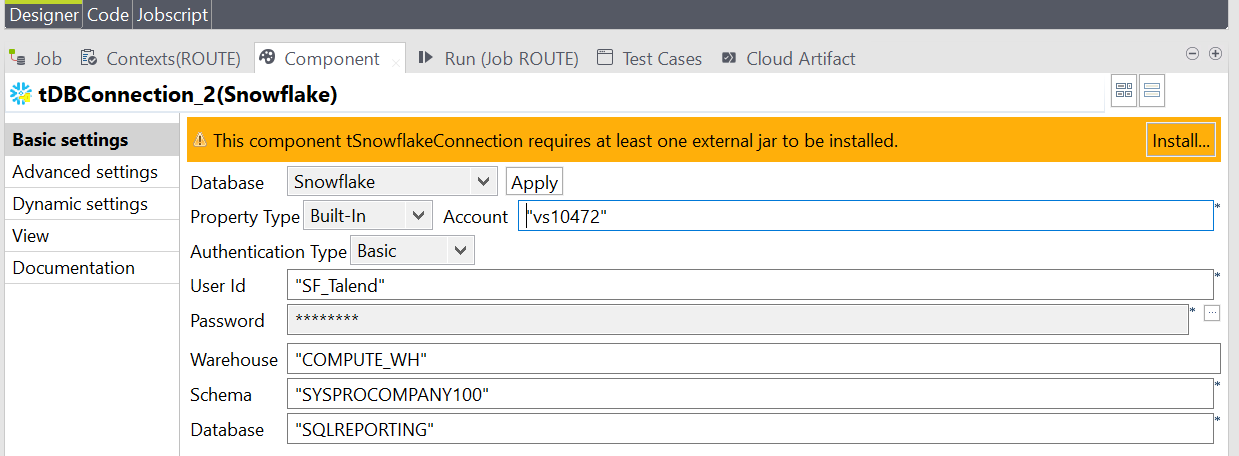
User id: "SF\_Talend"

Password: "V!6!i932t#FE@KZaA2fq" (ellipse to the right of Password field)

Warehouse: "COMPUTE\_WH"

Schema: "SYSPROCOMPANY100"

Database: "SQLREPORTING"



Advanced Settings:

Additional JDBC Parameters: "vs10472.central-us.azure.snowflakecomputing.com"

Login Timeout: 15

Role: "SYSADMIN"

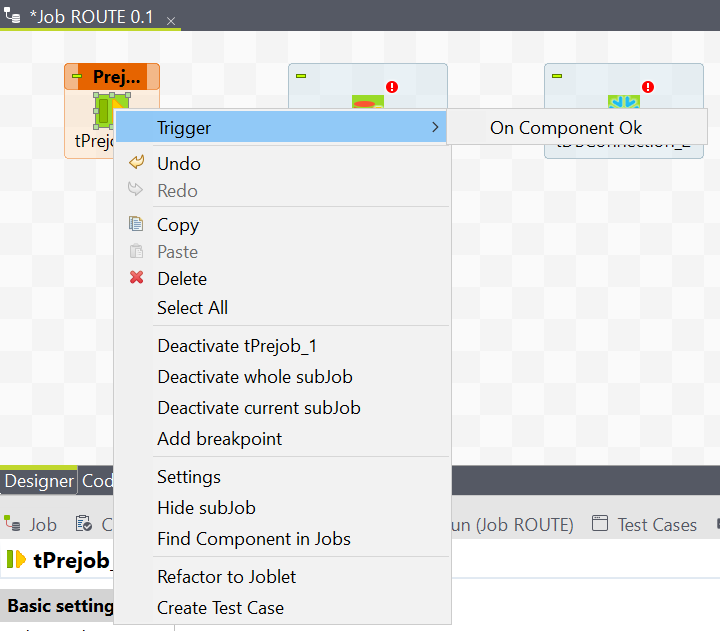
Region ID (Deprecated): "central-us.azure"

Graphical user interface, text, application, email

Description automatically generated

There is no save button. Whatever you enter is saved automatically.

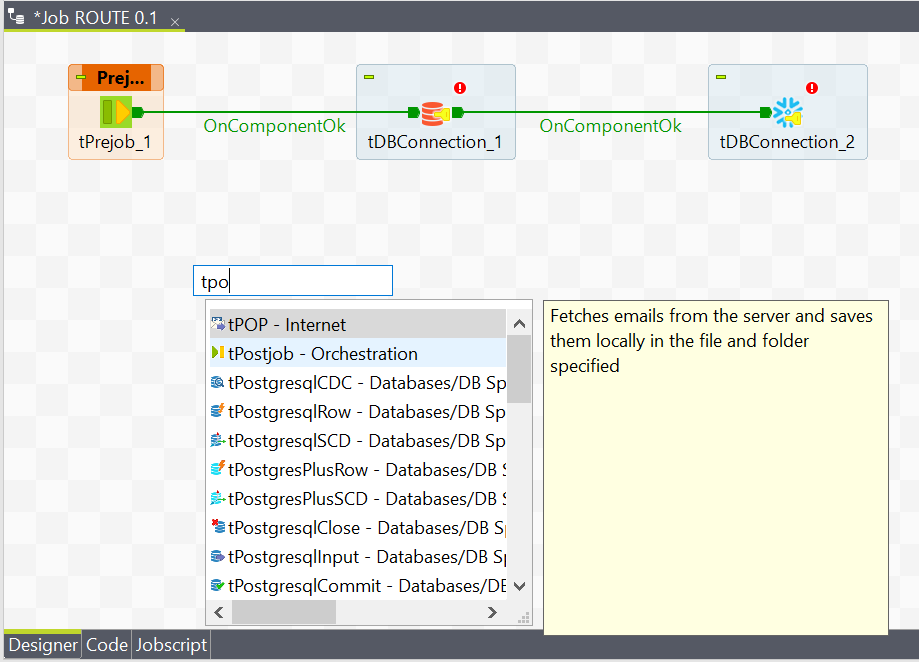
Connect the tPrejob\_1 component to the tDBConnection\_1 component. Right-click the tPrejob\_1 component and select Trigger -> On Component Ok. Connect the free end of the connector to the tDBConnection\_1 component.



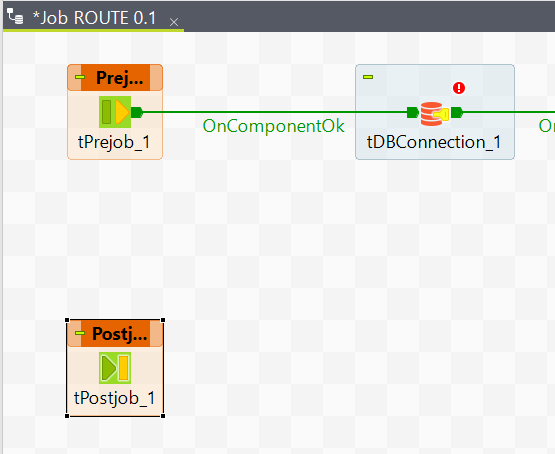
Use the same process to connect the tDBConnection\_1 component to the tDBConnection\_2 component.

# Create Post-Job Connections

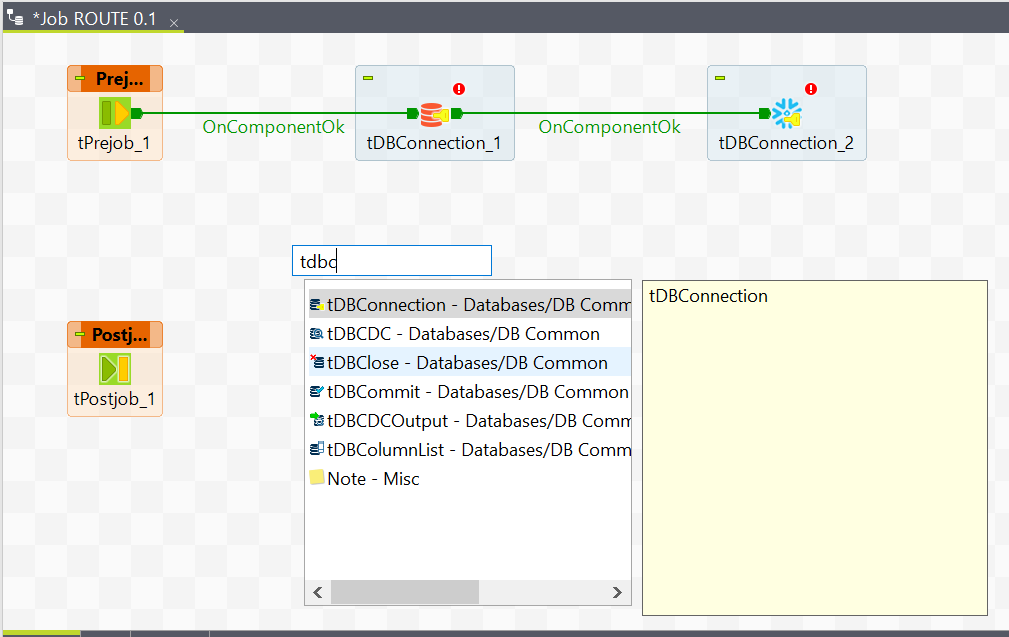
Select the canvas and begin typing tPostjob – Orchestration. When tPostjob – Orchestration appears select it.



