AT82.02

DATA MODELING AND MANAGEMENT

LAB9: DATA ENGINEERING

Co-funded by the
Erasmus+ Programme
of the European Union

DS&AI

Outline

- Extract Transform Load (ETL)
- Talend Open Studio for Integration
- Tutorial: Movie Rating Prediction
  - Part 1 - Extract
  - Part 2 – Transform & Load

Extract Transform Load

- **Extract** is the process of *reading data* from a database. In this stage, the data is collected, often from multiple and different types of sources.
- **Transform** is the process of *converting the extracted data* from its previous form into the form it needs to be in.
- **Load** is the process of *writing the data* into the target database.

ETL Process

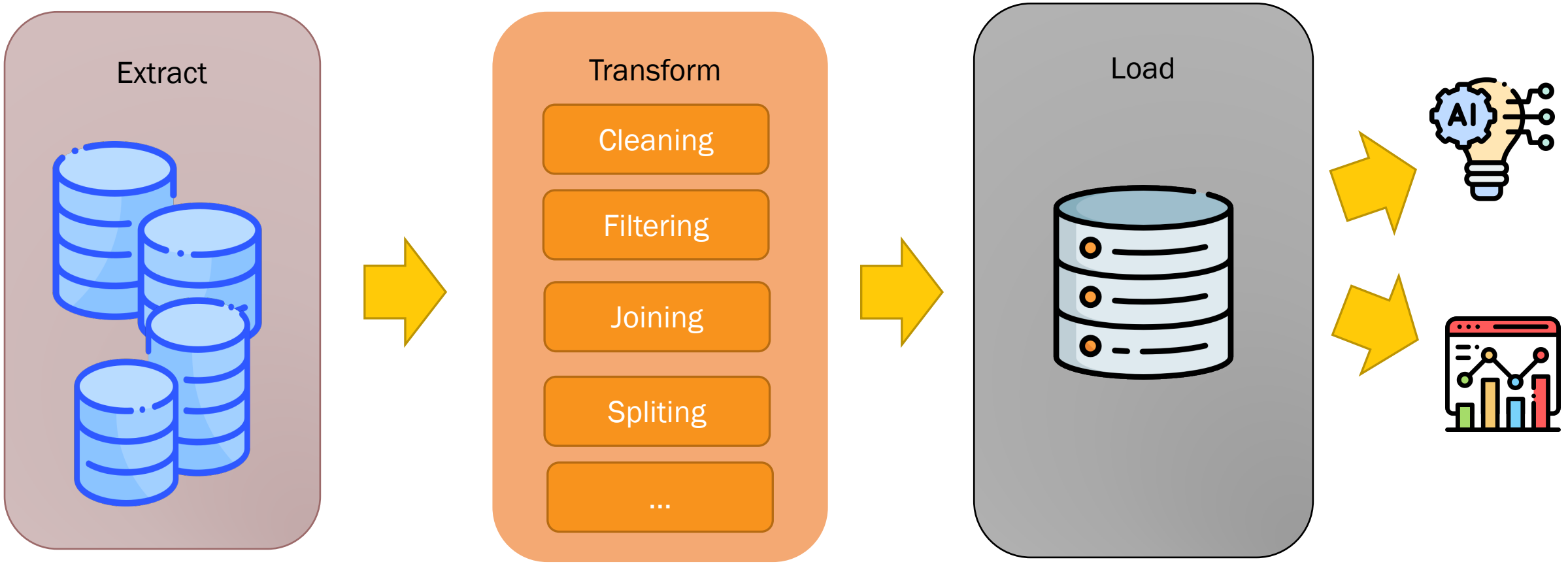
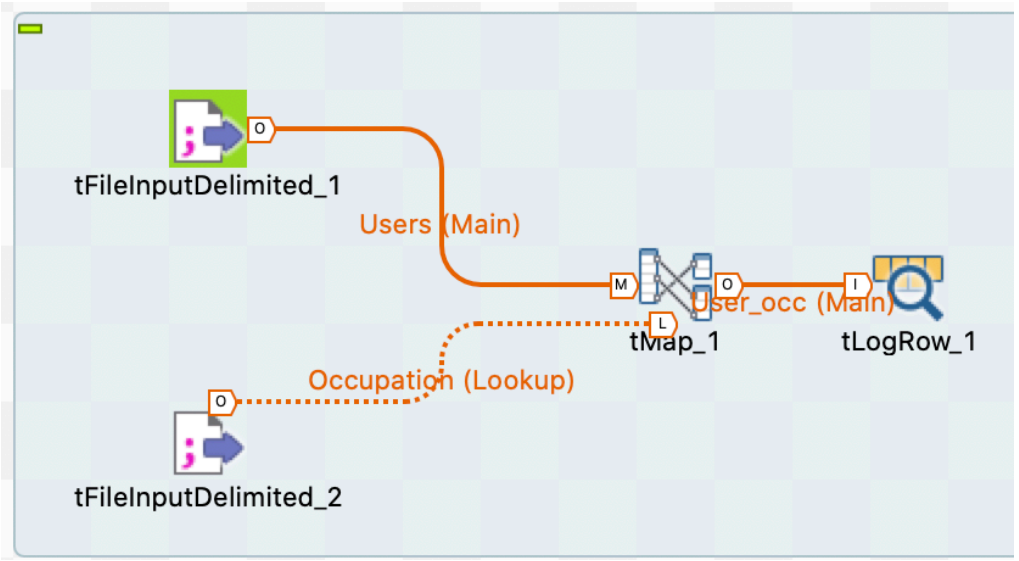


Image source: https://www.flaticon.com

Talend Open Studio for Data Integration

- Talend Open Studio is a free open source ETL tool for Data Integration
- Provide Drag and Drop components and connect them to create and run ETL Jobs.
- The tool will create the Java code for the job automatically
- There are multiple options to connect with Data Sources such as RDBMS, Excel, SaaS Big Data ecosystem



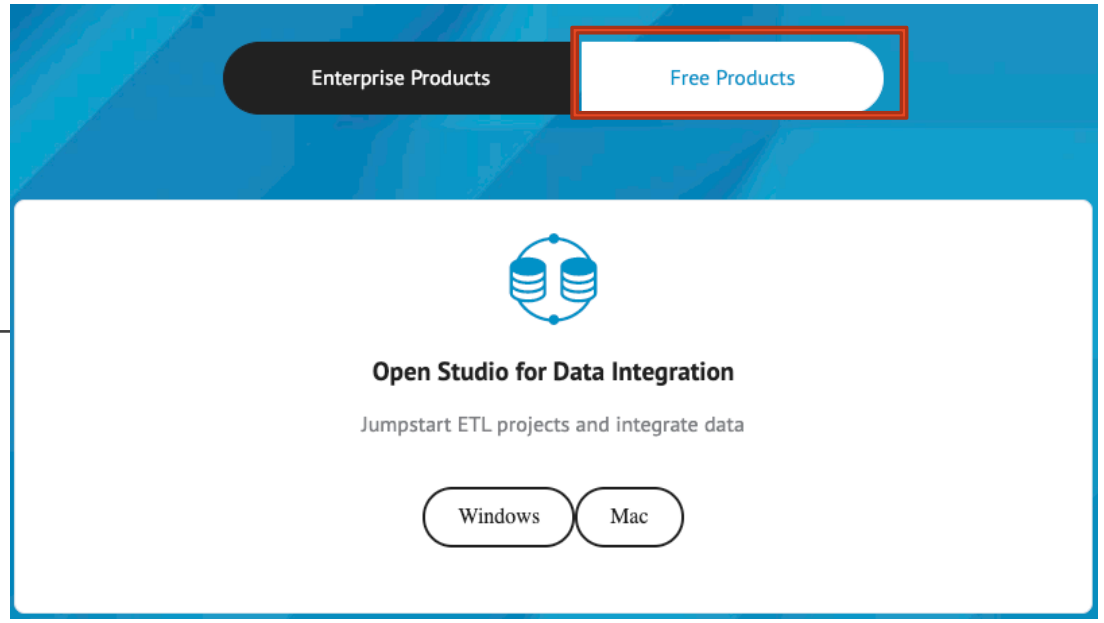
Talend Installation

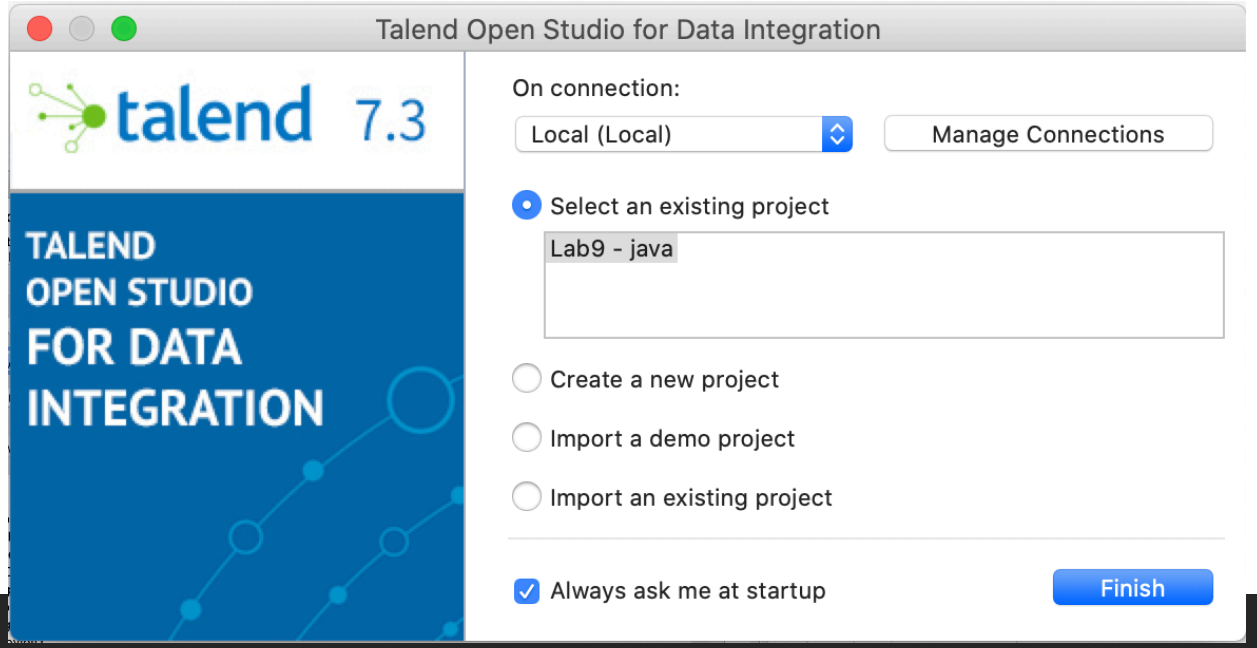
Open Studio for Data Integration

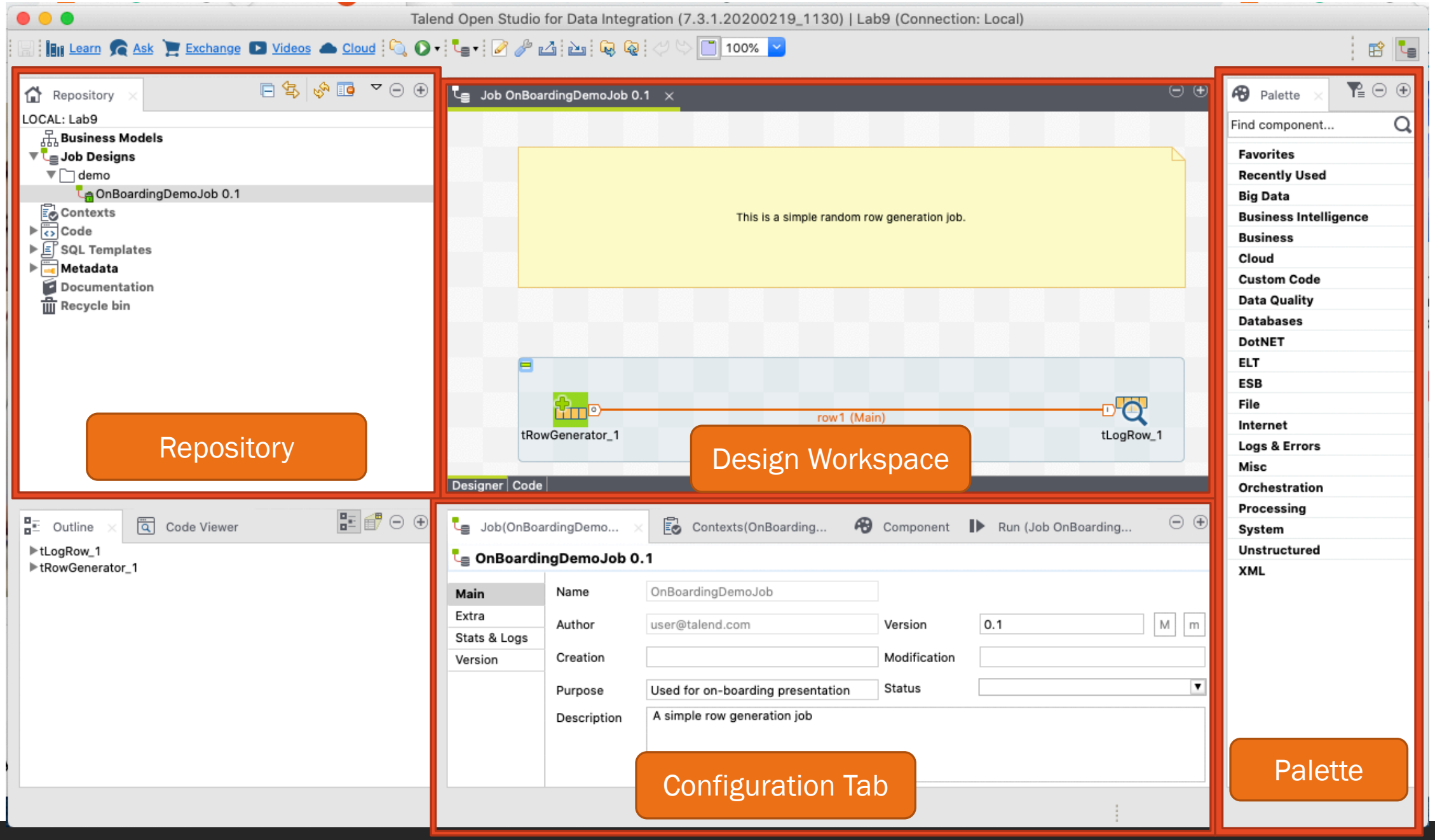
- https://www.talend.com/products/data-integration/data-integration-open-studio/

Cloud Integration (Free Trial):

- https://www.talend.com/free-trial/






Talend Open Studio for Data Integration (7.3.1.20200219_1130) | Lab9 (Connection: Local)
Learn
Ask
Exchange
Videos
Cloud
100%
Repository
LOCAL: Lab9
Business Models
Job Designs
demo
OnBoardingDemoJob 0.1
Contexts
Code
SQL Templates
Metadata
Documentation
Recycle bin
Repository
Job OnBoardingDemoJob 0.1
This is a simple random row generation job.
tRowGenerator_1
row1 (Main)
tLogRow_1
Design Workspace
Designer
Code
Palette
Find component...
Favorites
Recently Used
Big Data
Business Intelligence
Business
Cloud
Custom Code
Data Quality
Databases
DotNET
ELT
ESB
File
Internet
Logs & Errors
Misc
Orchestration
Processing
System
Unstructured
XML
Palette
Outline
Code Viewer
tLogRow_1
tRowGenerator_1
Job(OnBoardingDemo...
Contexts(OnBoarding...
Component
Run (Job OnBoarding...
OnBoardingDemoJob 0.1
Main
Extra
Stats & Logs
Version
Name
OnBoardingDemoJob
Author
user@talend.com
Version
0.1
M
m
Creation
Modification
Purpose
Used for on-boarding presentation
Status
Description
A simple row generation job
Configuration Tab

Components for Data Integration

No	Component	Description
1	tMysqlConnection	Connects to MySQL database defined in the component.
2	tMysqlInput	Runs database query to read a database and extract fields (tables, views etc.) depending on the query.
3	tMysqlOutput	Used to write, update, modify data in a MySQL database.
4	tFileInputDelimited	Reads a delimited file row by row and divides them into separate fields and passes it to the next component.
5	tFileInputExcel	Reads an excel file row by row and divides them into separate fields and passes it to the next component.

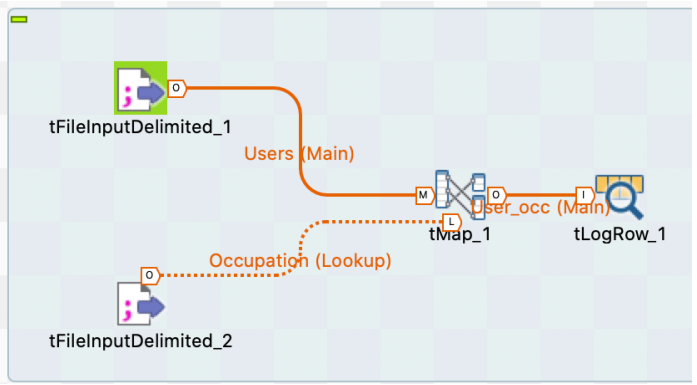
Components for Data Integration

No	Component	Description
6	tFileList	Gets all the files and directories from a given file mask pattern.
7	tFileArchive	Compresses a set of files or folders in to zip, gzip or tar.gz archive file.
8	tRowGenerator	Provides an editor where you can write functions or choose expressions to generate your sample data.
9	tMsgBox	Returns a dialog box with the message specified and an OK button.
10	tLogRow	Monitors the data getting processed. It displays data/output in the run console
11	tPreJob	Defines the sub jobs that will run before your actual job starts

Components for Data Integration

No	Component	Description
12	tMap	Acts as a plugin in Talend studio. It takes data from one or more sources, transforms it, and then sends the transformed data to one or more destinations.
13	tJoin	Joins 2 tables by performing inner and outer joins between the main flow and the lookup flow.
14	tJava	Enables you to use personalized java code in the Talend program.
15	tRunJob	Manages complex job systems by running one Talend job after another.
16	tUniqRow	Compares entries and sorts out duplicate entries from the input flow.

Definition



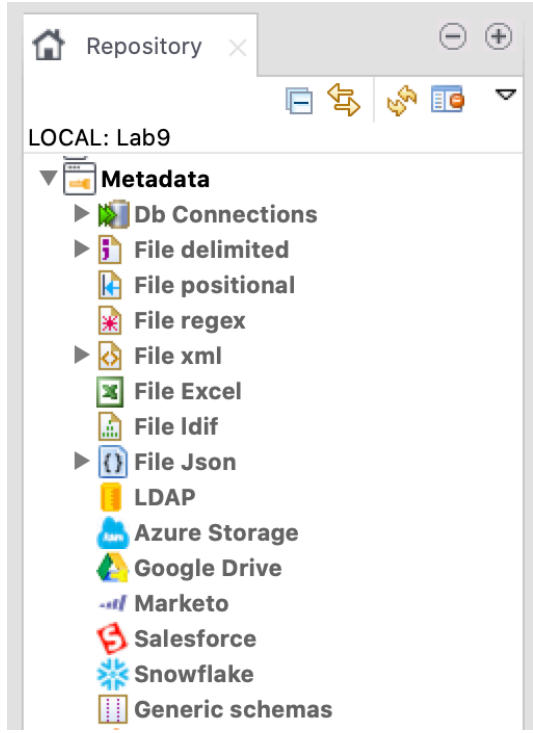
Example Job

Job Design

- This is the graphical representation of the business model.
- In this design, one or more components are connected with each other to run a data integration process.

Metadata

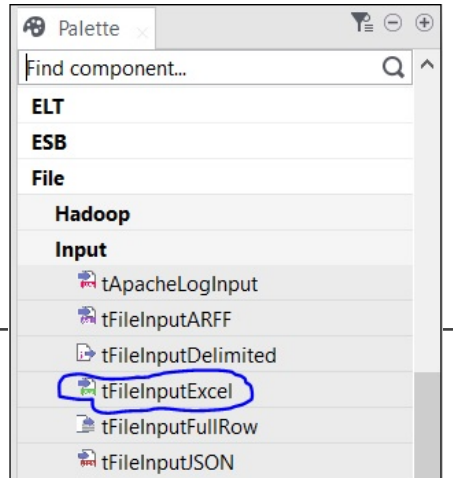
- Metadata basically means data about data.
- The main use of metadata in Talend Open Studio is that you can use these data sources in several jobs just by a simple drag and drop from the Metadata in repository panel.