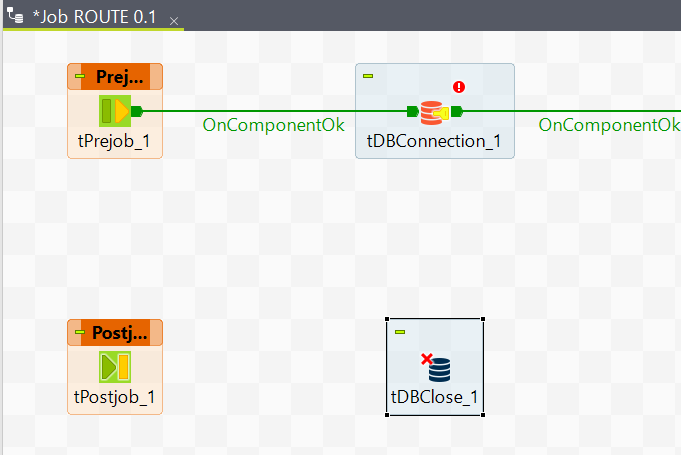
A tPostjob – Orchestration component named tPostjob\_1 is added to the canvas.



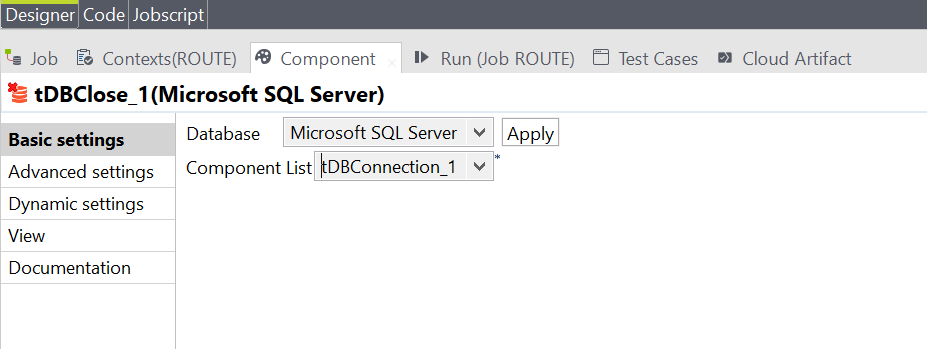
Create a database close component for the postjob. Select the canvas and begin typing tDBClose – Databases/DB Common. When tDBClose – Databases/DB Common appears select it.



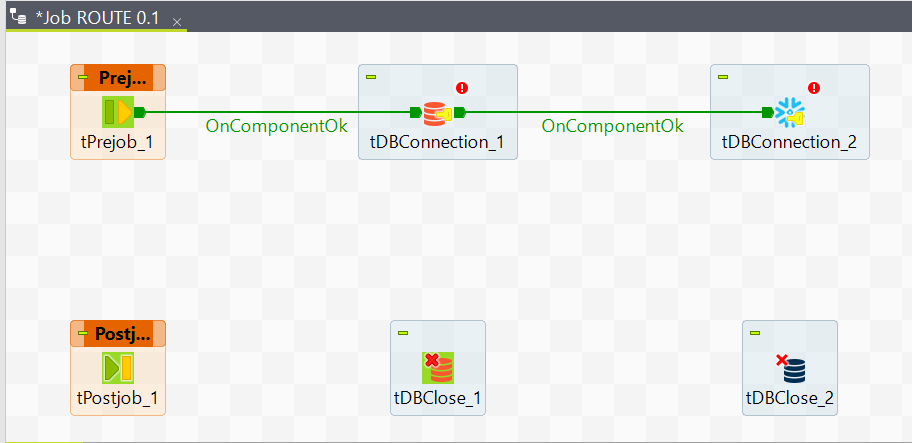
A new tDBClose – Databases/DB Common component named tDBClose\_1 is added to the canvas.



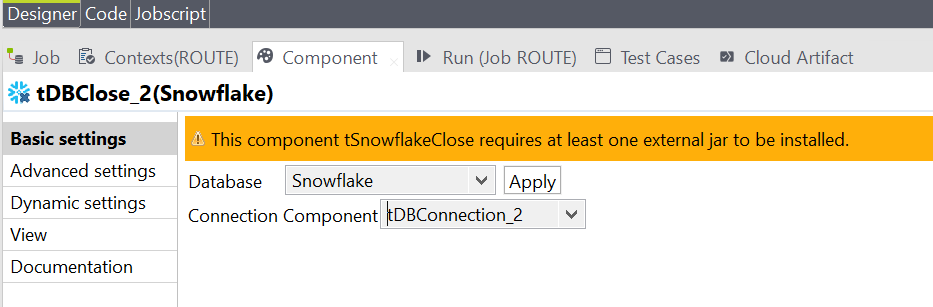
Double-click tDBClose\_1 to edit its properties. tDBConnection\_1 opens a connection to the Microsoft SQL Server database and tDBClose\_1 will close the connection to the Microsoft SQL Server database. Select Microsoft SQL Server from the Database drop-down box and click Apply. Select tDBConnection\_1 from the Component List drop-down box.



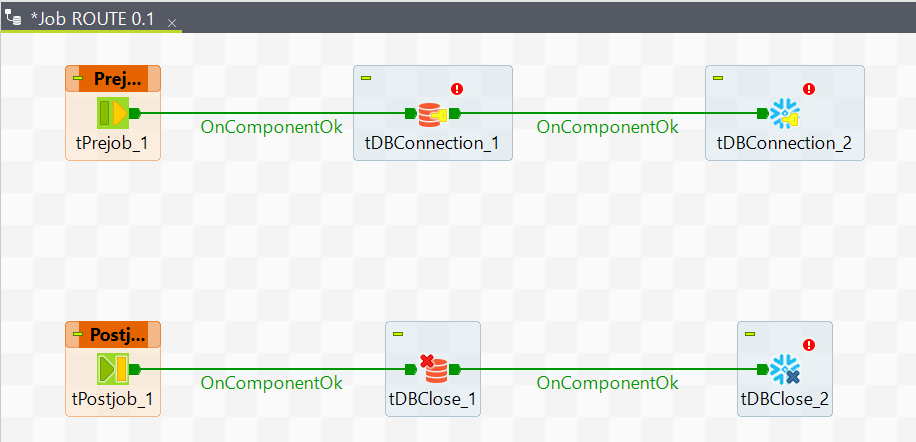
Create a second tDBClose – Databases/DB Common component using the same steps used to create the first one. It will be added to the canvas with the name tDBClose\_2. This component will close the connection to Snowflake, tDBConnection\_2.



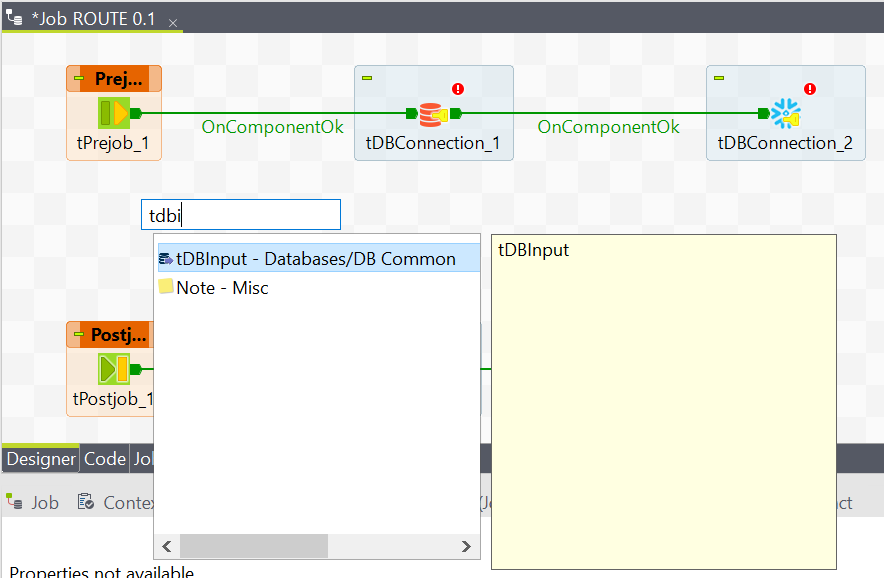
Double-click it to edit its properties. Select Snowflake from the Database drop-down box and click Apply. Select tDBConnection\_2 from the Component List drop-down box.



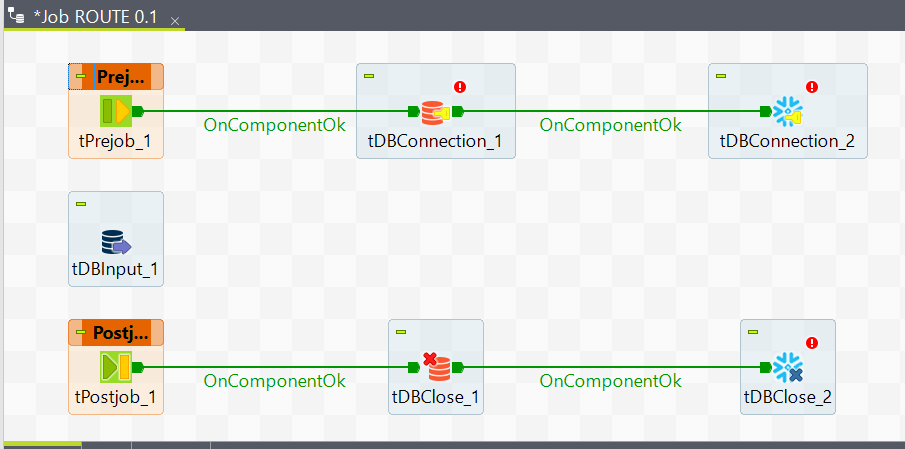
Connect tPostjob\_1 to tDBClose\_1 and tDBClose\_1 to tDBClose\_2 using OnComponentOk triggers just as you connected the pre-job components



# Create Workflow Components

Create a database input component. Begin typing tDBInput – Databases/DB Common. When tDBInput – Databases/DB Common appears select it.

A new tDBInput – Databases/DB Common component named tDBInput\_1 will be added to the canvas.



Double-click the tDBInput\_1 component to edit its properties. Select Microsoft SQL Server from the Database drop-down box and click Apply. Select the Use an existing connection check box. Select tDBConnection\_1 in the Component List drop-down box. Paste the data retrieval query created previously into the Query field. Double-click tDBInput\_1 to view its properties. Click Guess Schema.

Graphical user interface, text, application, email

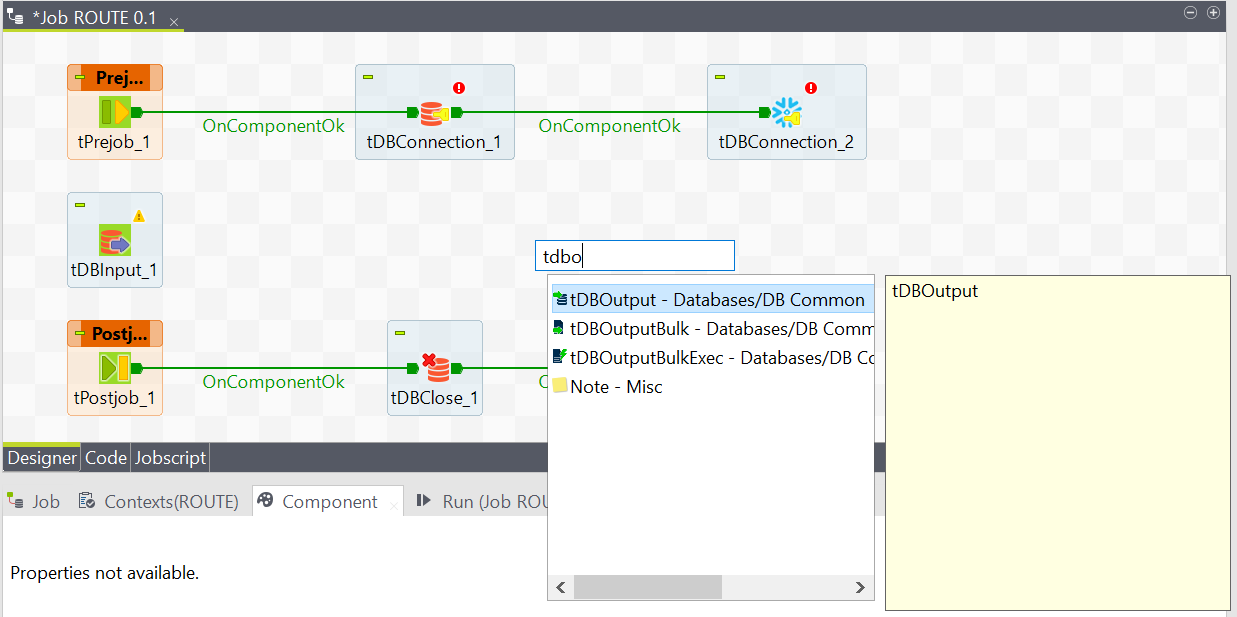
Description automatically generated

Graphical user interface, table

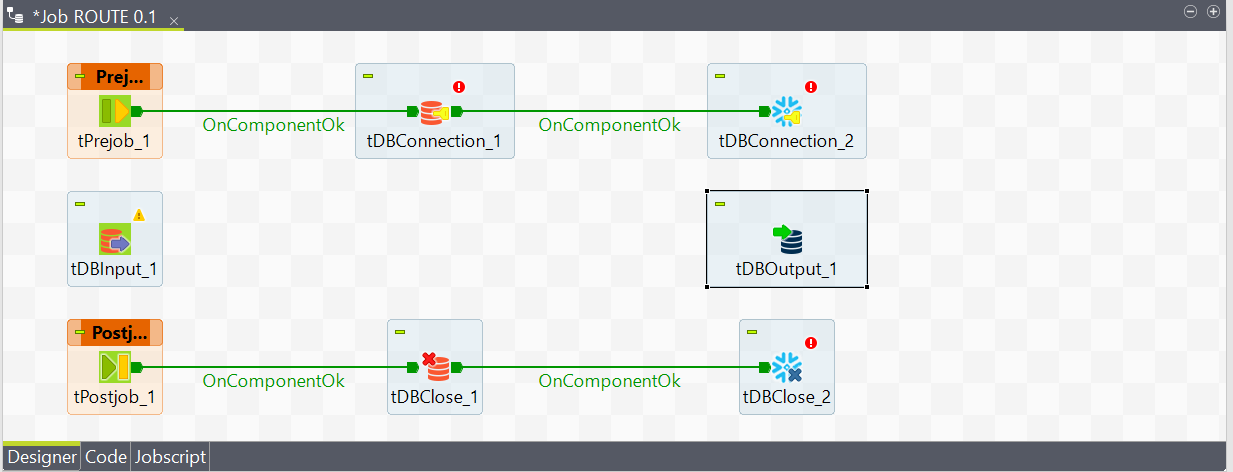
Description automatically generated

Talend will try to figure out the source schema. Make any corrections that need to be made and then click OK.

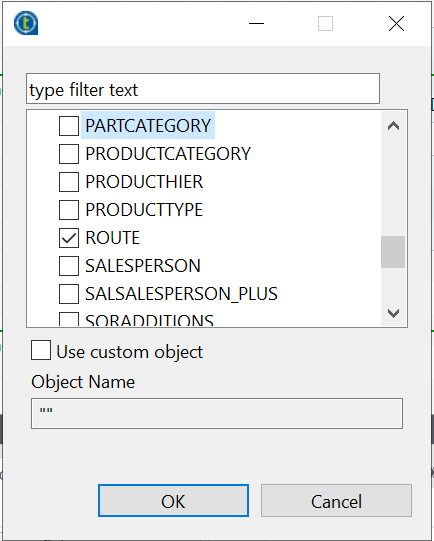
Create a database output component. Begin typing tDBOutput – Databases/DB Common. When tDBOutput – Databases/DB Common appears select it.



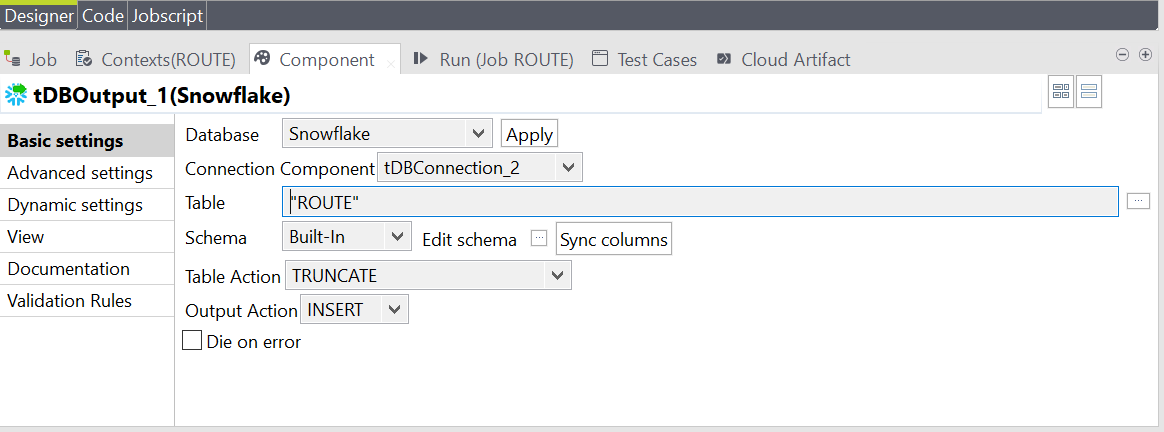
A new tDBInput – Databases/DB Common component named tDBOutput\_1 will be added to the canvas.