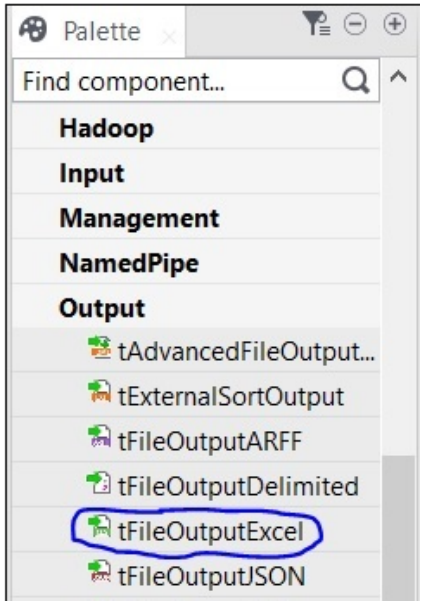


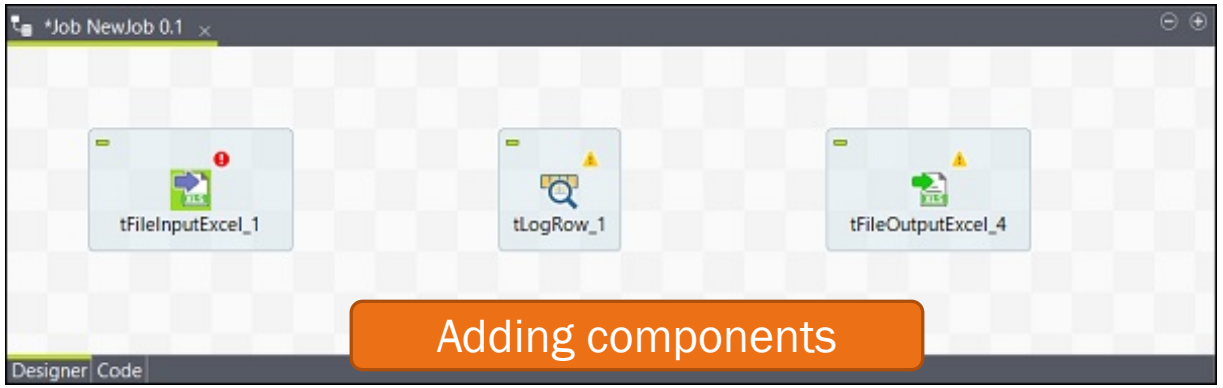
Metadata

Job Design

1) Create a job
2) Adding Components to Job Design
3) Connecting the components
4) Configuring the components
5) Executing the Job





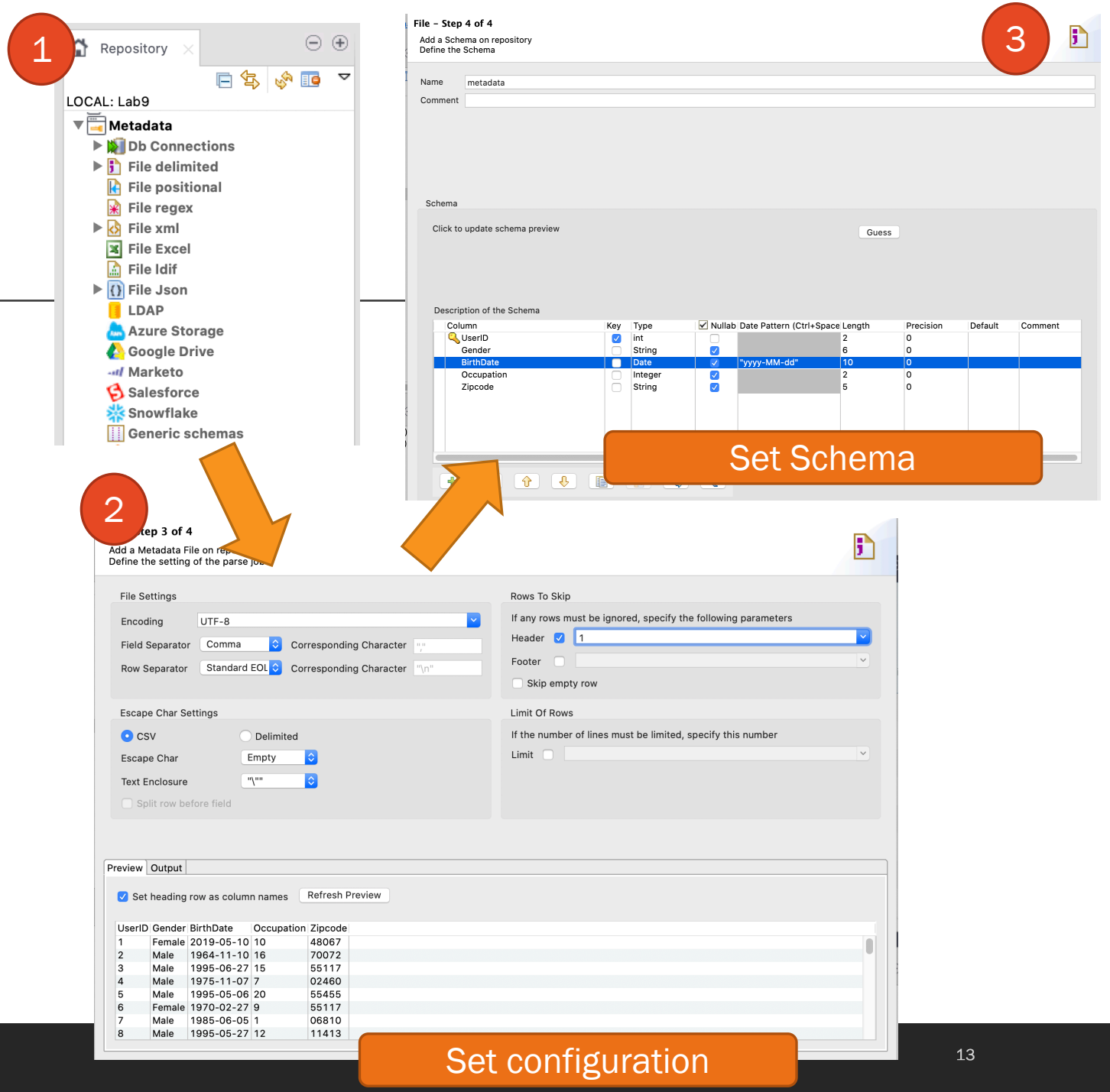


Metadata

1) Select type of Metadata
2) Set Configuration of the data

Example: File Delimited (CSV)

- Encoding
- Field Separator
- Schema



Movie Rating Prediction

- Movie rating is an important element to decide movie quality.
- People prefer to use rating as reference to decide before deciding to watch a movie or not.
- We plan to use historical values of the movie as features (e.g. user profile, movie category, user rating) to predict movie rating before the movie released.

Movie Rating Prediction

Datasets: Movies, User Profile , Movie Rating by Users

The variables used for **input**:

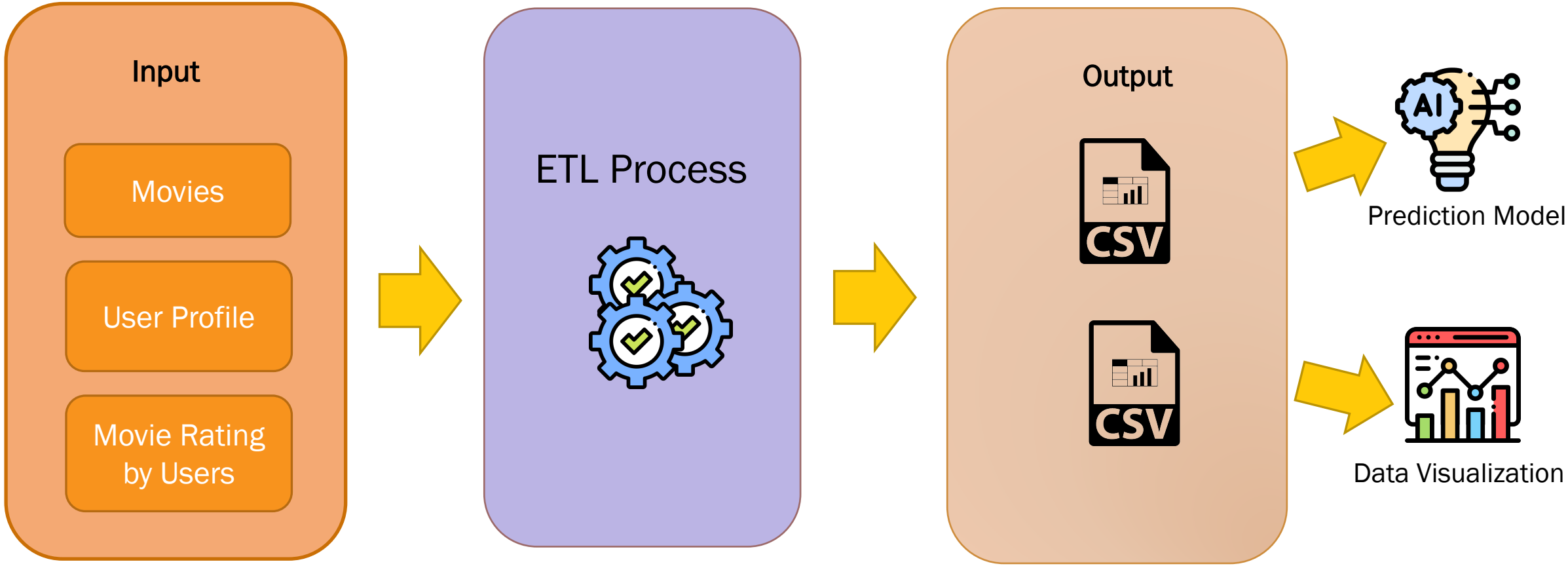
- Age , Gender , Occupation, Movie Category

The variables used for **model prediction**:

- User Rating

source: https://www.kaggle.com/sherinclaudia/movie-rating-prediction

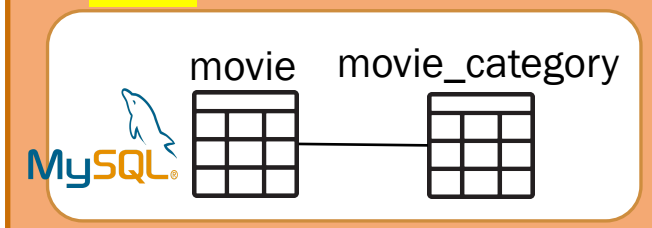
Movie Rating Prediction



Movie Rating Prediction

Input

Movie



Relational Database

User Profile

CSV File

Movie Rating by Users

MongoDB

ETL Process

Extract

Step1: get movie data from RDB and convert data to CSV file
Step2: get user profile in CSV format
Step3: get movie rating by users from MongoDB
- 3.1 export data to JSON file
- 3.2 convert Json to CSV file

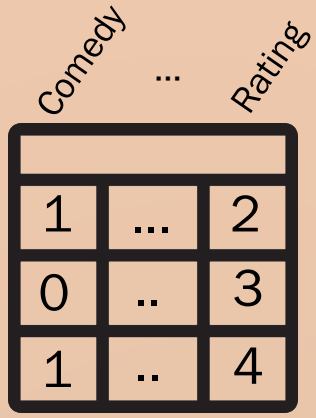
Transform

Step.4 Prepare data for desired datasets
- 4.1 Transform movie category data
- 4.2 Remove Duplicate Records
- 4.3 Integrate 3 datasets
- 4.4 Convert birthdate to Age
- 4.5 Convert Gender to numeric
- 4.6 Filter age >= 7 and Gender = 1|2
- 4.7 Merge output of 4.1 and 4.6

Load

Step.5 Write output CSV files

Output



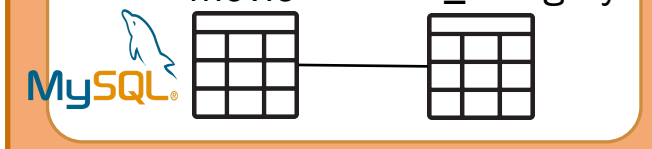
CSV for Prediction

CSV for Visualization

Movie Rating Prediction

Input

Movie



Relational Database

User Profile

CSV File

Movie Rating by Users

MongoDB

ETL Process

Extract

Step1: get movie data from RDB and convert data to CSV file
Step2: get user profile in CSV format
Step3: get movie rating by users from MongoDB
3.1 export data to JSON file
3.2 convert Json to CSV file

Transform

Step.4 Prepare data for desired datasets
4.1 Transform movie category data
4.2 Remove Duplicate Records
4.3 Integrate 3 datasets
4.4 Convert birthdate to Age
4.5 Convert Gender to numeric
4.6 Filter age >= 7 and Gender = 1|2
4.7 Merge output of 4.1 and 4.6

Load

Step.5 Write output CSV files

Talend

Task1 for step 1
- 1.1 Create Metadata for DB Connection
- 1.2 Create TJob#1
- 1.3 Create Metadata for movie

Task2 for step 2
- Create Metadata for user profile

Task3 for step 3
- 3.1 Create Metadata for movie rating (Json)
- 3.2 Create TJob#2
- 3.3 Create Metadata for movie rating (File Delimited)

Task4 for step 4.1
- 4.1 Create TJob#3
- 4.2 Create Metadata for movie category

Task5 for step 4.2-5
- Create TJob#4

Desired Output for Prediction

UserID	MovieID	Gender2	Age	Occupation	Animation	Children_s	Comedy	Adventure	Fantasy	Romance	Drama	Action	Crime	...	Rating
2	1357	1	50	16	0	0	0	0	0	1	1	0	0	...	5
2	3068	1	50	16	0	0	0	0	0	0	1	0	0	...	4
2	1537	1	50	16	0	0	1	0	0	0	0	0	0	...	4
2	647	1	49	16	0	0	0	0	0	0	1	0	0	...	3
2	2194	1	50	16	0	0	0	0	0	0	1	1	1	...	4
2	648	1	50	16	0	0	0	1	0	0	0	1	0	...	4
2	2268	1	49	16	0	0	0	0	0	0	1	0	1	...	5
2	2628	1	50	16	0	0	0	1	1	0	0	1	0	...	3
2	1103	1	50	16	0	0	0	0	0	0	1	0	0	...	3
2	2916	1	50	16	0	0	0	1	0	0	0	1	0	...	3
2	3468	1	50	16	0	0	0	0	0	0	1	0	0	...	5
2	1210	1	49	16	0	0	0	1	0	1	0	1	0	...	4
2	1792	1	50	16	0	0	0	0	0	0	0	1	0	...	3
2	1687	1	49	16	0	0	0	0	0	0	0	1	0	...	3
2	1213	1	50	16	0	0	0	0	0	0	1	0	1	...	2
2	3578	1	50	16	0	0	0	0	0	0	1	1	0	...	5
2	2881	1	50	16	0	0	0	0	0	0	0	1	0	...	3
2	3030	1	49	16	0	0	1	0	0	0	1	0	0	...	4
2	1217	1	49	16	0	0	0	0	0	0	1	0	0	...	3
2	3105	1	50	16	0	0	0	0	0	0	1	0	0	...	4
2	434	1	50	16	0	0	0	1	0	0	0	1	1	...	

Gender Age Occupation

Movie category

Rating

Desired Output for Visualization

A	B	C	D	E	F	G	H	I
UserID	MovieID	Title	Year	Gender	Gender2	Age	Timestamp	Rating
2	1357	Shine (1996)	1996	Male	1	50	2015-10-18 15:24:38	5
2	3068	Verdict, The (1982)	1982	Male	1	50	2015-01-14 15:31:09	4
2	1537	Shall We Dance? (Shall We Dansu?) (1996)	1996	Male	1	50	2015-05-22 09:03:30	4
2	647	Courage Under Fire (1996)	1996	Male	1	49	2014-02-21 07:02:49	3
2	2194	Untouchables, The (1987)	1987	Male	1	50	2015-10-25 03:23:12	4
2	648	Mission: Impossible (1996)	1996	Male	1	50	2015-01-07 10:58:19	4
2	2268	Few Good Men, A (1992)	1992	Male	1	49	2014-05-27 15:12:42	5
2	2628	Star Wars: Episode I - The Phantom Menace (1	1999	Male	1	50	2015-11-03 14:11:03	3
2	1103	Rebel Without a Cause (1955)	1955	Male	1	50	2015-06-07 05:36:56	3
2	2916	Total Recall (1990)	1990	Male	1	50	2015-09-19 18:07:05	3
2	3468	Hustler, The (1961)	1961	Male	1	50	2015-08-17 09:42:45	5
2	1210	Star Wars: Episode VI - Return of the Jedi (198	1983	Male	1	49	2014-02-09 14:01:03	4
2	1792	U.S. Marshalls (1998)	1998	Male	1	50	2015-07-02 00:47:44	3
2	1687	Jackal, The (1997)	1997	Male	1	49	2014-05-11 10:57:57	3
2	1213	GoodFellas (1990)	1990	Male	1	50	2014-11-28 21:19:25	2
2	3578	Gladiator (2000)	2000	Male	1	50	2015-06-24 00:51:45	5
2	2881	Double Jeopardy (1999)	1999	Male	1	50	2014-12-29 21:27:23	3
2	3030	Yojimbo (1961)	1961	Male	1	49	2014-10-09 16:27:37	4
2	1217	Ran (1985)	1985	Male	1	49	2014-03-12 16:41:31	3
2	3105	Awakenings (1990)	1990	Male	1	50	2015-10-17 05:59:39	4
2	434	Cliffhanger (1993)	1993	Male	1	50	2015-10-03 23:12:06	2
2	2126	Snake Eyes (1998)	1998	Male	1	49	2014-02-04 15:38:02	3