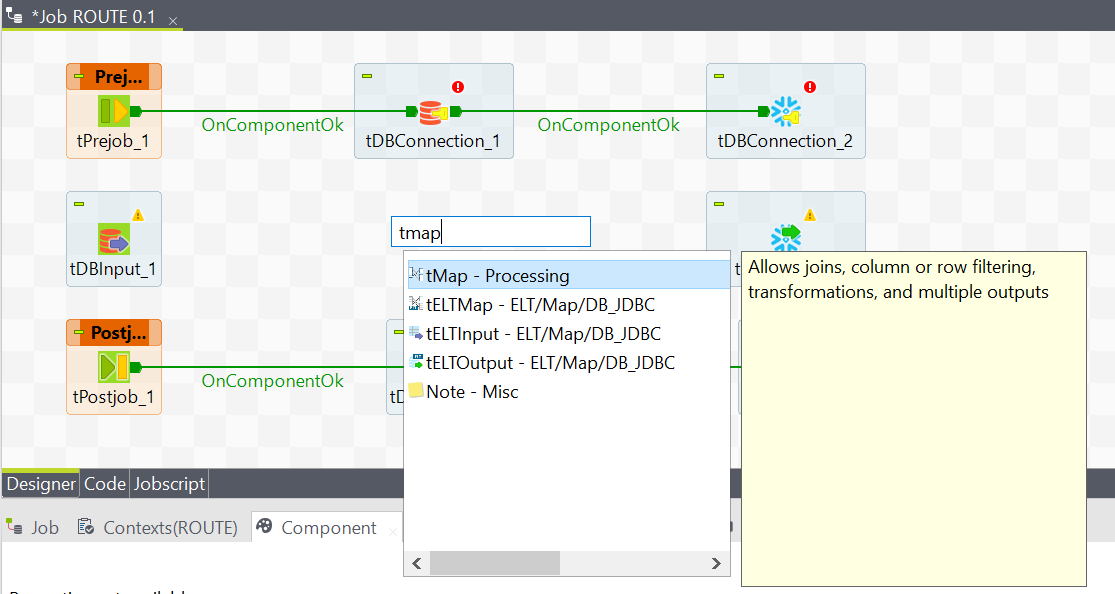
Double-click the tDBOutput\_1 component to edit its properties. Select Snowflake from the Database drop-down box and click Apply. Select the Use an existing connection check box. Select tDBConnection\_2 in the Component List drop-down box. Select Table Action = TRUNCATE. Select the ellipse to the right of the Table field on the Basic Settings tab. Talend will return a list of all table in the warehouse, database and schema specified for the connection. Select the table created earlier in Snowflake and click OK.



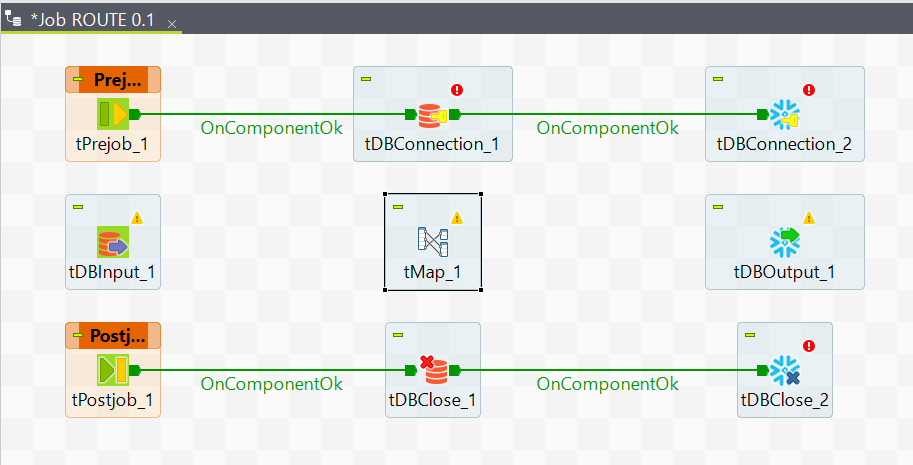
The Table field should now be populated.



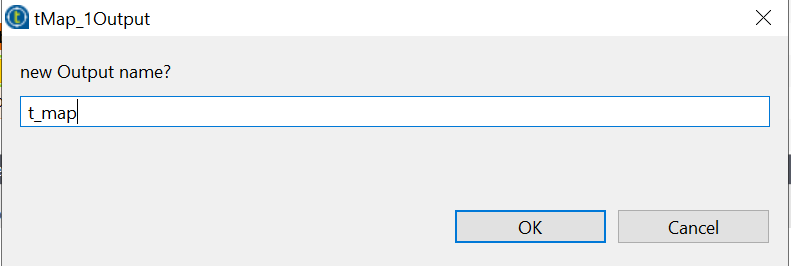
Create a map component. Begin typing tMap – Processing. When tMap – Processing appears select it.



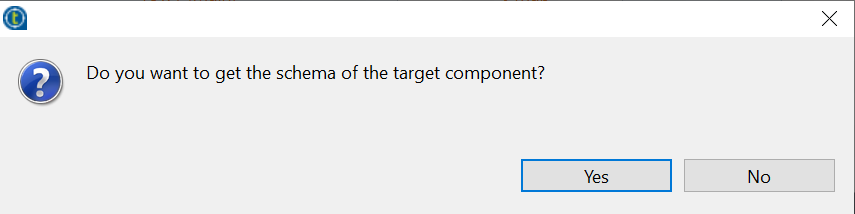
A new map component named tMap\_1 appears on the canvas.



Select tDBInput\_1. Left-click on the small red whatever it is called on the right side of tDBInput\_1. While holding down the mouse button drag a connector from tDBInput\_1 to tMap\_1. Follow the same process to create a connection between tMap\_1 and tDBOutput\_1. When creating the connection between tMap\_1 and tDBOutput\_1 the following window will appear. Enter a name for the output. Here we have used t\_map. Click OK

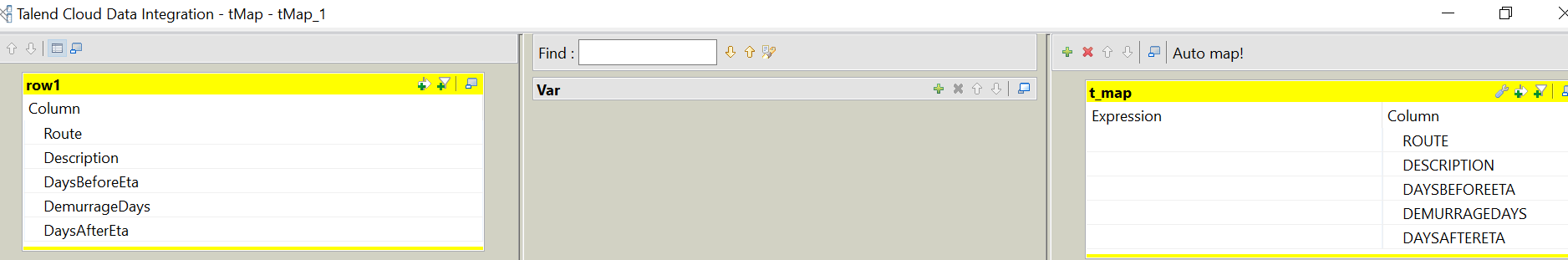


Select Yes to get the schema of the target component.

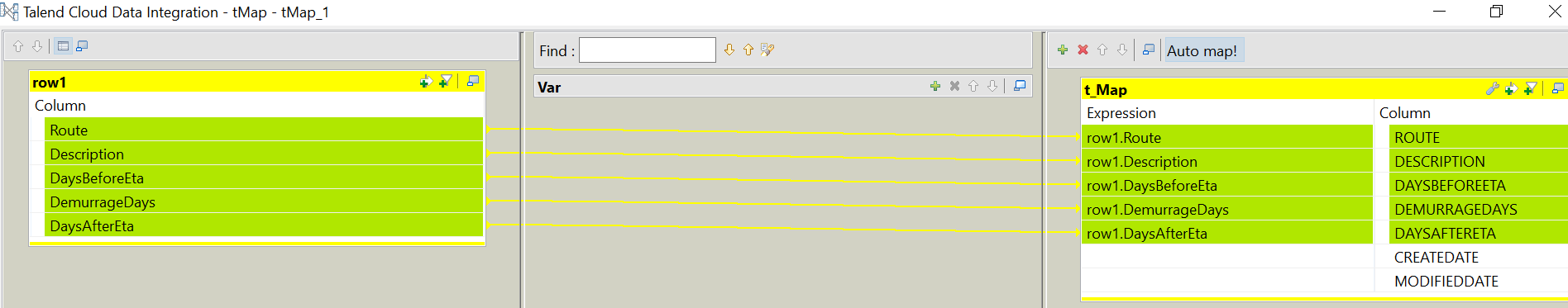


# Create the Mappings

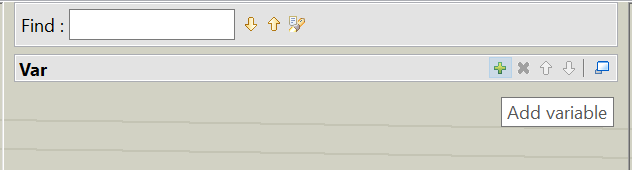
Double-click tMap\_1 to edit its properties. It should show the schema for the source and destination.