Part 1

ETL Process

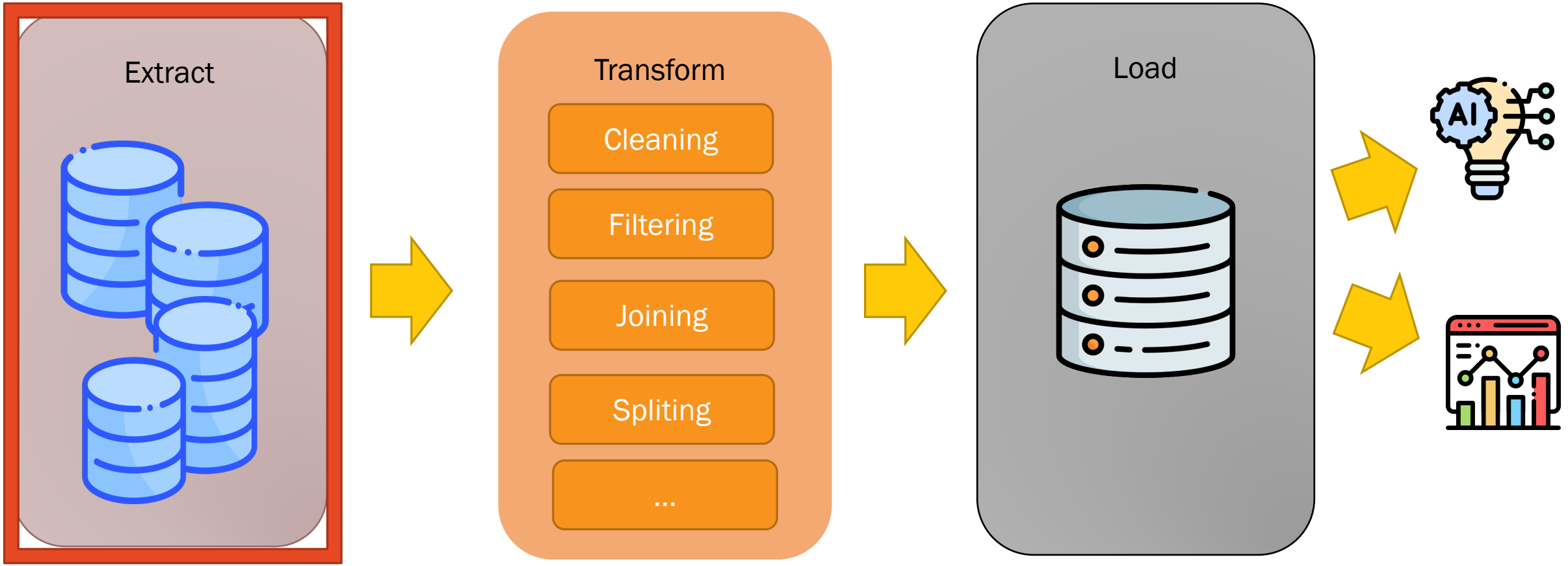
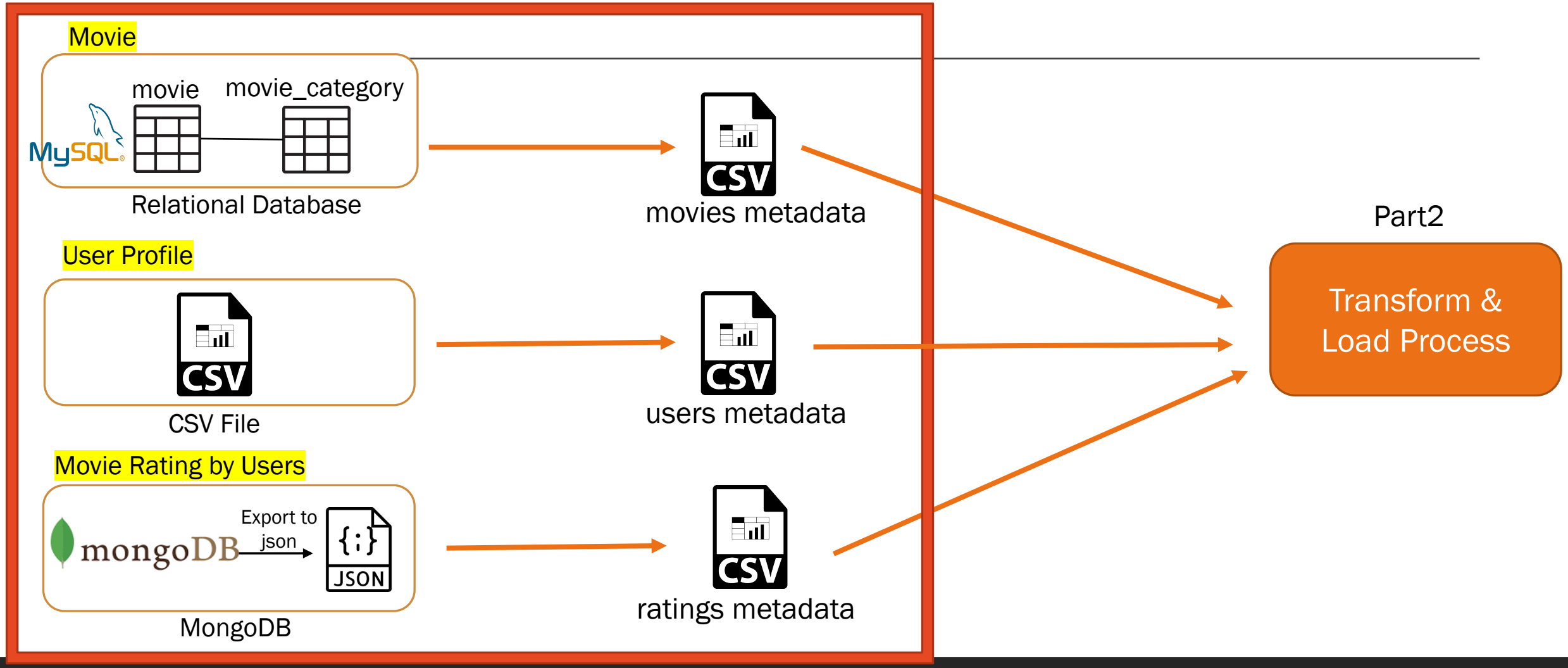


Image source: https://www.flaticon.com

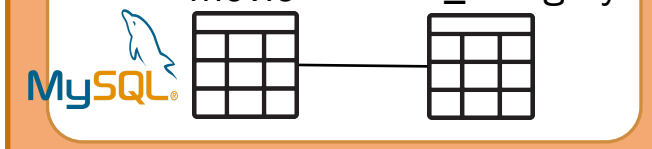
Goal of Part1



Movie Rating Prediction

Input

Movie



Relational Database

User Profile

CSV File

Movie Rating by Users

MongoDB

ETL Process

Extract

Step1: get movie data from RDB and convert data to CSV file
Step2: get user profile in CSV format
Step3: get movie rating by users from MongoDB
3.1 export data to JSON file
3.2 convert Json to CSV file

Transform

Step.4 Prepare data for desired datasets
4.1 Transform movie category data
4.2 Remove Duplicate Records
4.3 Integrate 3 datasets
4.4 Convert birthdate to Age
4.5 Convert Gender to numeric
4.6 Filter age >= 7 and Gender = 1|2
4.7 Merge output of 4.1 and 4.6

Load

Step.5 Write output CSV files

Talend

Task1 for step 1
- 1.1 Create Metadata for DB Connection
- 1.2 Create TJob#1
- 1.3 Create Metadata for movie

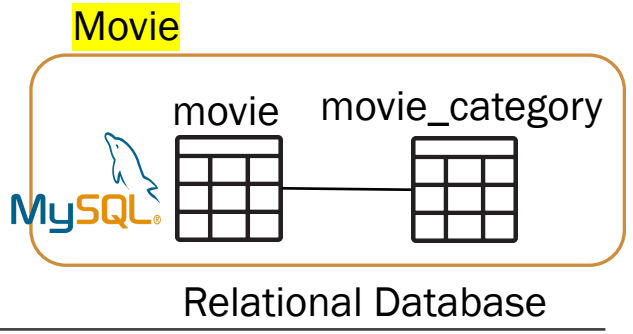
Task2 for step 2
- Create Metadata for user profile

Task3 for step 3
- 3.1 Create Metadata for movie rating (Json)
- 3.2 Create TJob#2
- 3.3 Create Metadata for movie rating (File Delimited)

Task4 for step 4.1
- 4.1 Create TJob#3
- 4.2 Create Metadata for movie category

Task5 for step 4.2-5
- Create TJob#4

Task1 for step 1



Step1: get movie data from RDB and convert data to CSV file

Task1:

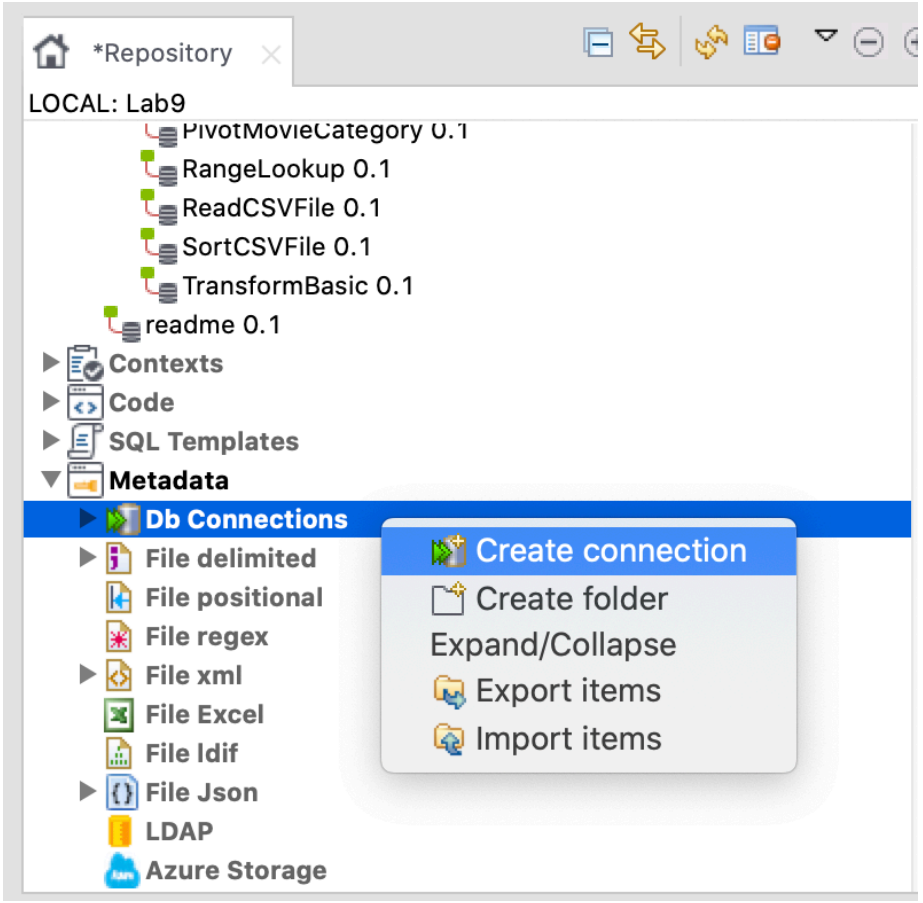
1.1 Create Metadata for DB Connection

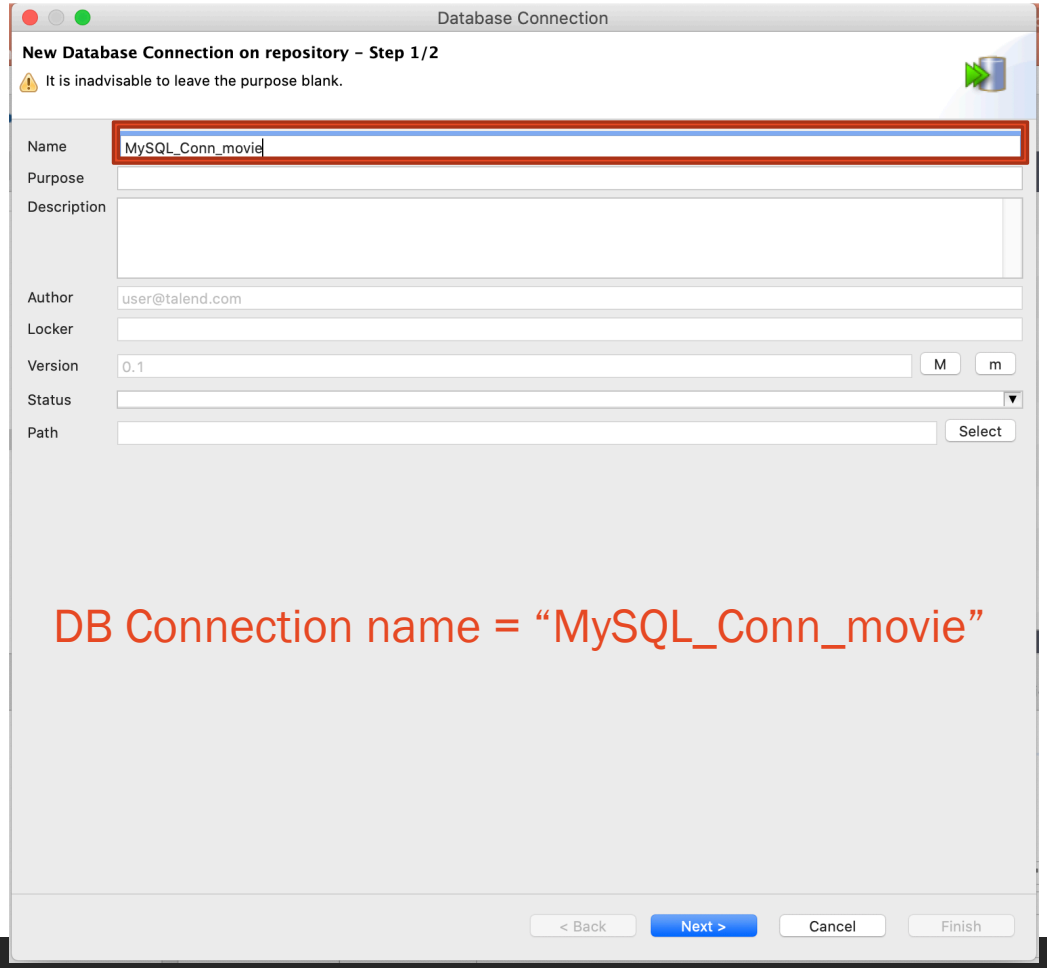
1.2 Create TJob#1 => Convert data in MySQL to CSV file named “movie_data.csv”

1.3 Create Metadata for movie data (File Delimited)

Task1

1.1 Create Metadata for DB Connection (1)

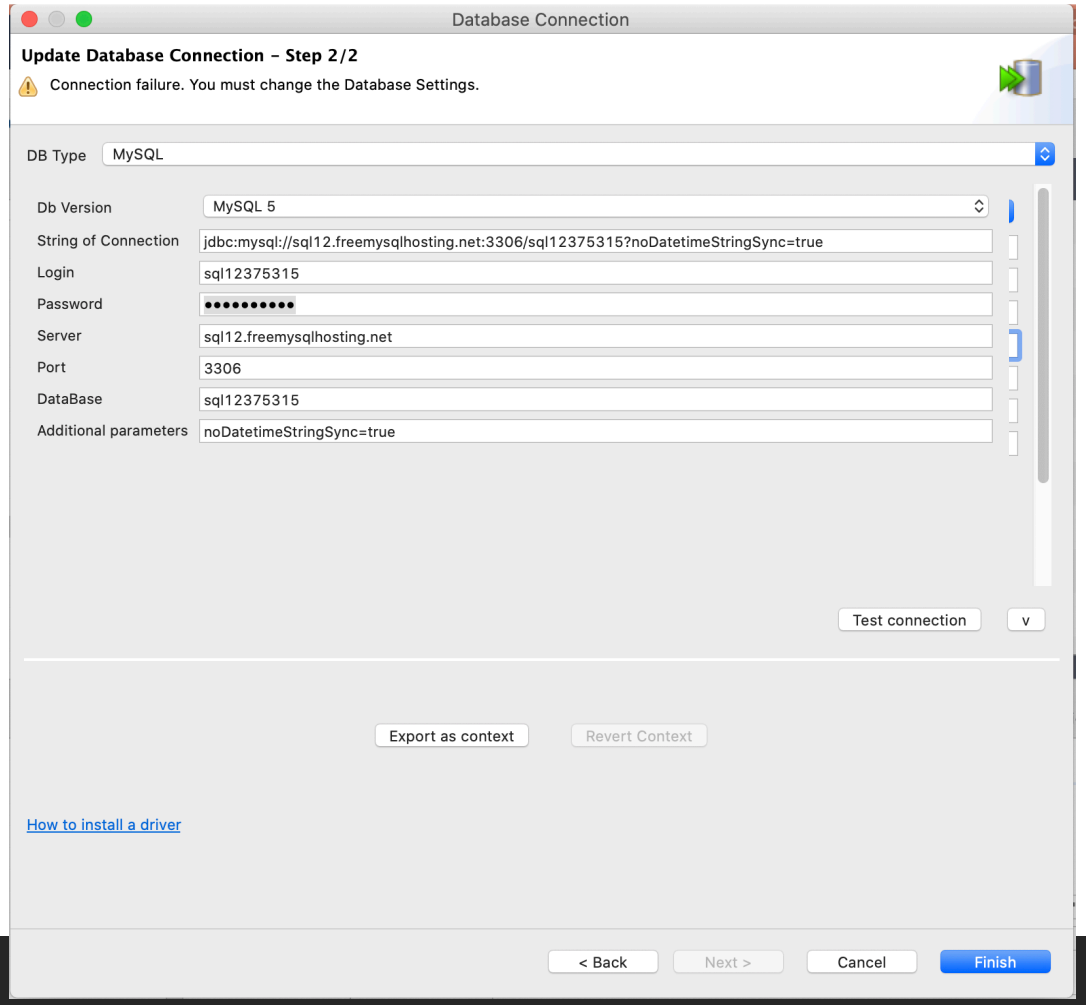




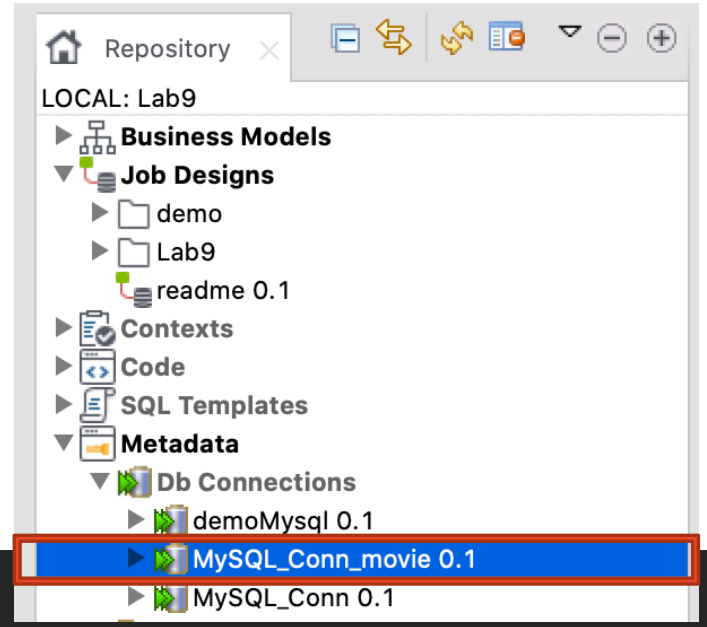
DB Connection name = “MySQL_Conn_movie”

Task1

1.1 Create Metadata for DB Connection (2)

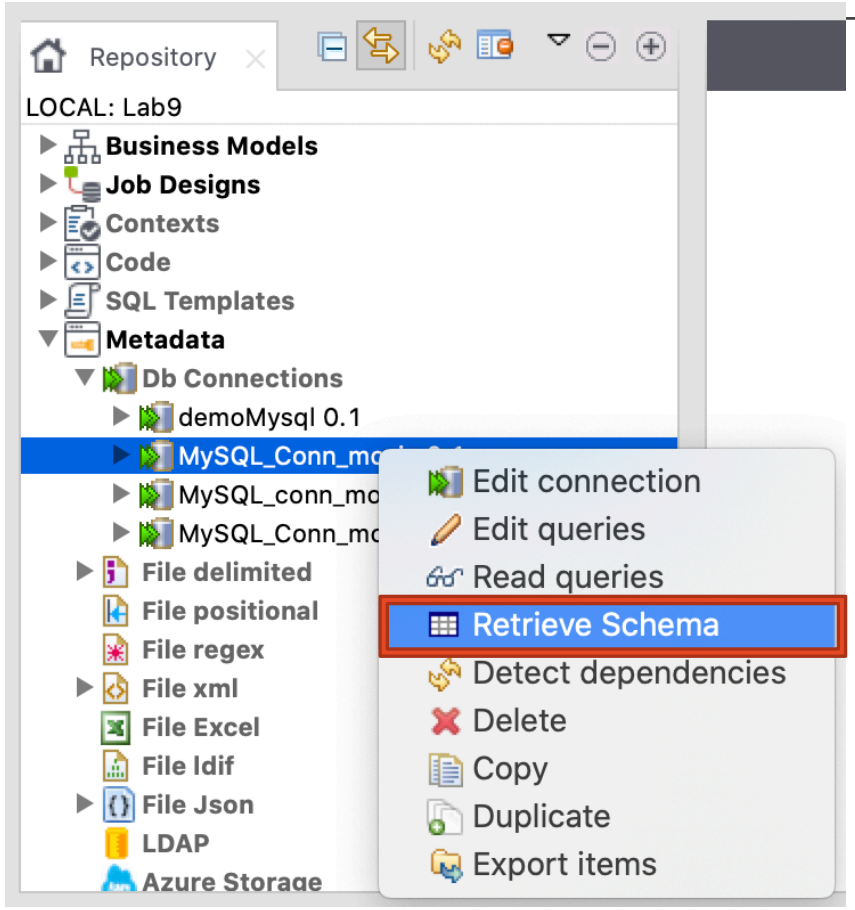


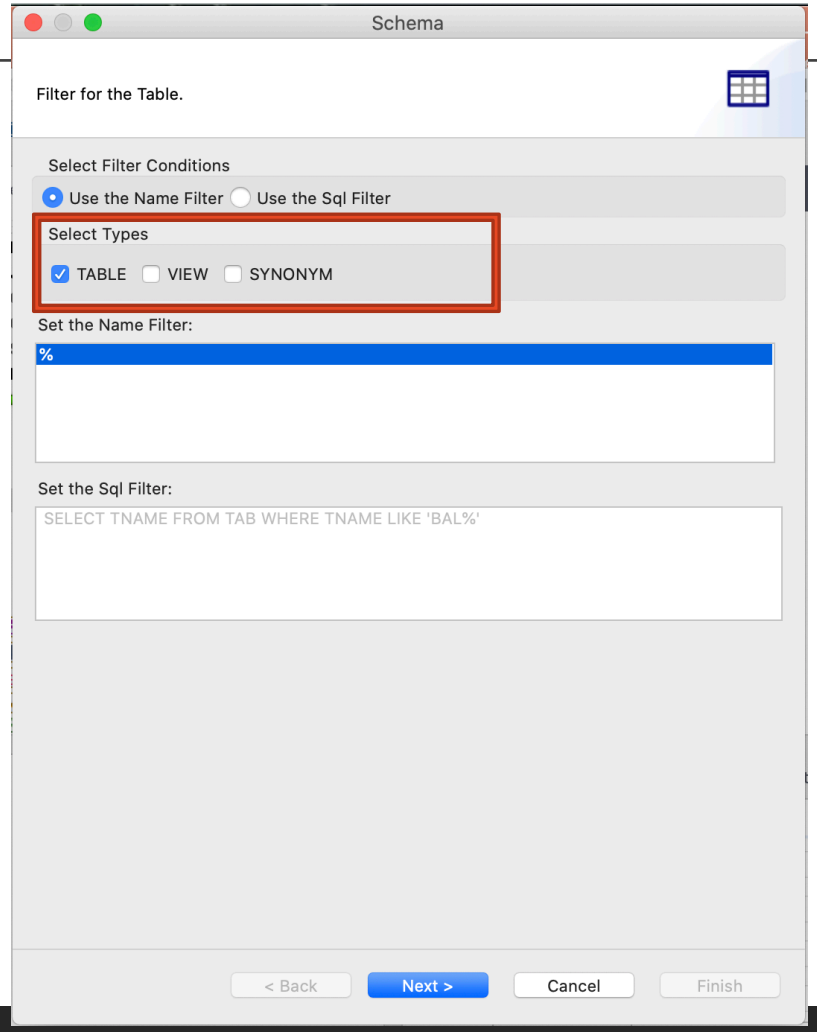
Server: sql12.freemysqlhosting.net
DB Name: sql12375315
Username: sql12375315
Password: mIejQH4MPw
Port number: 3306