If no transformations need to be made click the Auto map! button. Talend matches up the columns and creates the mappings.

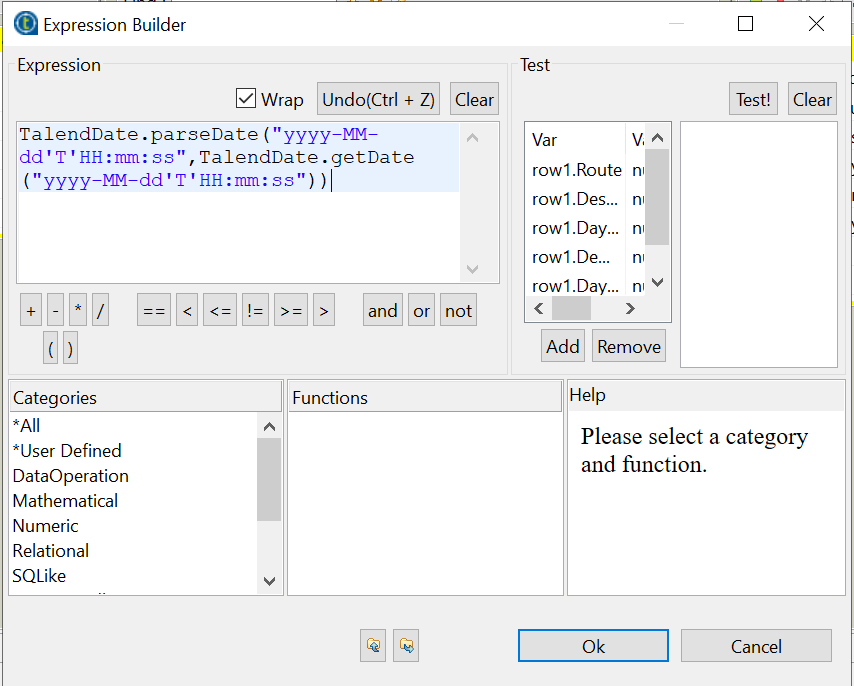


Create time stamp columns if needed. Double-click tMap\_1 to edit its properties. Click the green Add Variable button.

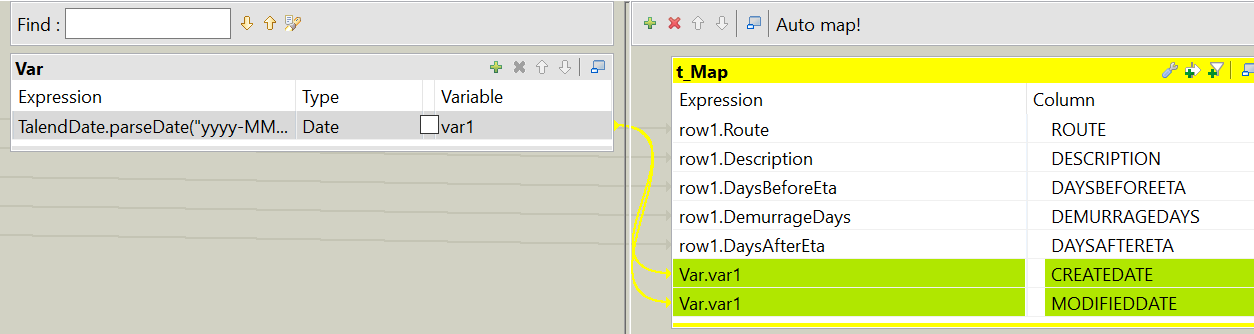


Click the ellipse next to the Expression field and enter the following in the Expression field and click OK

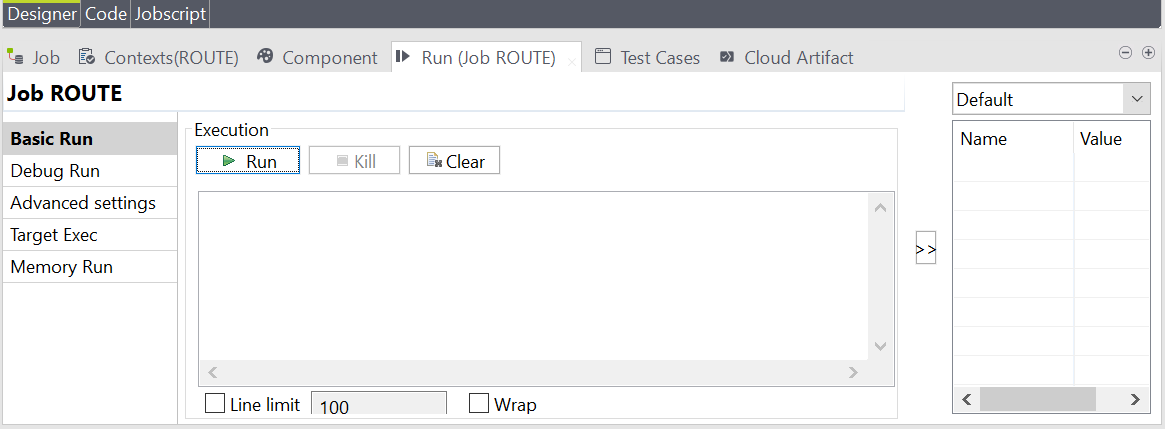
TalendDate.parseDate("yyyy-MM-dd'T'HH:mm:ss",TalendDate.getDate("yyyy-MM-dd'T'HH:mm:ss"))

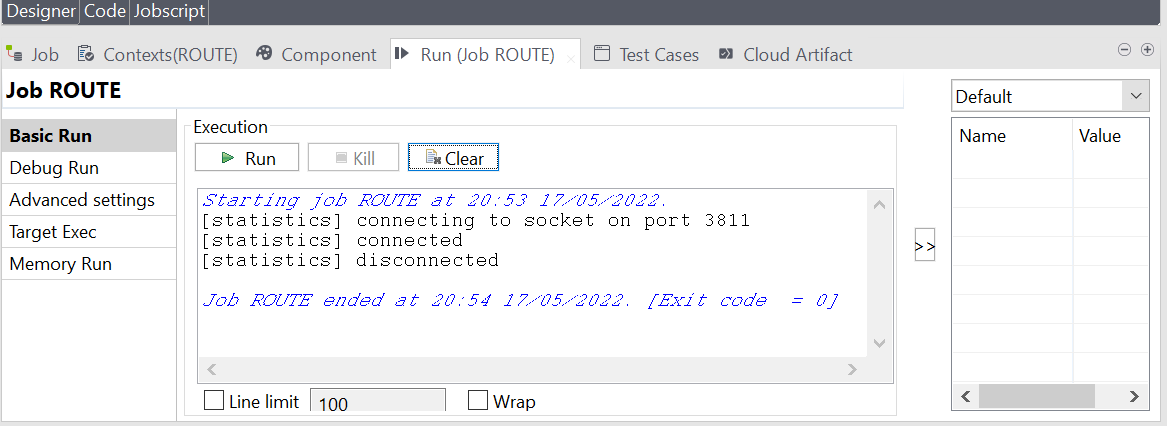


Change the Type to Date. Drag the variable field to the CREATEDATE field on the destination schema. This will create a mapping from the date value created by the expression to the CREATEDATE column. Do the same for the MODIFIEDDATE column. Click OK to save changes.



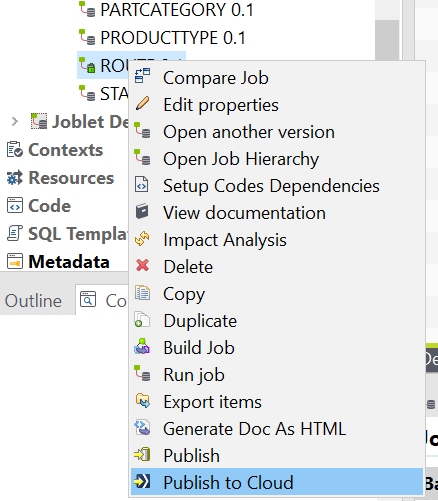
Run the job with Talend Studio. Select the Run (Job *job\_name*) tab and click Run



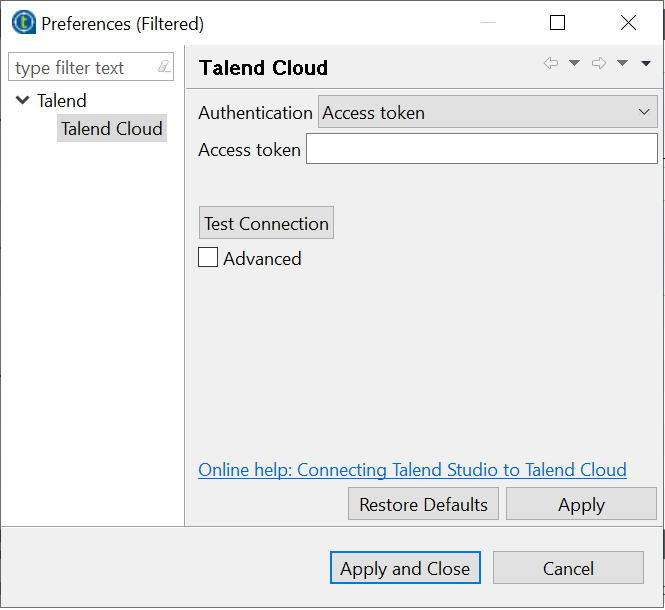


# Publish the Job to Talend Cloud

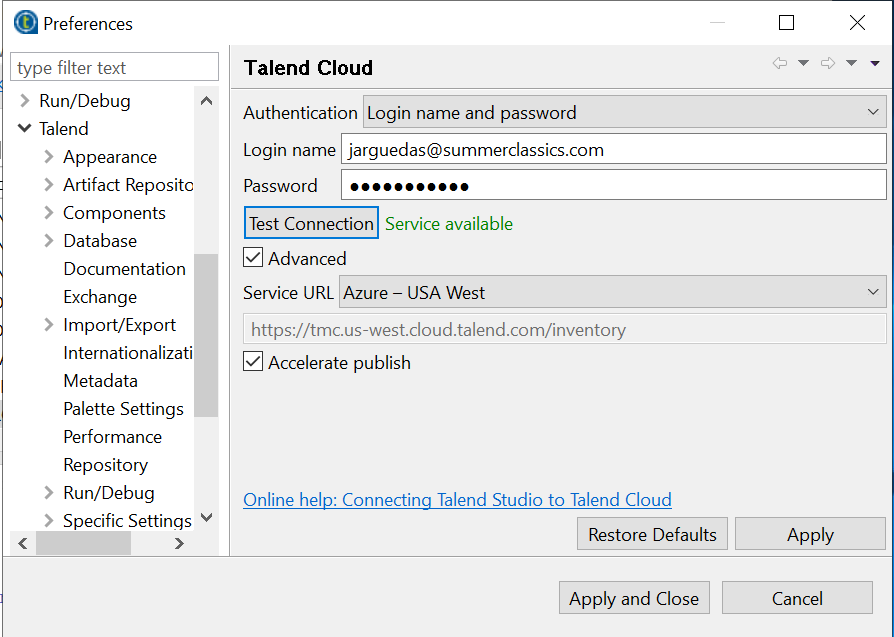
Right-click the job and select Publish to Cloud



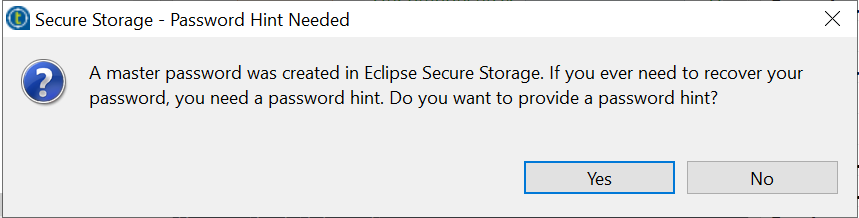
If you receive the below you are not connected to Talend Cloud. Select Login name and password in the Authentication drop-down box.



Fill in the info shown below and click Test Connection. If you receive a Service available message click Apply and Close. This window has issues. If you are pretty sure you entered everything correctly and you get an Unauthorized message when you click Test Connection close the window and open it again from Window -> Preferences.

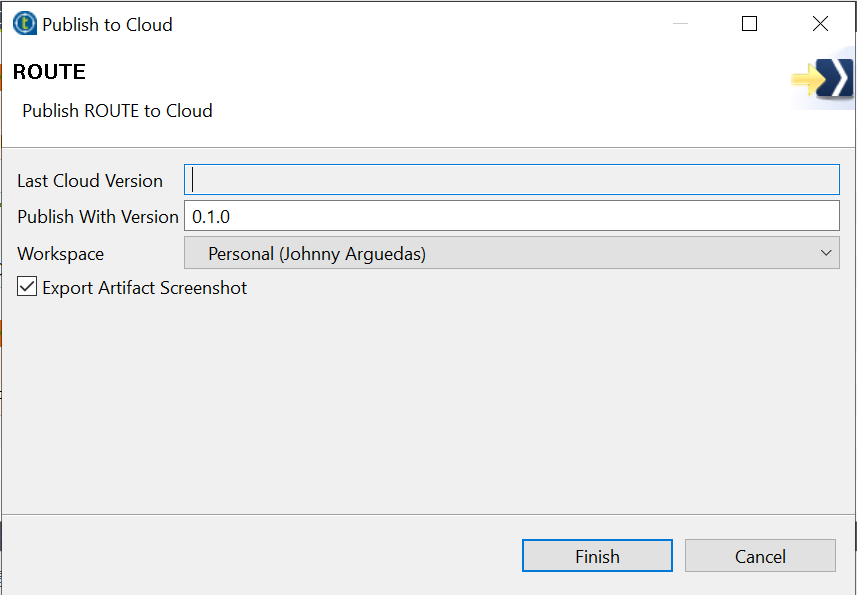


After clicking Apply and Close



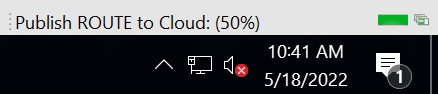
Click Yes.

Publish to Cloud again

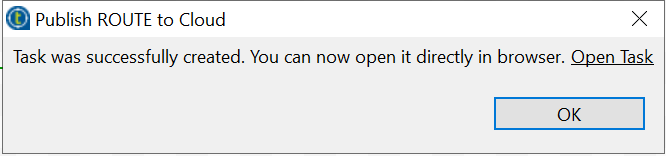


Click Finish

Bottom right corner shows progress



Congratulations. Click OK



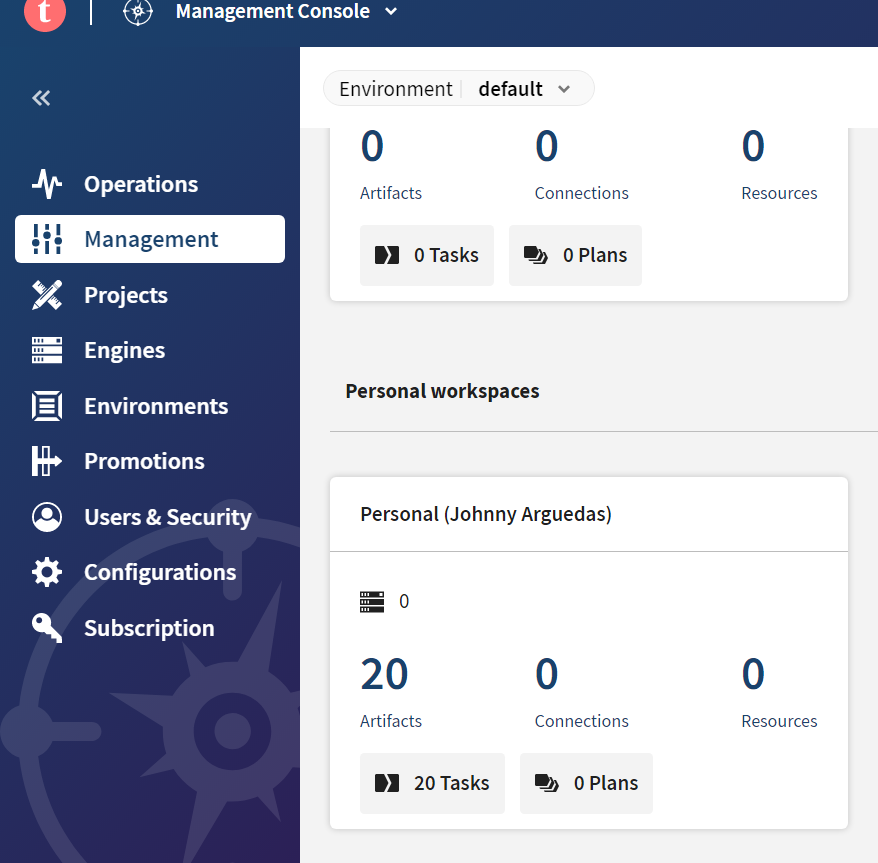
# View and configure job in Talend Cloud

Login to Talend Cloud using the following:

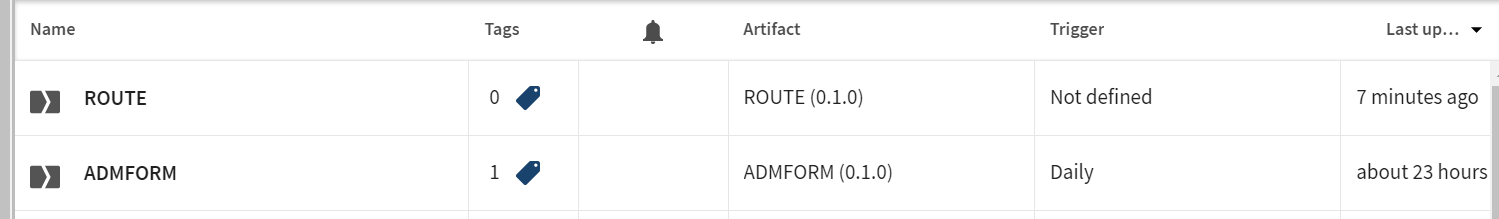
Talend Cloud: <https://portal.us-west.cloud.talend.com/Database>