Task1

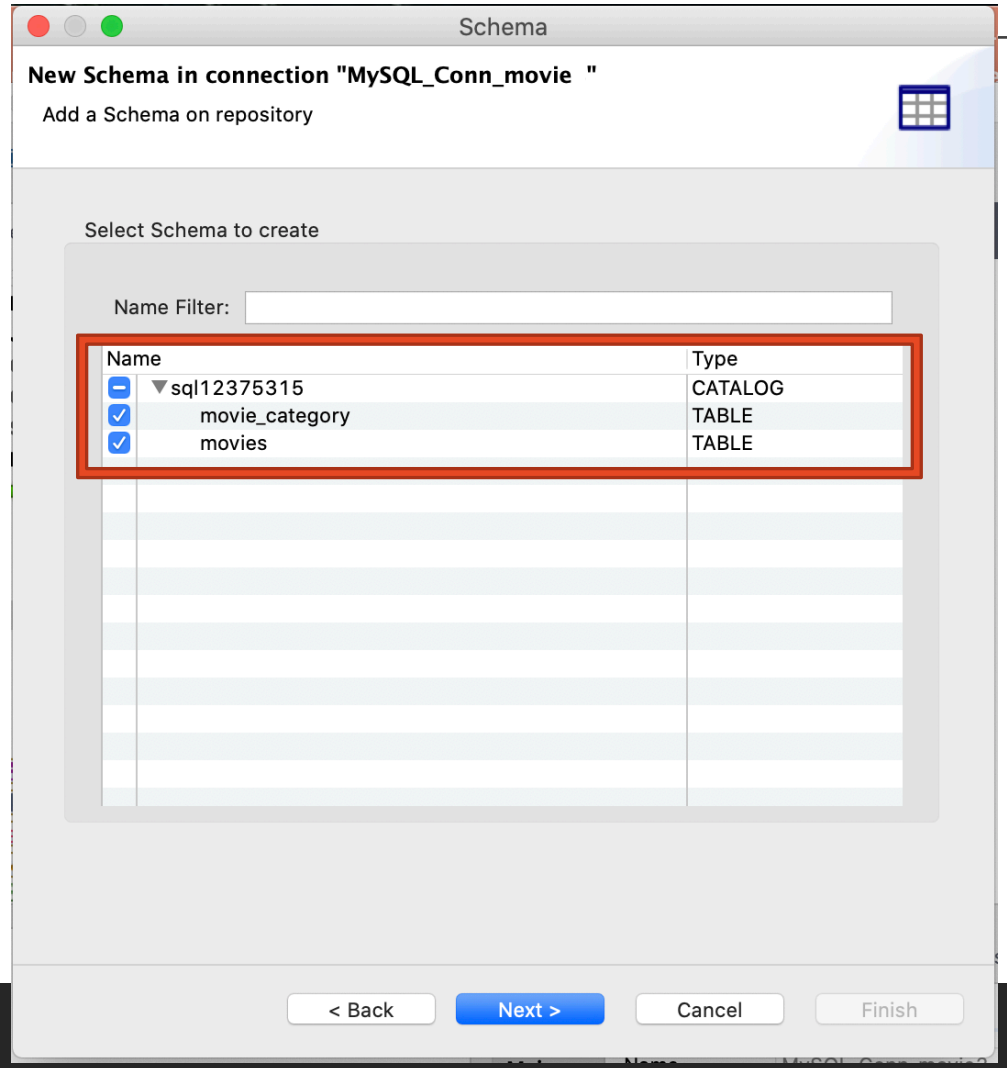
1.1 Create Metadata for DB Connection (3)

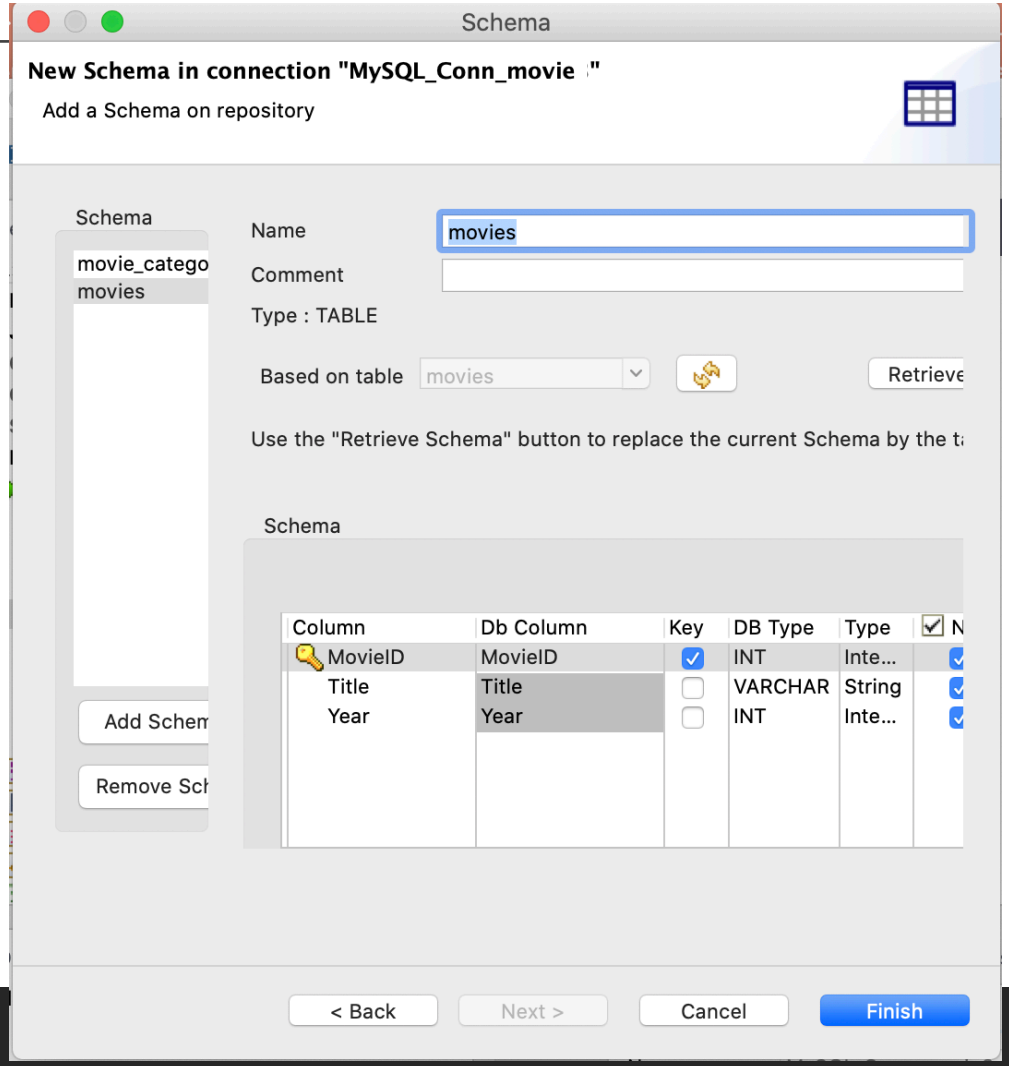




Task1

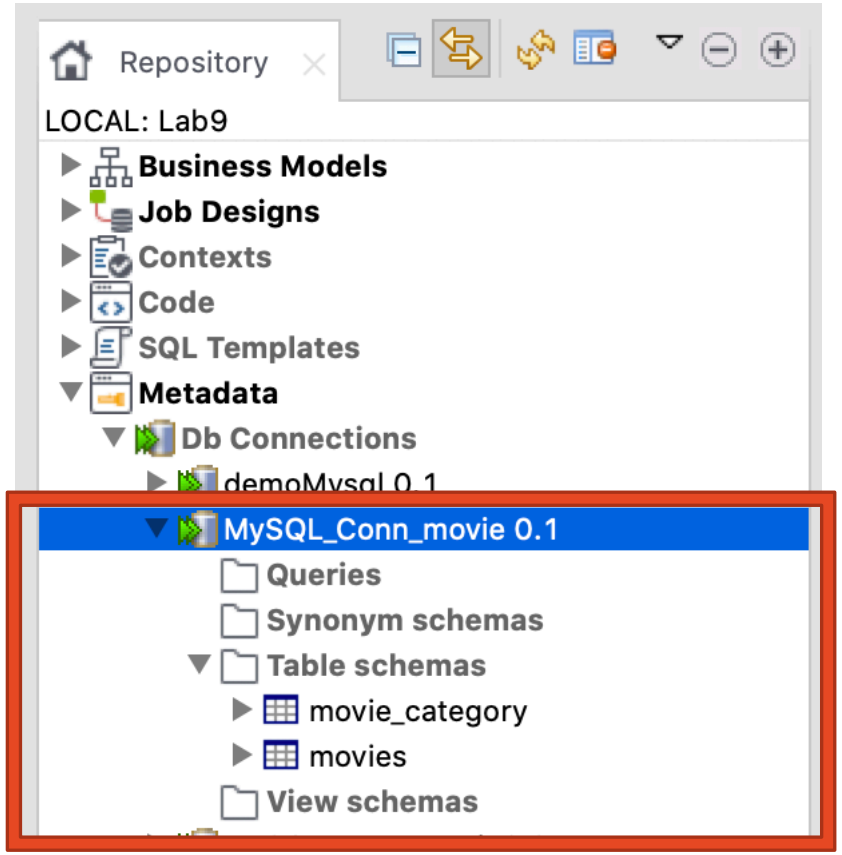
1.1 Create Metadata for DB Connection (4)





Task1

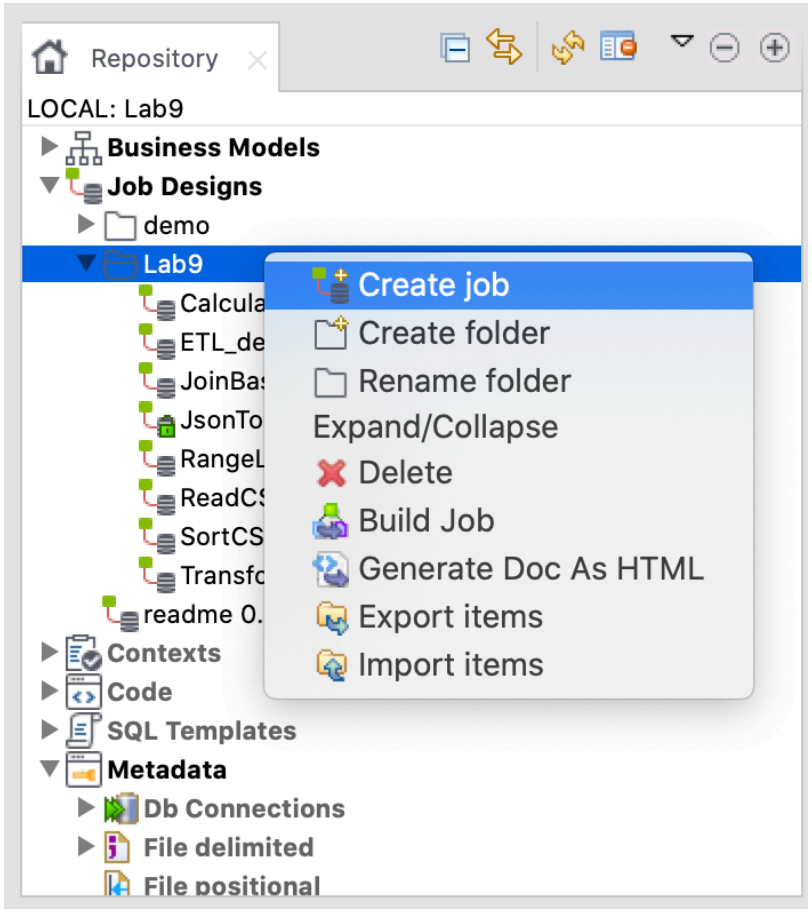
1.1 Create Metadata for DB Connection (5)