Username: [jarguedas@summerclassics.com](mailto:jarguedas@summerclassics.com)

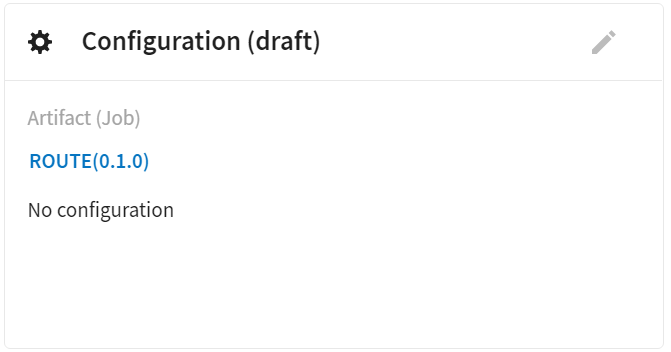
Password: Eifoohei3u-



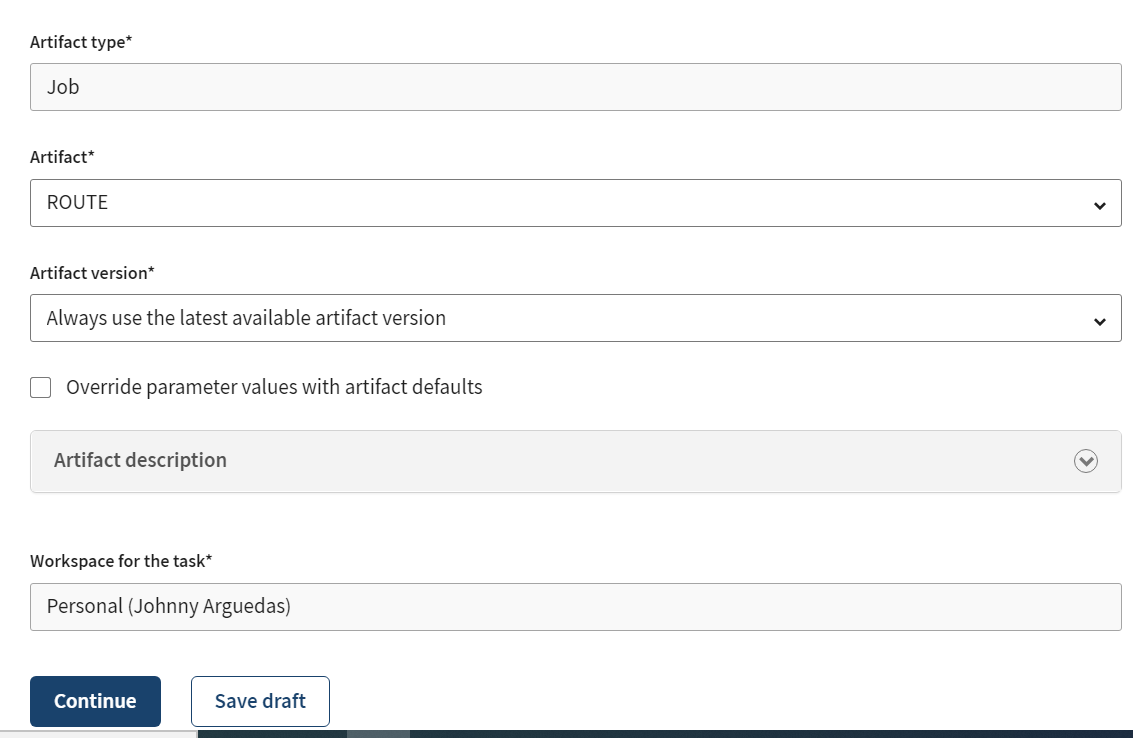
Click on Tasks under Personal (Johnny Arguedas). The new ROUTE job has been added.



Click on the job name. Click on the pencil icon to the right of Configuration (draft) to edit the configuration



Click Continue to configure the engine

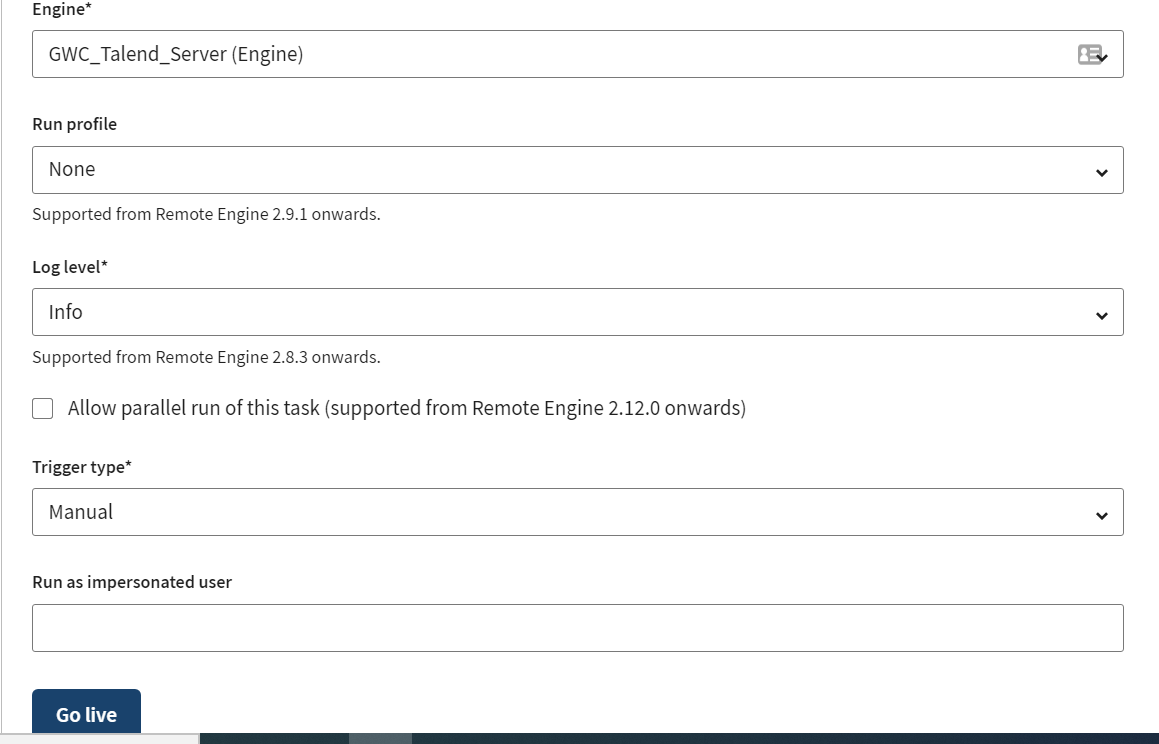


Change the following

Engine: GWC\_Talend\_Server (Engine)

Log level: Info

Trigger type: Manual

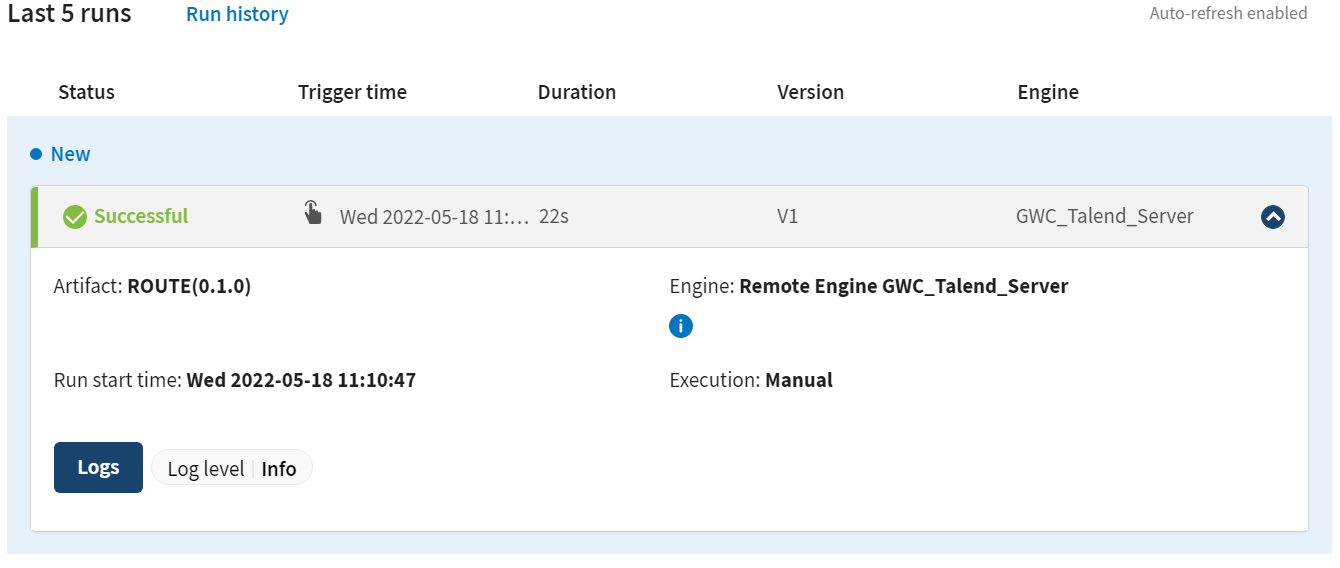


Click Go live.

Click Run Now.

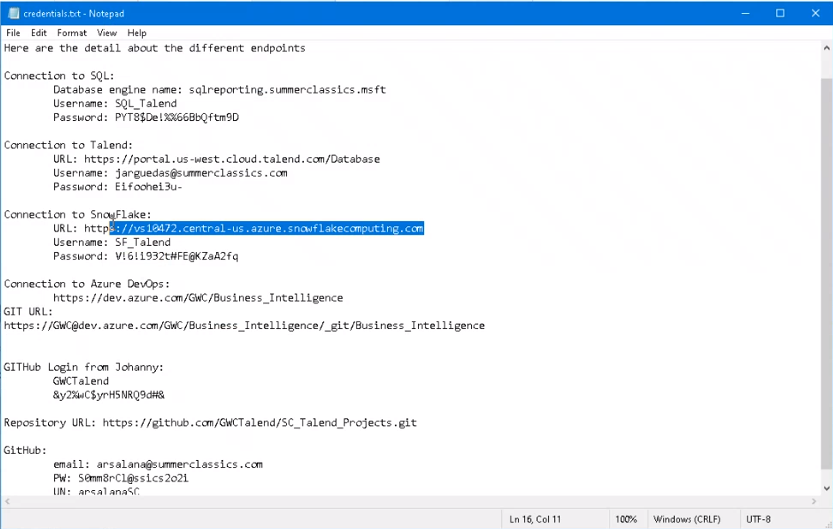
Graphical user interface, application

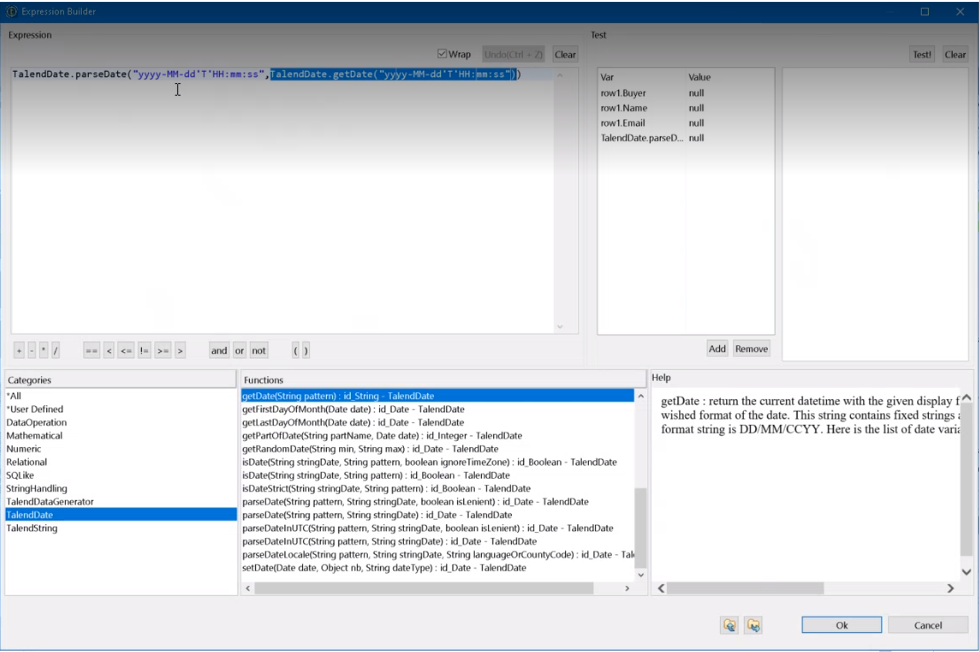
Description automatically generated

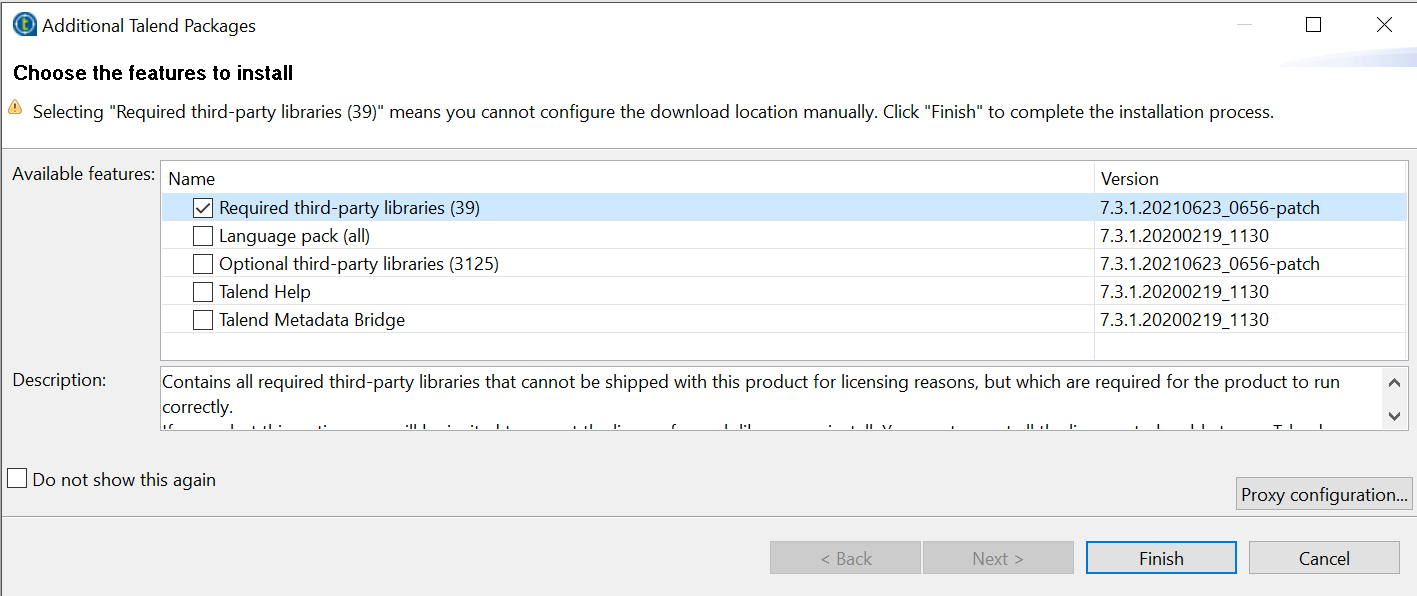


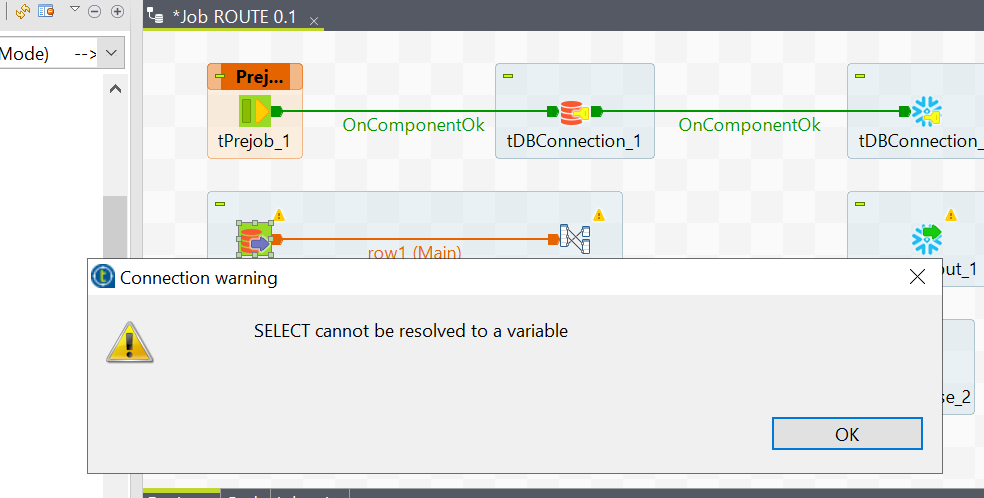
If it runs successfully we will schedule it.

# MISC









MISC

Can use tBDRow component will allow you to run almost any SQL statement, including updates DDL, etc. Only one statement per component.

Built-in component tLogCatcher – Logs\_Errors. Connect tSendMail – Internet to it. System errors are not very helpful.