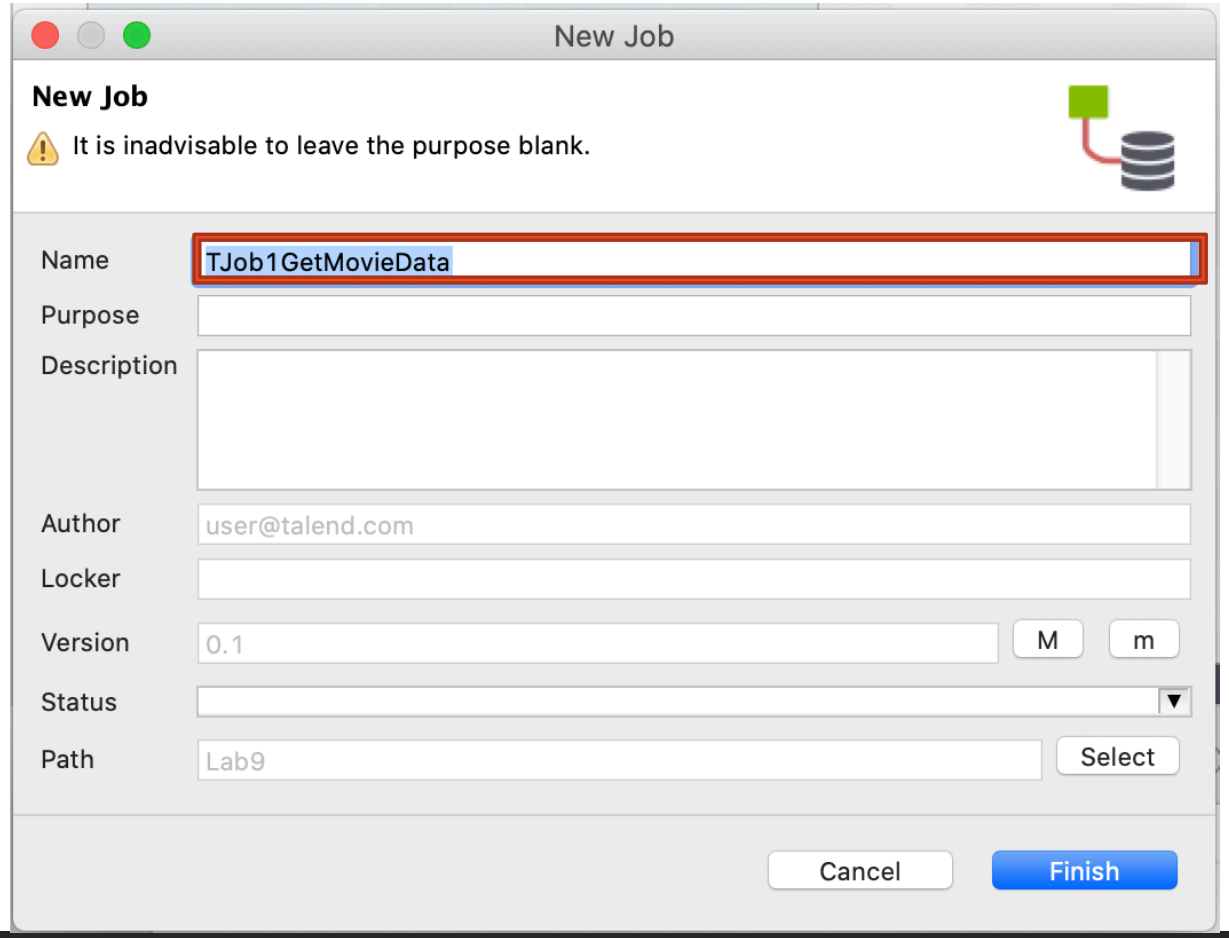


Output of Metadata of DB Connection

Task1

1.2 Create TJob#1: get movie data from RDB and convert data to CSV file (1)

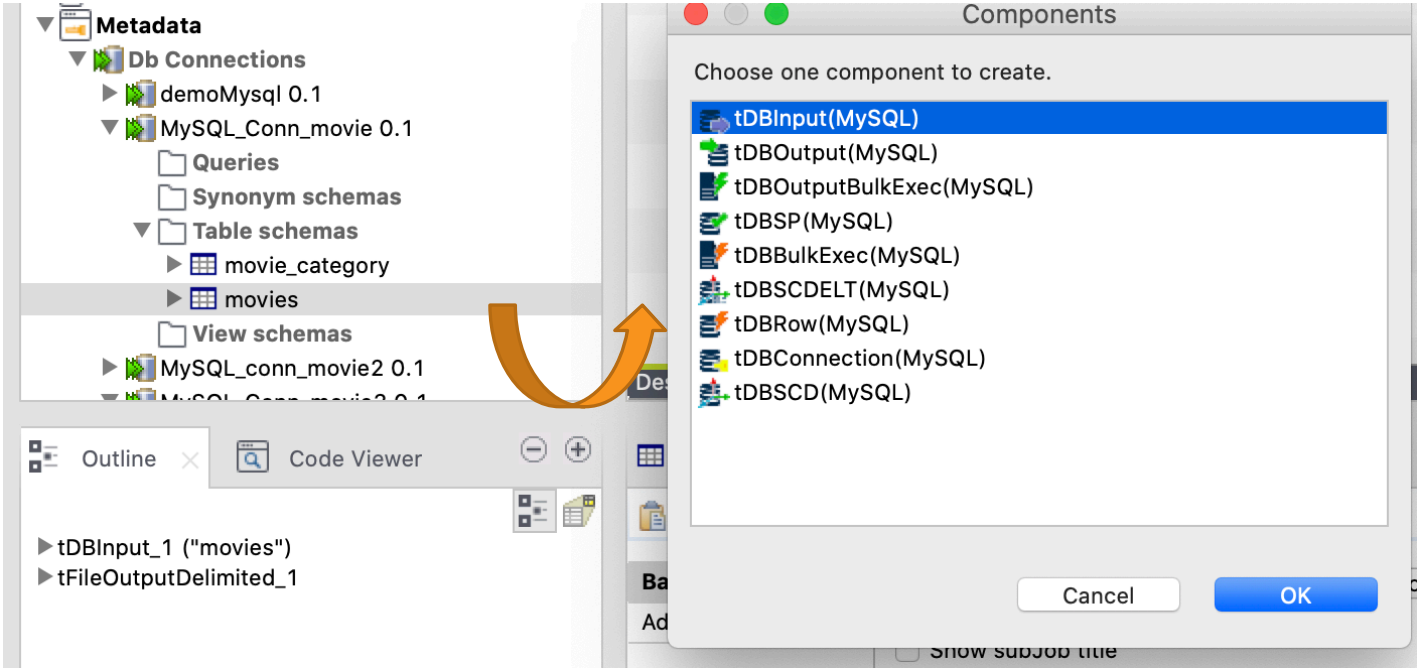




Task1

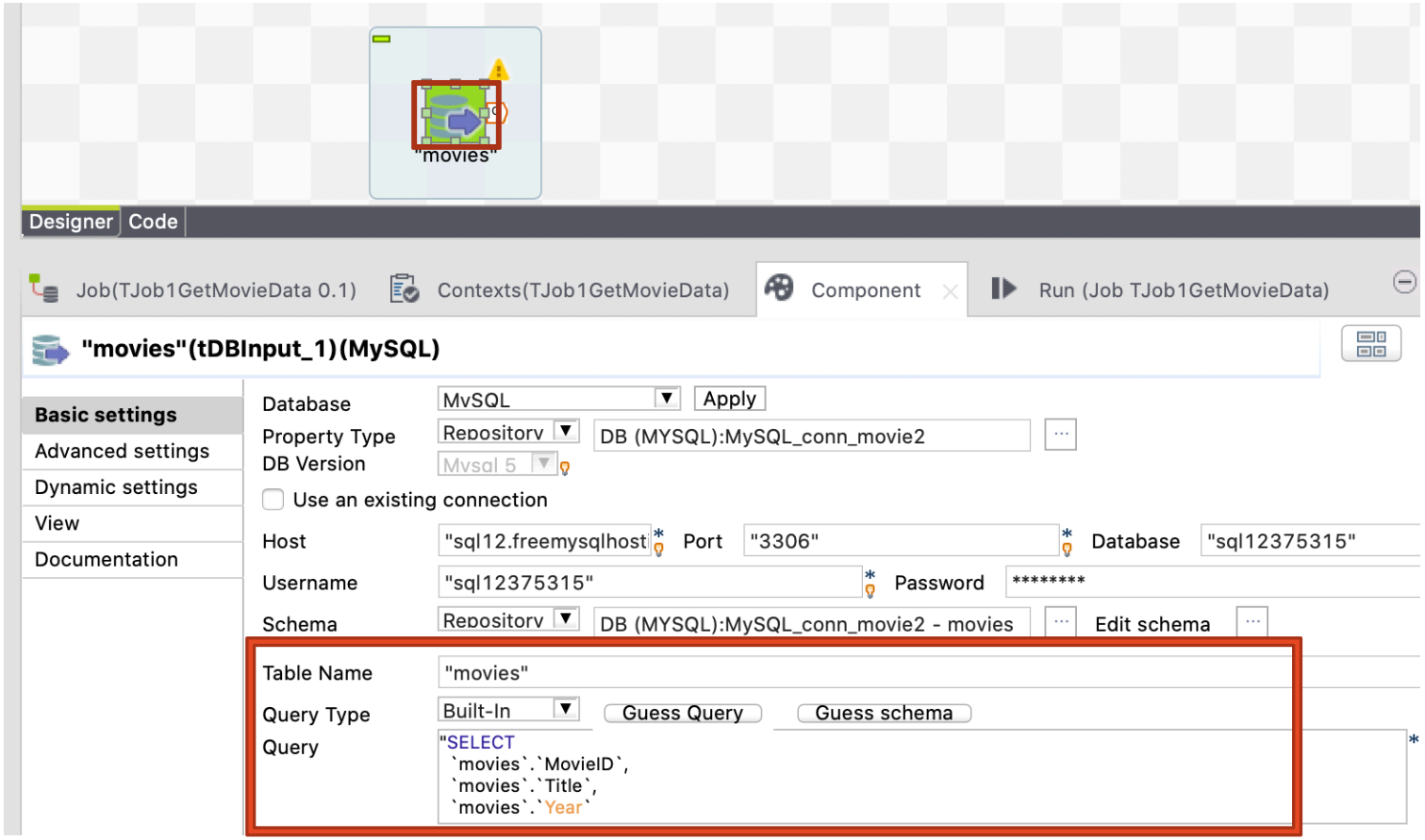
1.2 Create TJob#1: get movie data from RDB and convert data to CSV file (2)

- Drag Metadata of table "movies" under "MySQL_Conn_movie" to Design Workspace



Task1

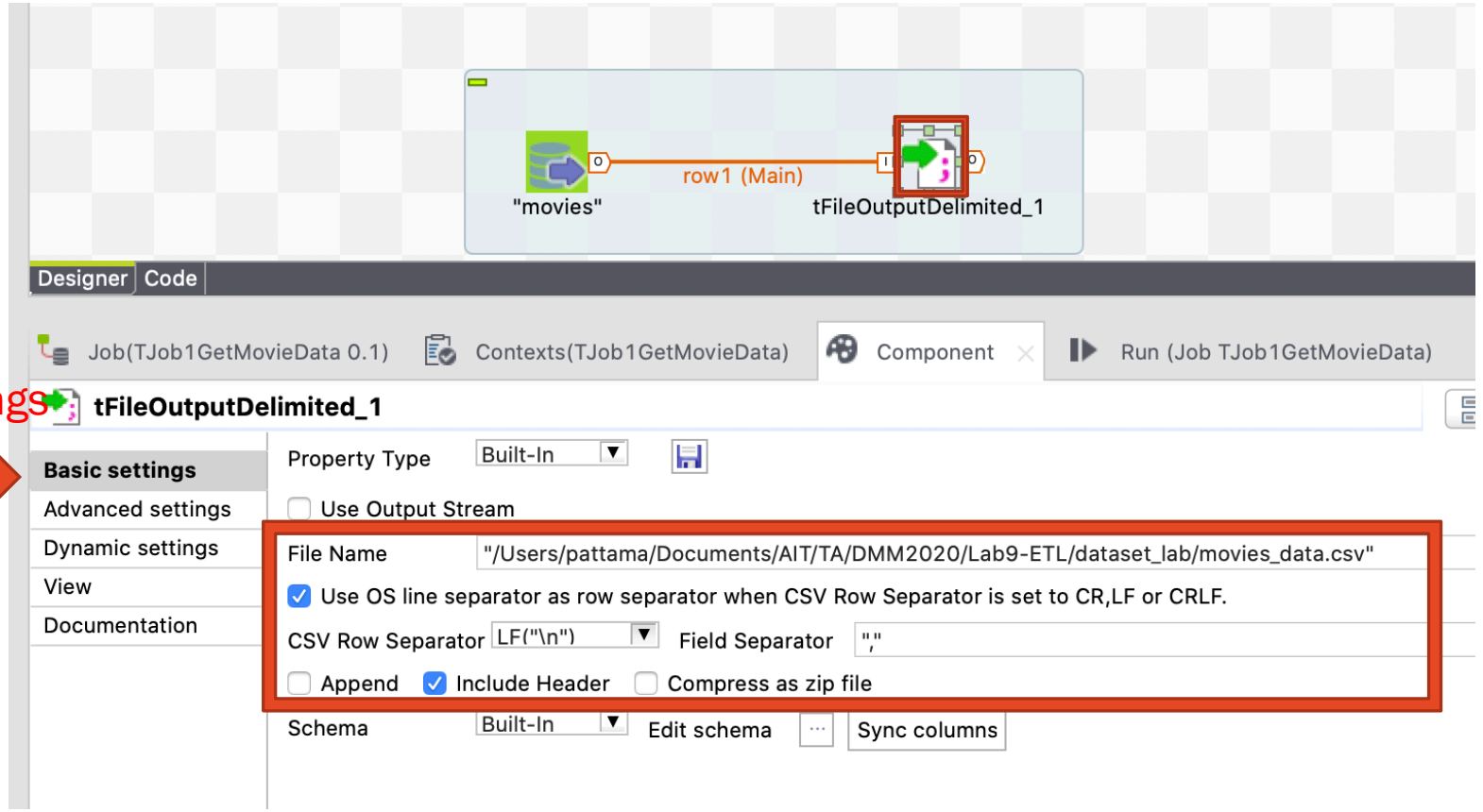
1.2 Create TJob#1: get movie data from RDB and convert data to CSV file (2)



Task1

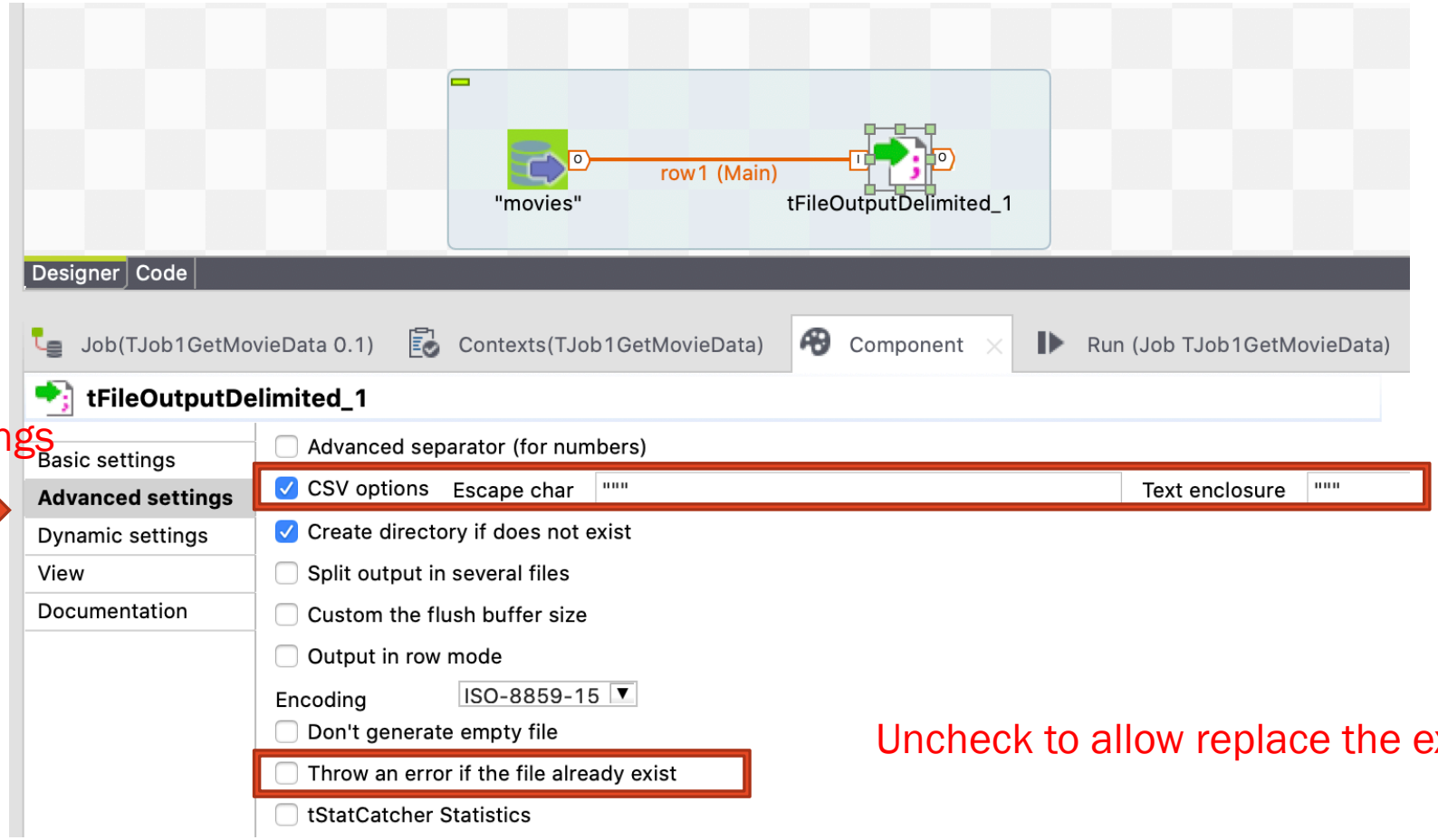
1.2 Create TJob#1: get movie data from RDB and convert data to CSV file (3)

Basic settings



Task1

1.2 Create TJob#1: get movie data from RDB and convert data to CSV file (4)

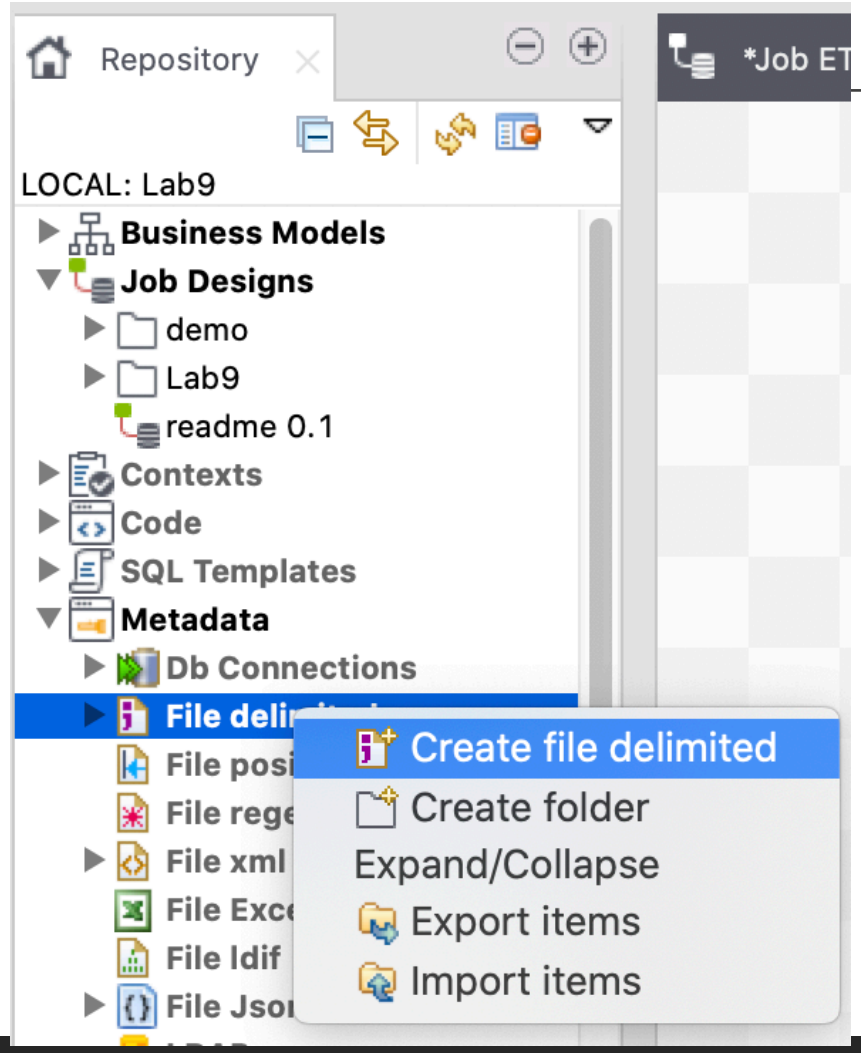


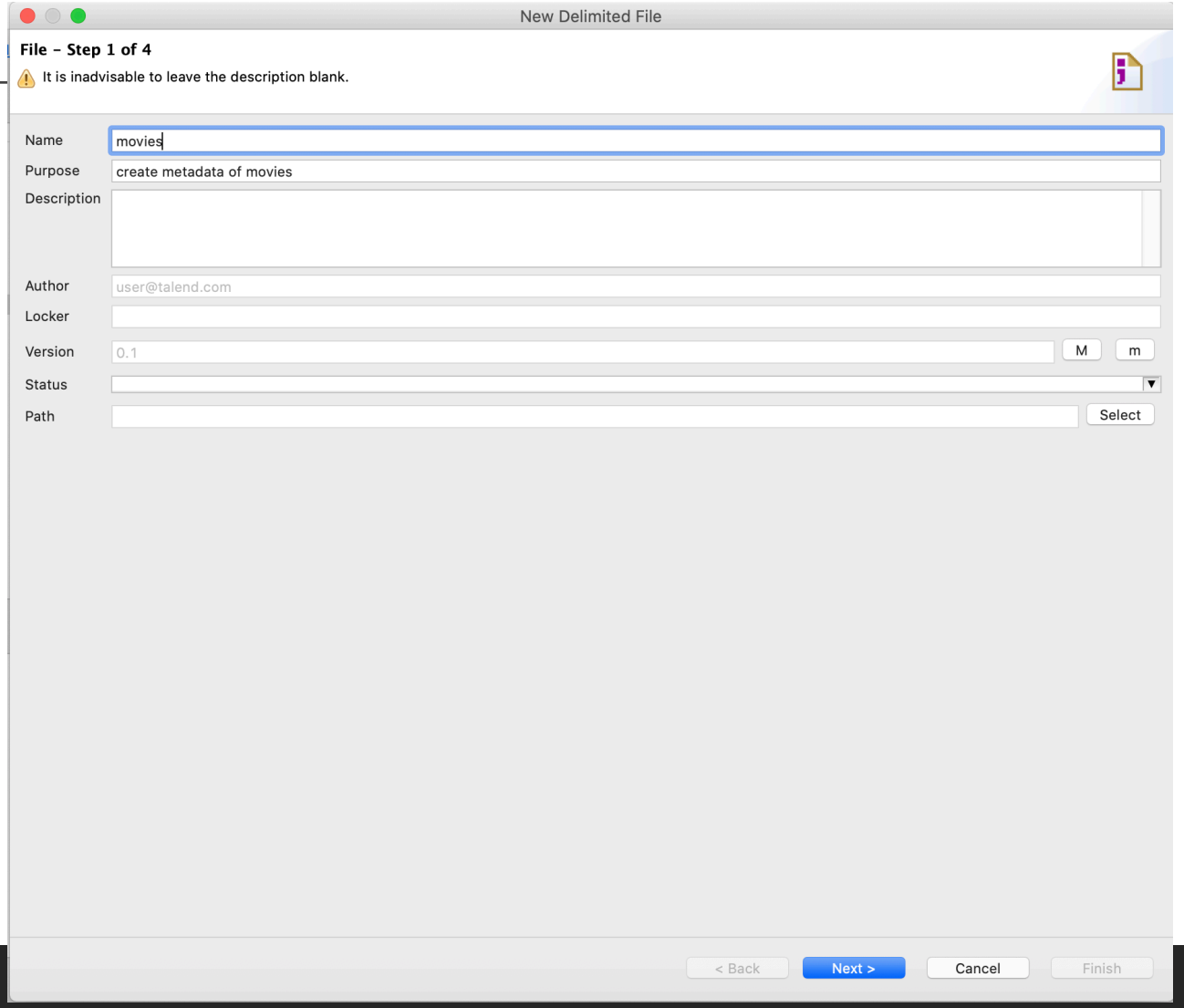
Advanced settings

Uncheck to allow replace the existing file

Task1

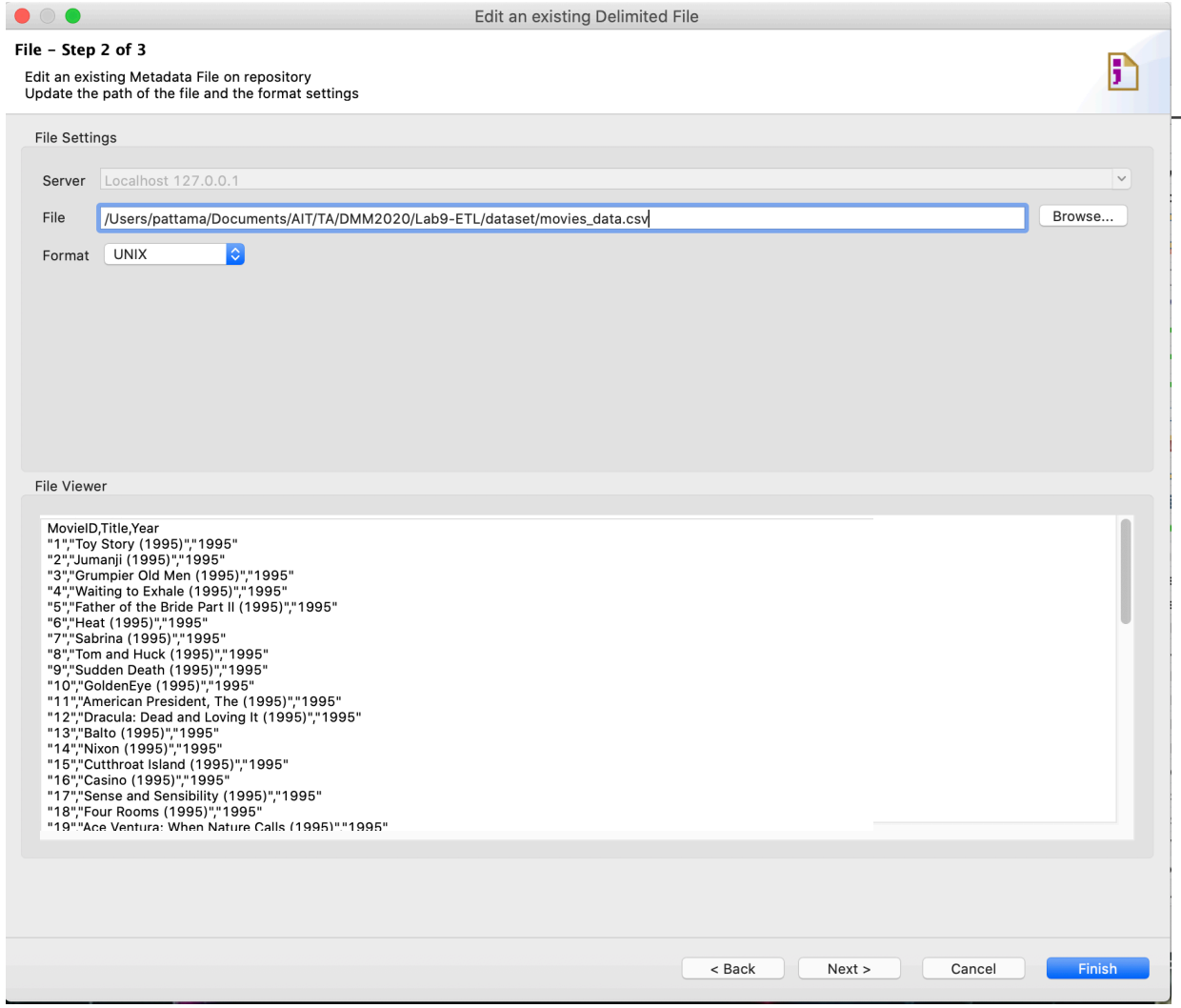
1.3 Create CSV Metadata for Movies (1)

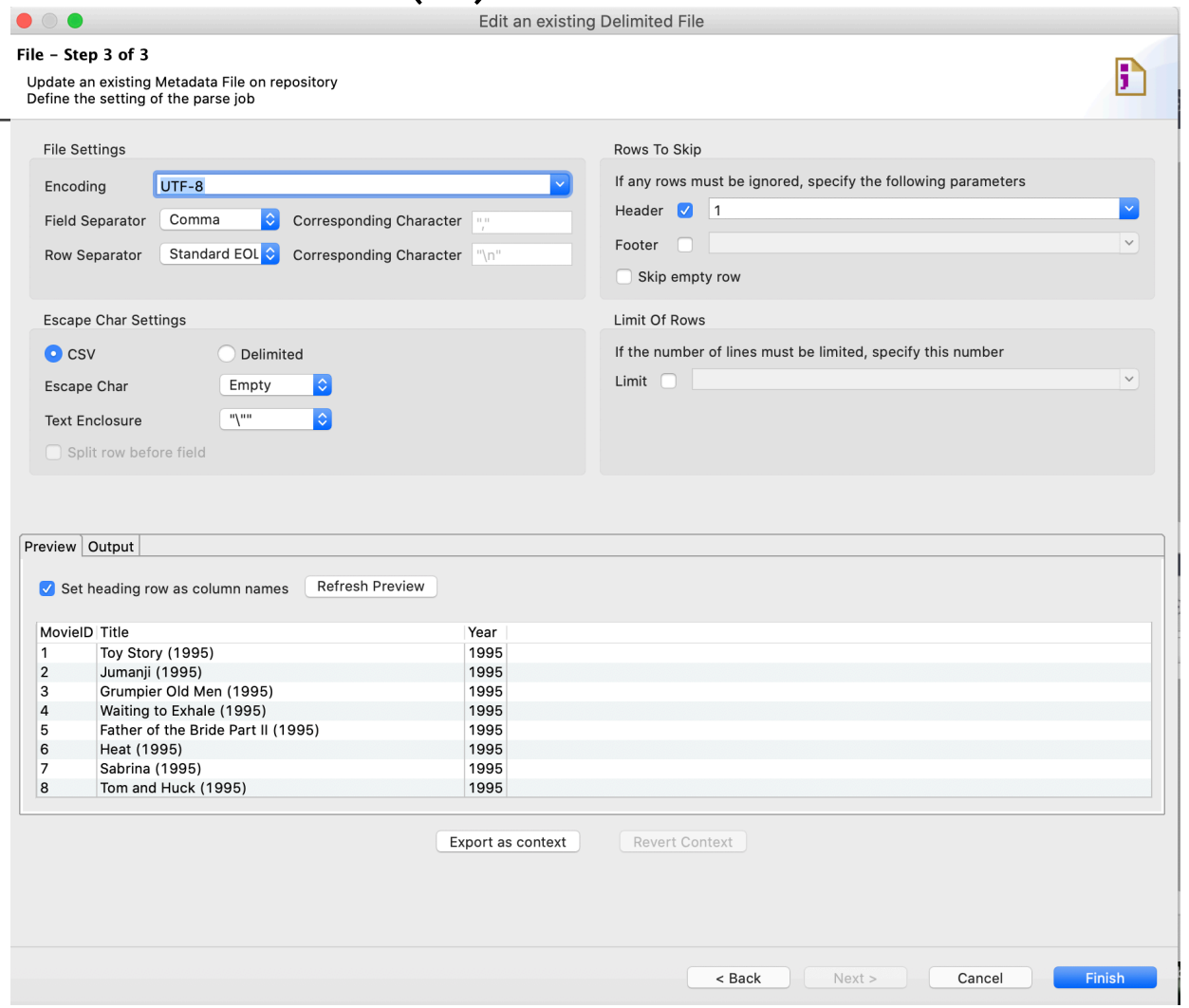




Task1

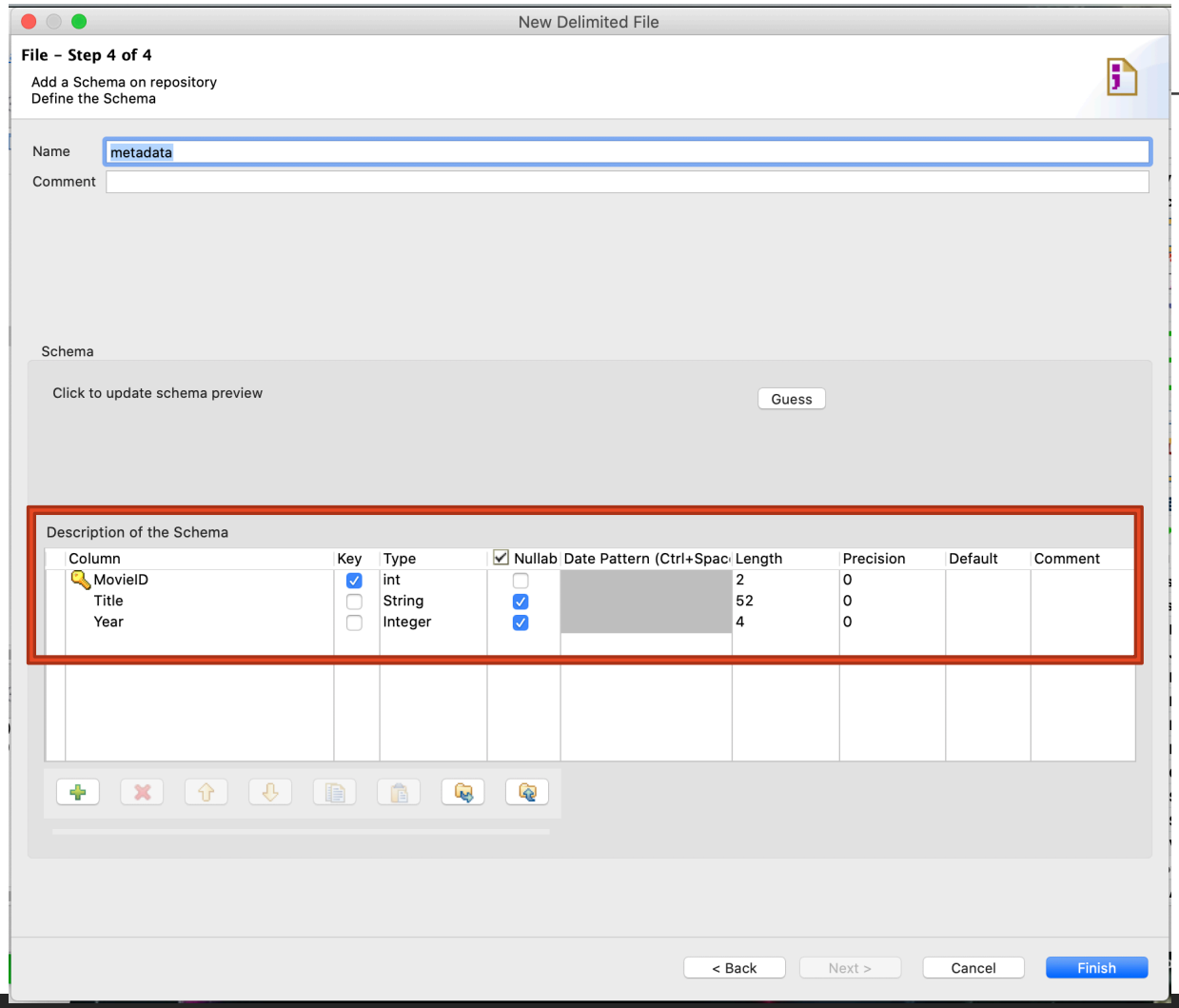
1.3 Create CSV Metadata for Movies (2)

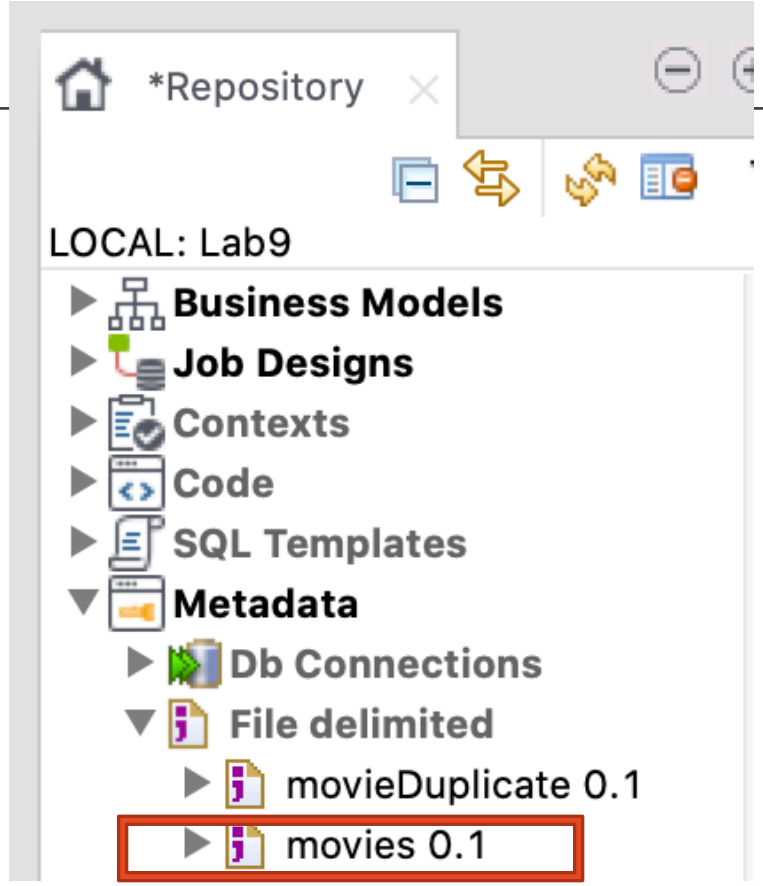




Task1

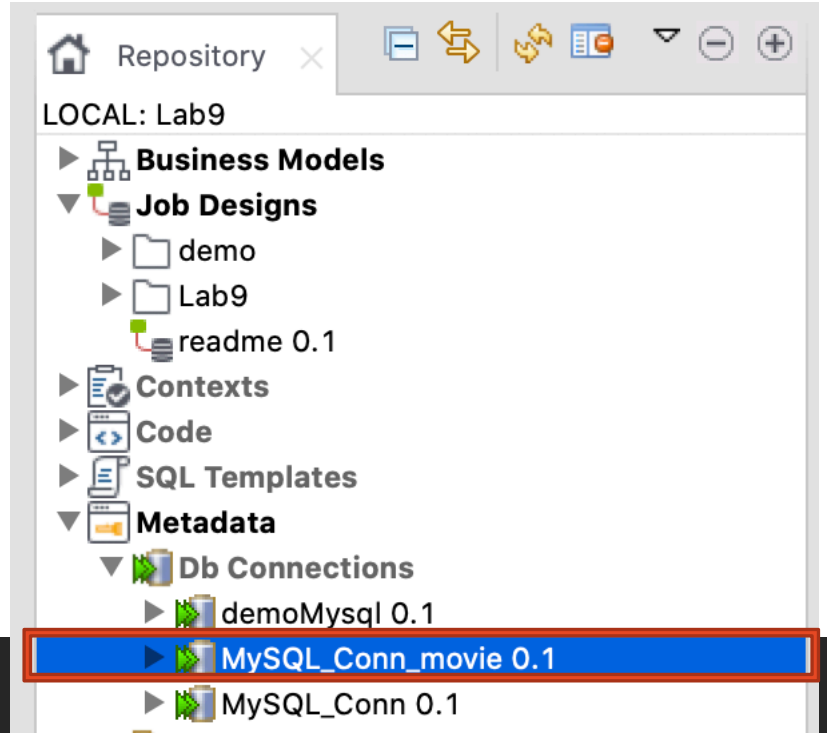
1.3 Create CSV Metadata for Movies (3)

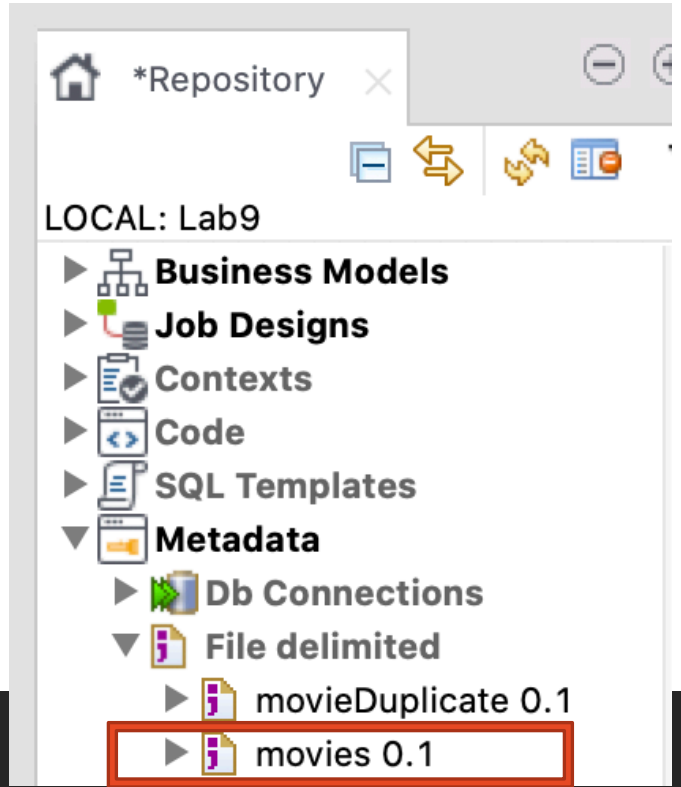




Output of Task1

1. Metadata of DB Connection “MySQL_Conn_movie”
2. Metadata of File delimited “movies”

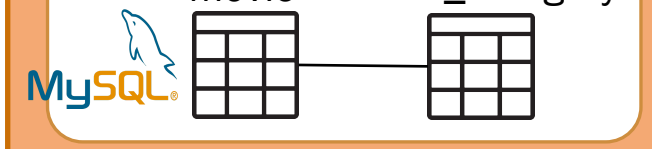




Movie Rating Prediction

Input

Movie



Relational Database

User Profile

CSV File

Movie Rating by Users

MongoDB

ETL Process

Extract

Step1: get movie data from RDB and convert data to CSV file
Step2: get user profile in CSV format
Step3: get movie rating by users from MongoDB
3.1 export data to JSON file
3.2 convert Json to CSV file

Transform

Step.4 Prepare data for desired datasets
4.1 Transform movie category data
4.2 Remove Duplicate Records
4.3 Integrate 3 datasets
4.4 Convert birthdate to Age
4.5 Convert Gender to numeric
4.6 Filter age >= 7 and Gender = 1|2
4.7 Merge output of 4.1 and 4.6

Load

Step.5 Write output CSV files

Talend

Task1 for step 1
- 1.1 Create Metadata for DB Connection
- 1.2 Create TJob#1
- 1.3 Create Metadata for movie

Task2 for step 2
- Create Metadata for user profile

Task3 for step 3
- 3.1 Create Metadata for movie rating (Json)
- 3.2 Create TJob#2
- 3.3 Create Metadata for movie rating (File Delimited)

Task4 for step 4.1
- 4.1 Create TJob#3
- 4.2 Create Metadata for movie category

Task5 for step 4.2-5
- Create TJob#4

User Profile

CSV File

Task2 for step 2

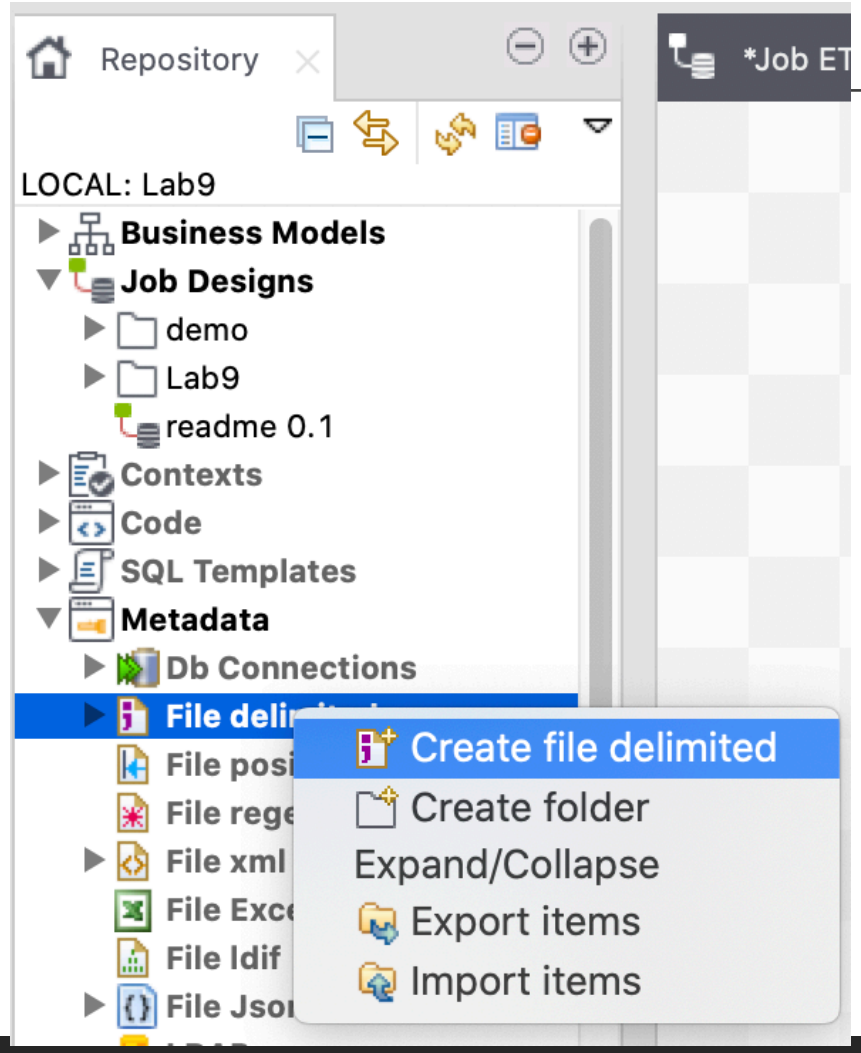
Step2: get user profile in CSV format

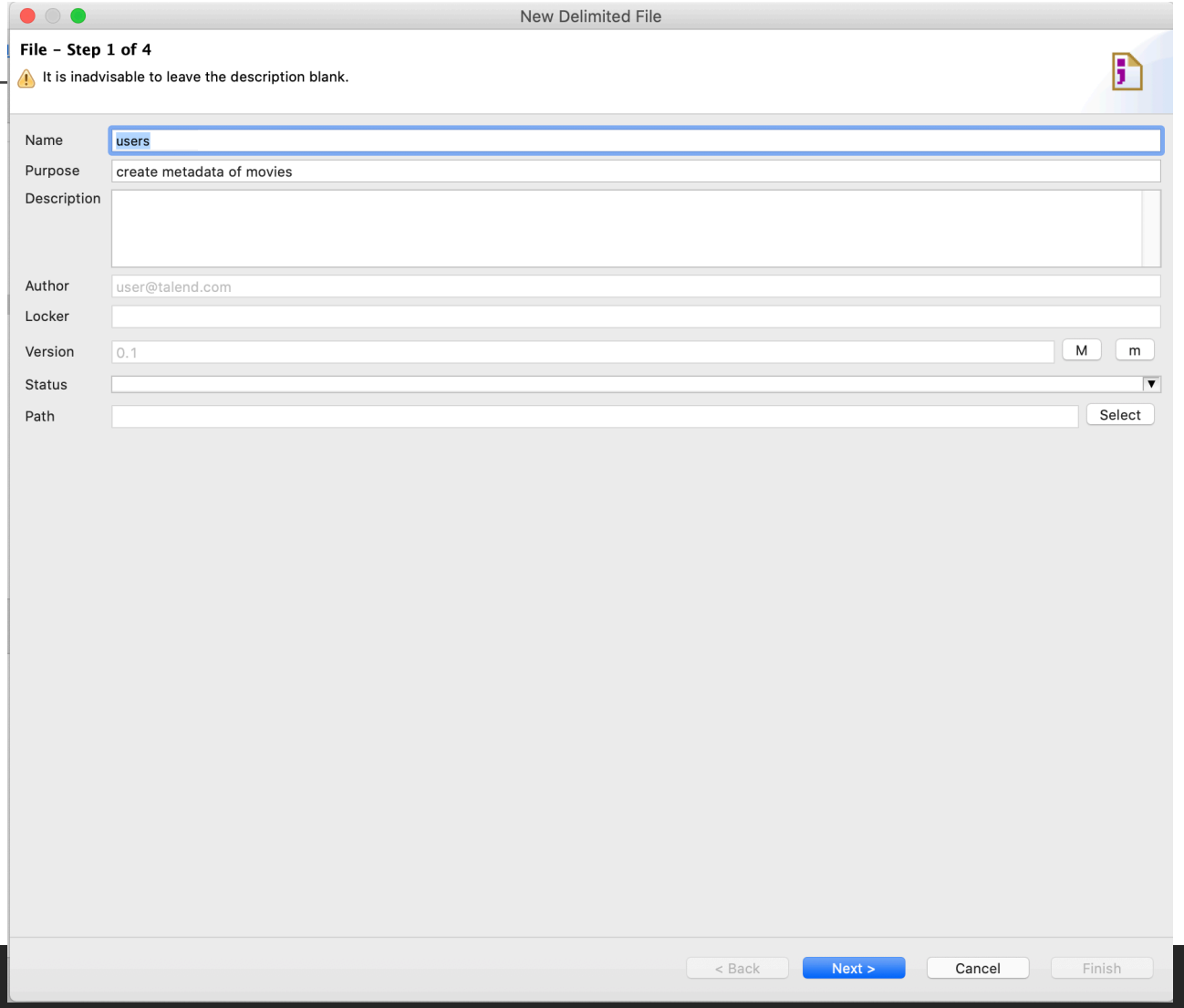
Task2:

Create Metadata for user profile (File Delimited)

Task2

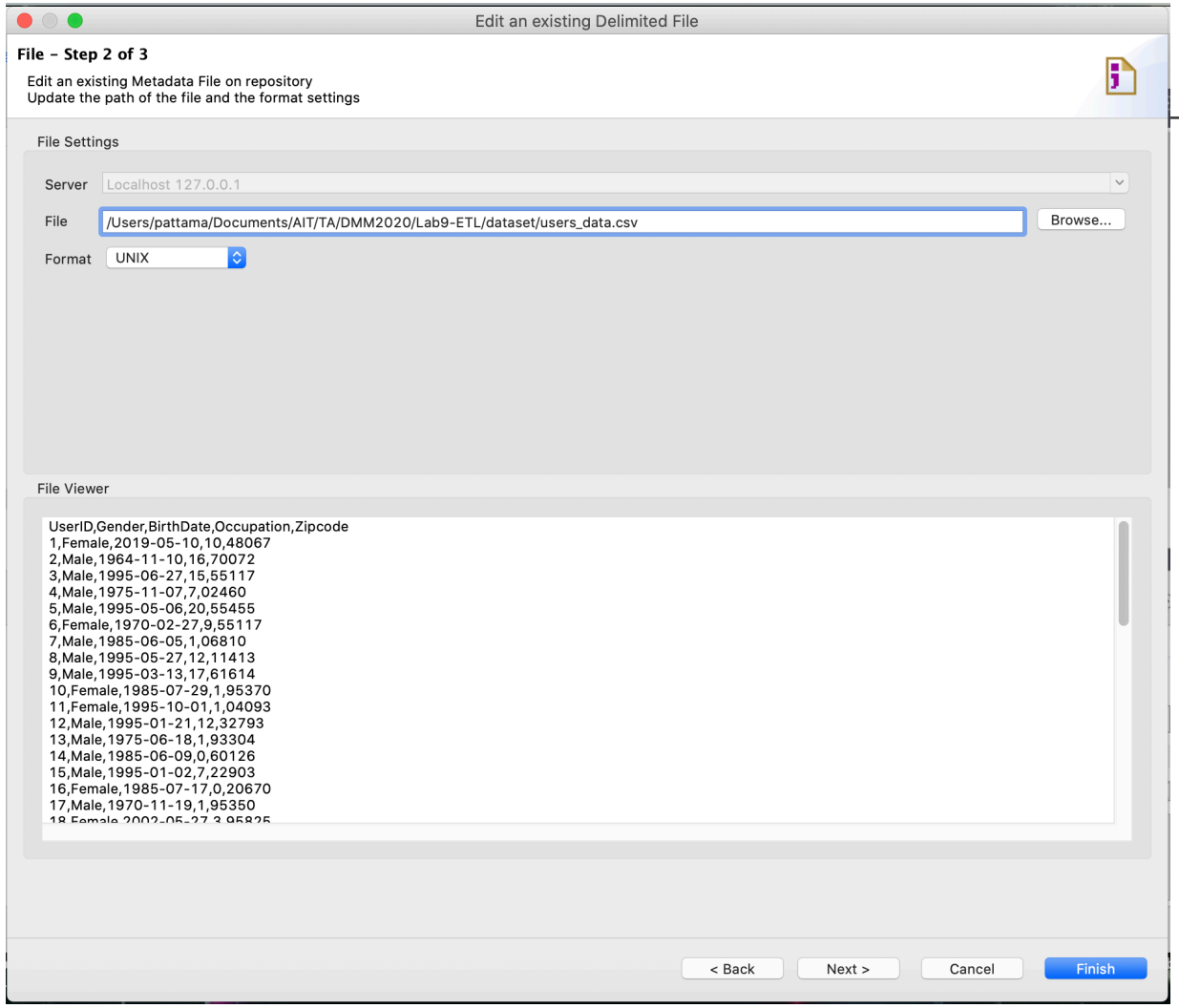
Create CSV Metadata for User Profile (1)

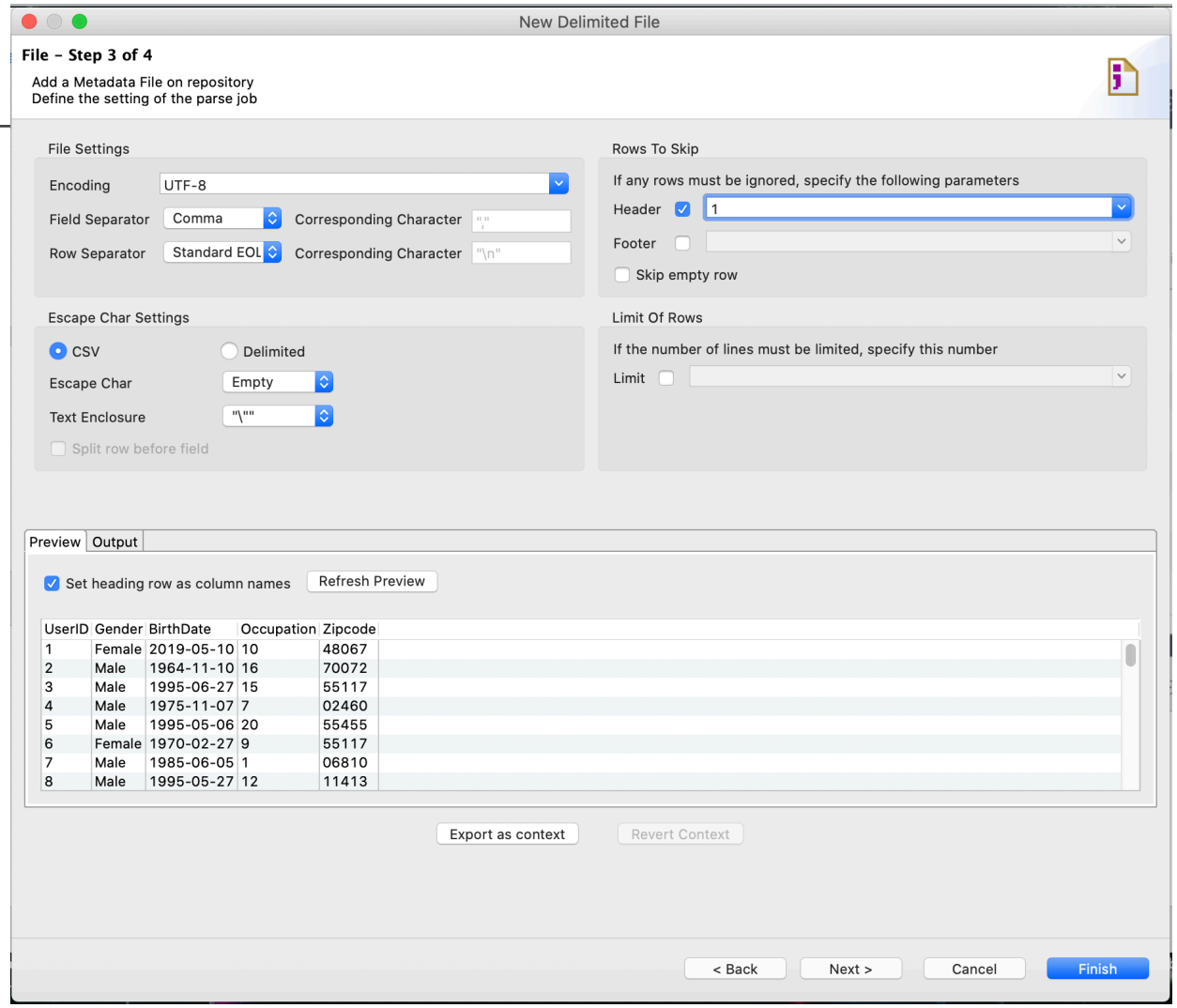




Task2

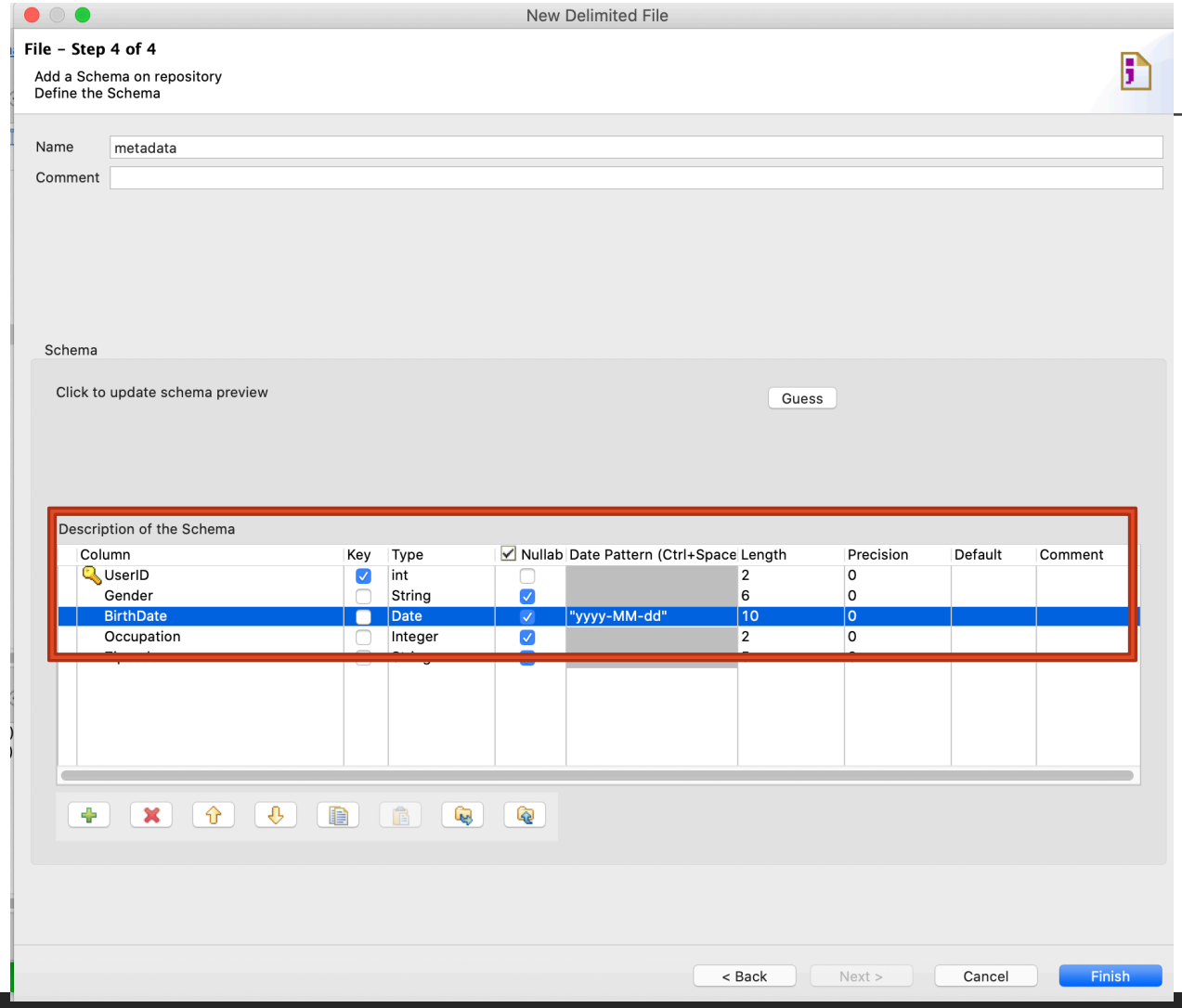
Create CSV Metadata for User Profile (2)

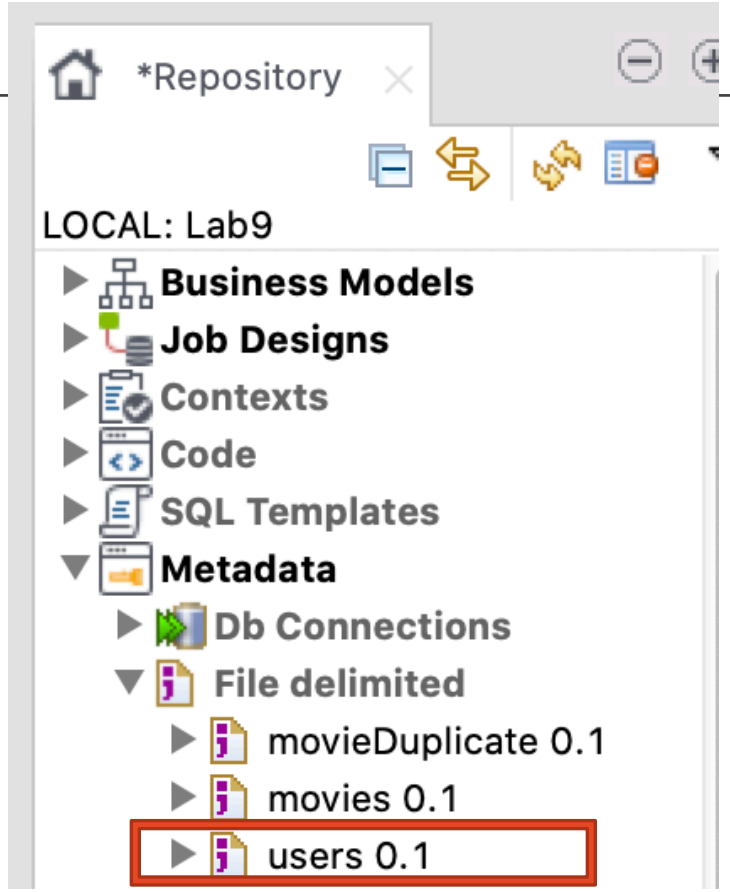




Task2

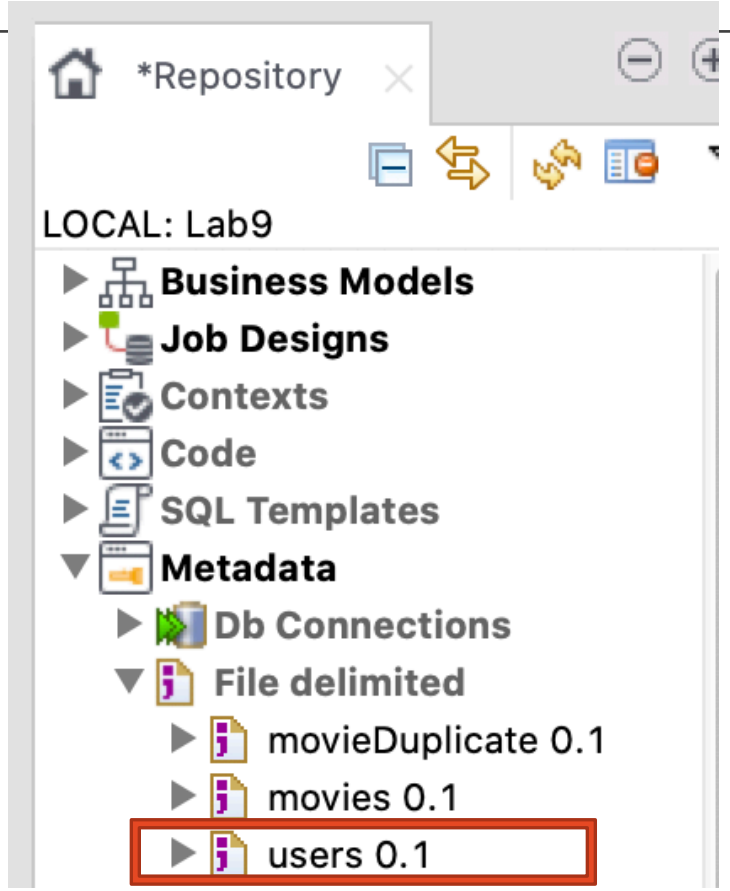
Create CSV Metadata for User Profile (3)





Output of Task2

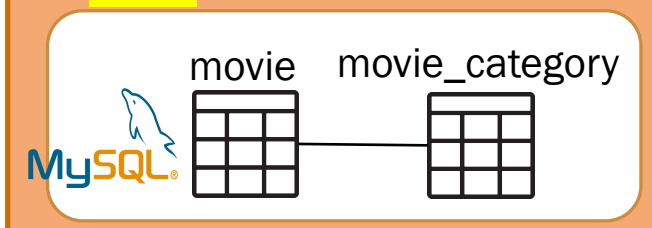
1. Metadata of File delimited “users”



Movie Rating Prediction

Input

Movie



Relational Database

User Profile

CSV File

Movie Rating by Users

MongoDB

ETL Process

Extract

Step1: get movie data from RDB and convert data to CSV file
Step2: get user profile in CSV format
Step3: get movie rating by users from MongoDB
3.1 export data to JSON file
3.2 convert Json to CSV file

Transform

Step.4 Prepare data for desired datasets
4.1 Transform movie category data
4.2 Remove Duplicate Records
4.3 Integrate 3 datasets
4.4 Convert birthdate to Age
4.5 Convert Gender to numeric
4.6 Filter age >= 7 and Gender = 1|2
4.7 Merge output of 4.1 and 4.6

Load

Step.5 Write output CSV files

Talend

Task1 for step 1
- 1.1 Create Metadata for DB Connection
- 1.2 Create TJob#1
- 1.3 Create Metadata for movie

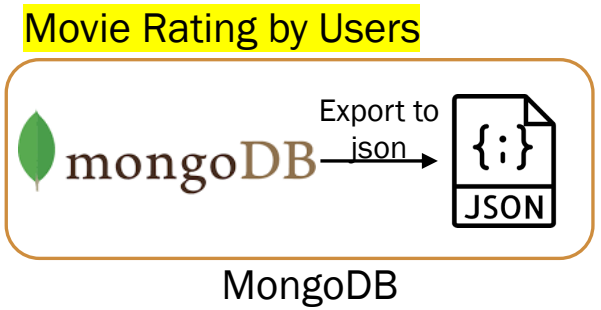
Task2 for step 2
- Create Metadata for user profile

Task3 for step 3
- 3.1 Create Metadata for movie rating (Json)
- 3.2 Create TJob#2
- 3.3 Create Metadata for movie rating (File Delimited)

Task4 for step 4.1
- 4.1 Create TJob#3
- 4.2 Create Metadata for movie category

Task5 for step 4.2-5
- Create TJob#4

Task3 for step 3



Step3: get movie rating by users from MongoDB

Task3:

- Export data from MongoDB to JSON file

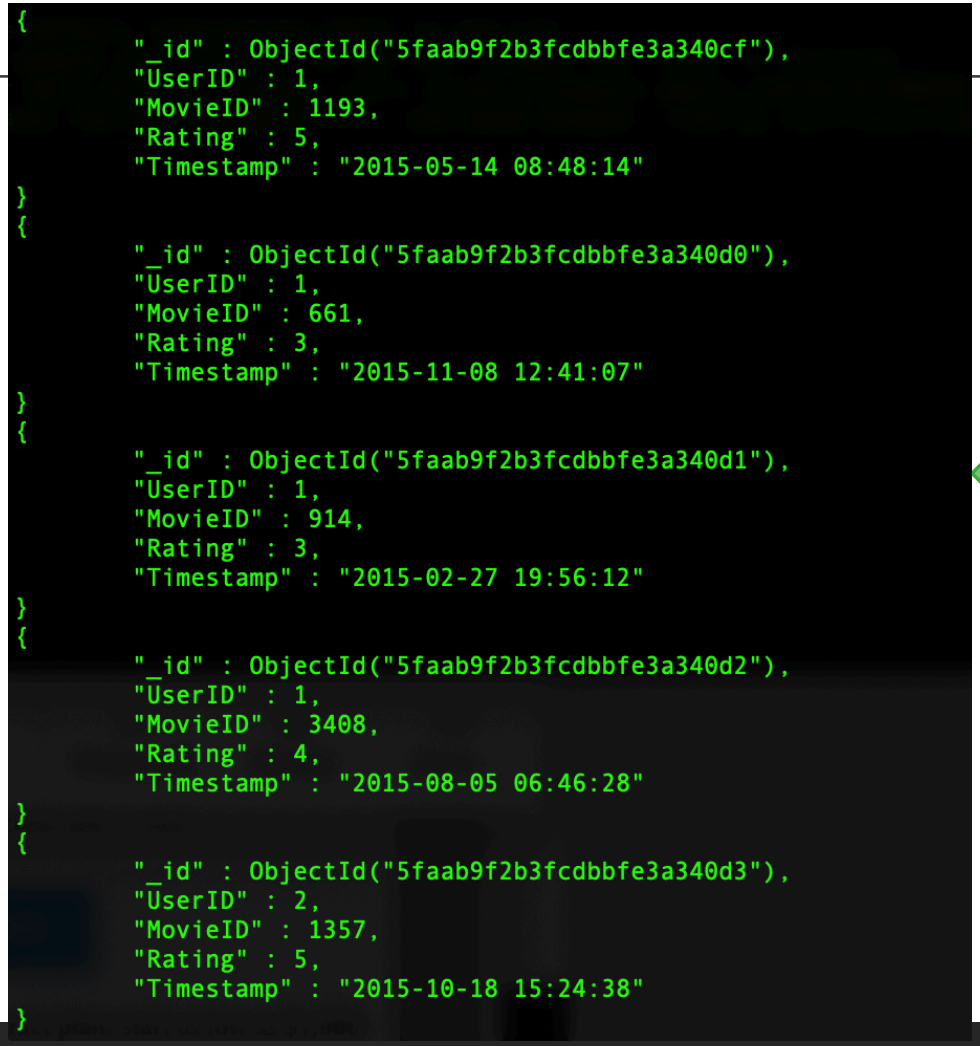
Atlas MongoDB

- Convert Json to CSV file
  - 3.1 Create Metadata for movie rating (Json)
  - 3.2 Create TJob#2
  - 3.3 Create Metadata for movie rating (File Delimited)

Talend

Task3

Export data from MongoDB to JSON file



Atlas MongoDB

movie rating collection in MongoDB

Task3

Export data from MongoDB to JSON file

- Export Data from MongoDB to JSONArray via Terminal

Example:

```
mongoexport --uri="mongodb://localhost:27017/dmm2020" --collection=ratings --jsonArray
 --out=/Users/pattama/Documents/AIT/TA/DMM2020/Lab9-ETL/datasets/ratings.json
```

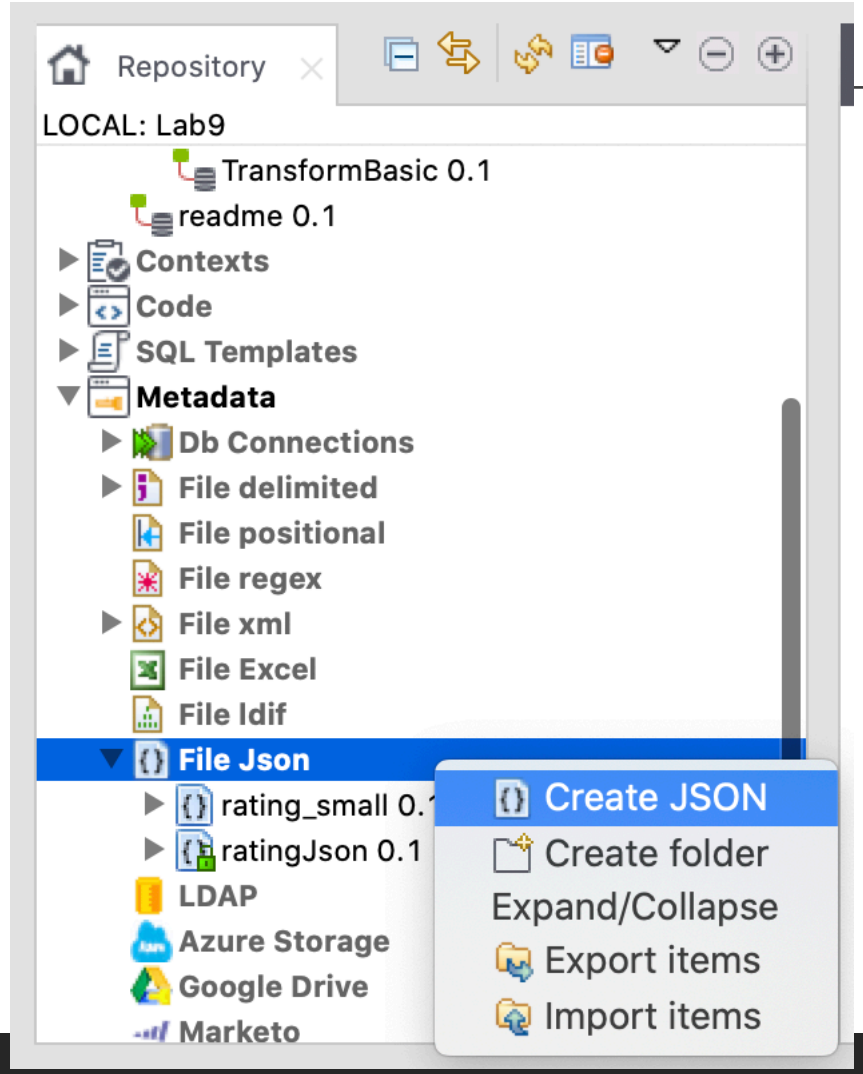
- Export Data from MongoDB to JSONArray via MongoCompass

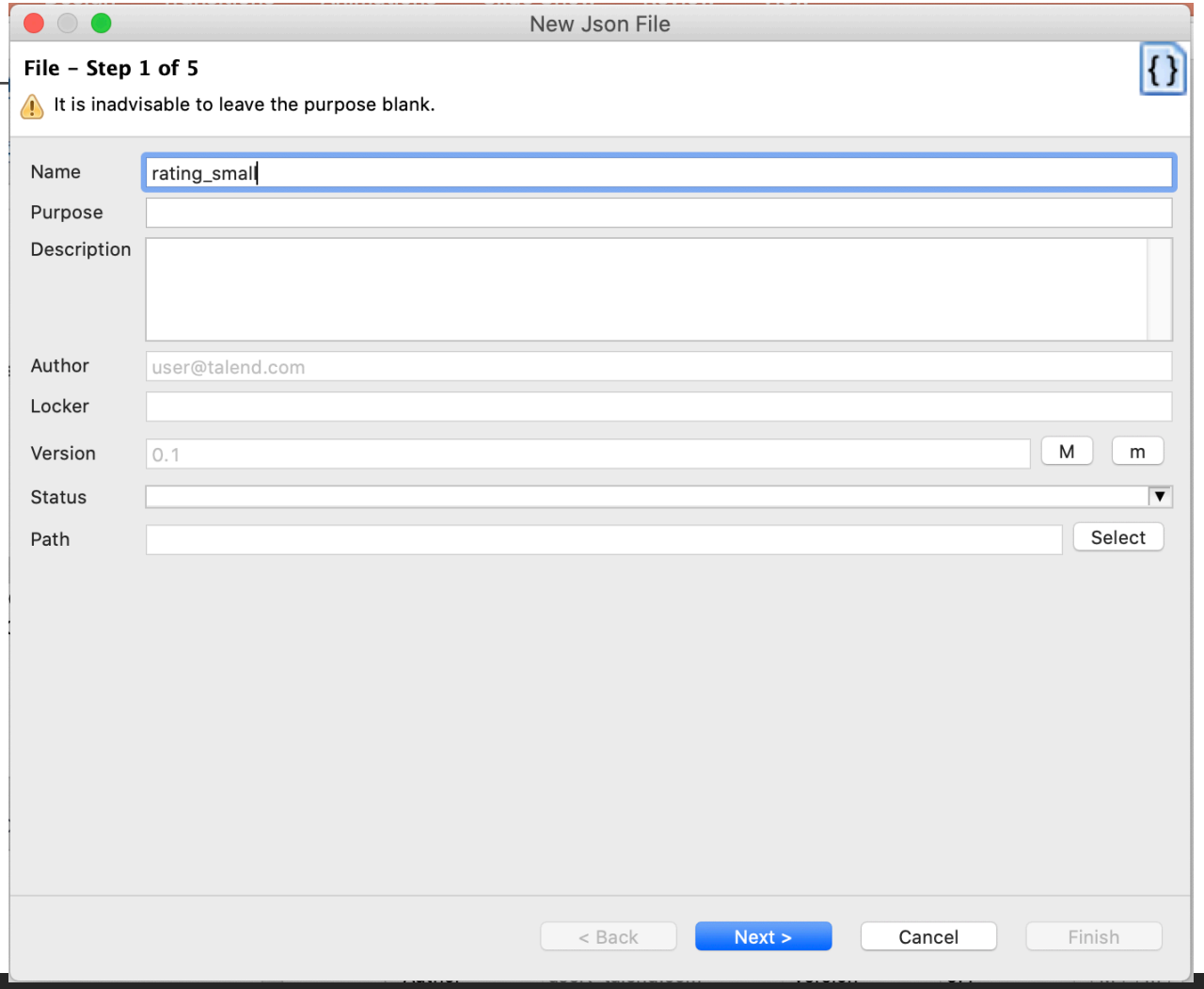
URL: https://youtu.be/L6usQ1KbSRs

Source: https://docs.mongodb.com/manual/reference/program/mongoexport/

Task3

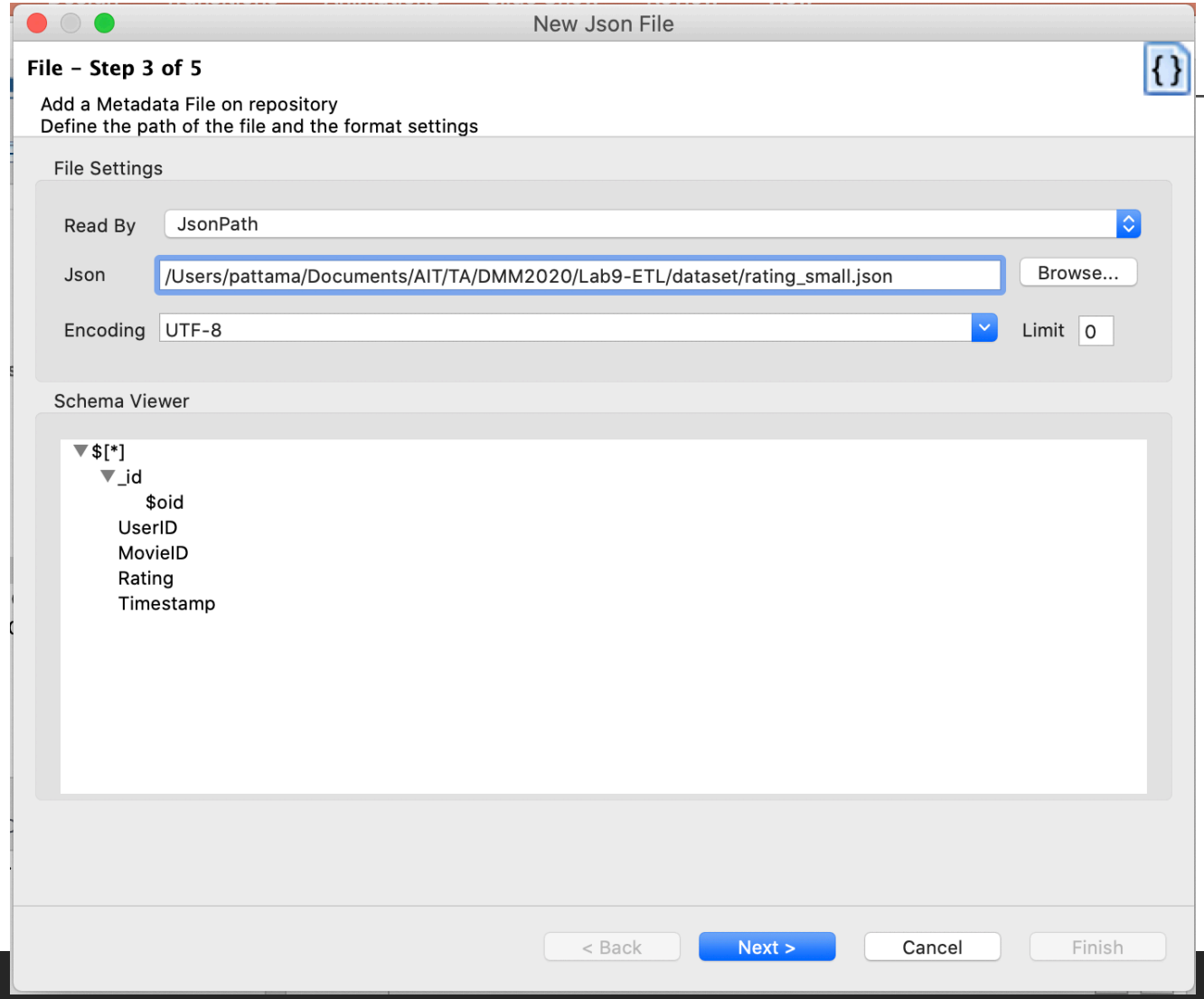
3.1 Create Metadata for movie rating (Json) (1)

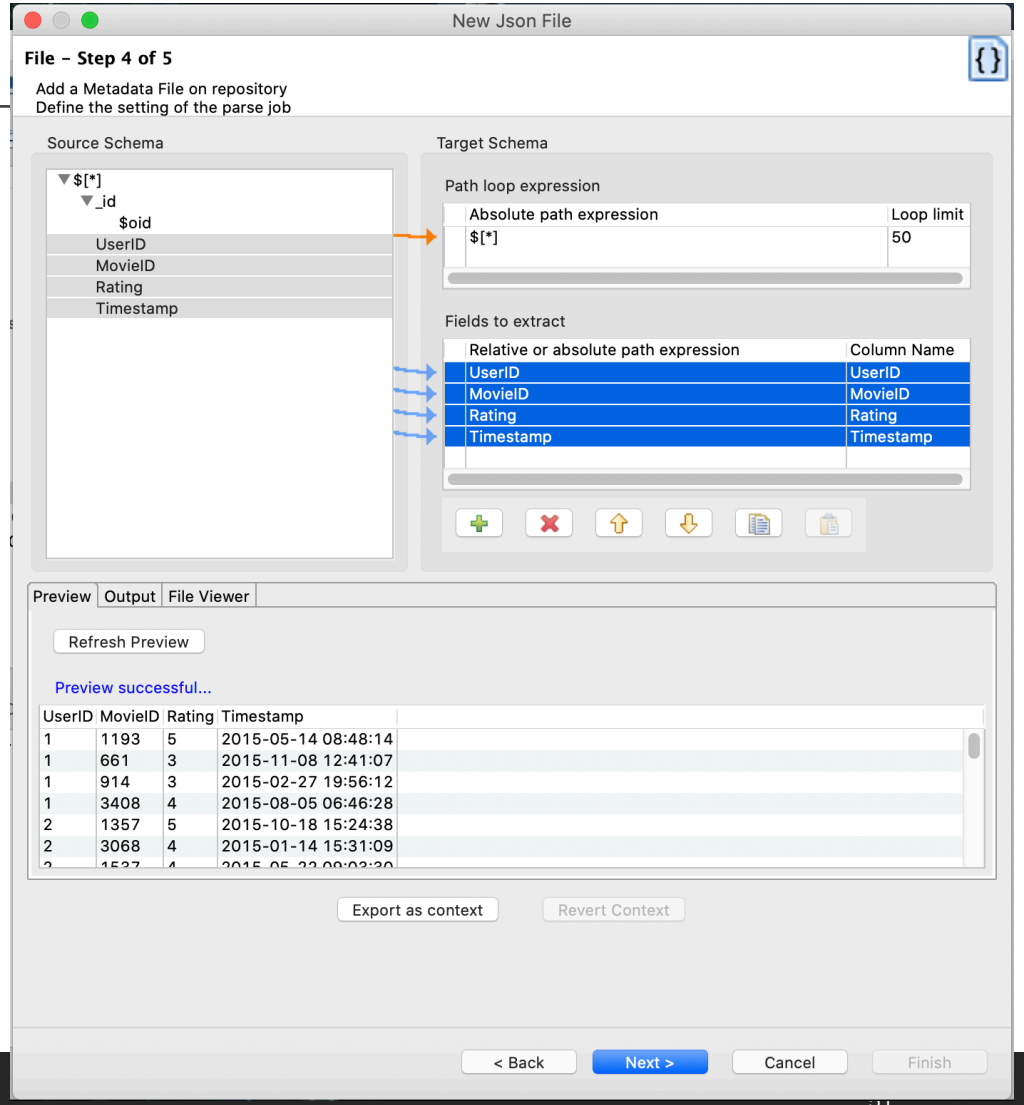




Task3

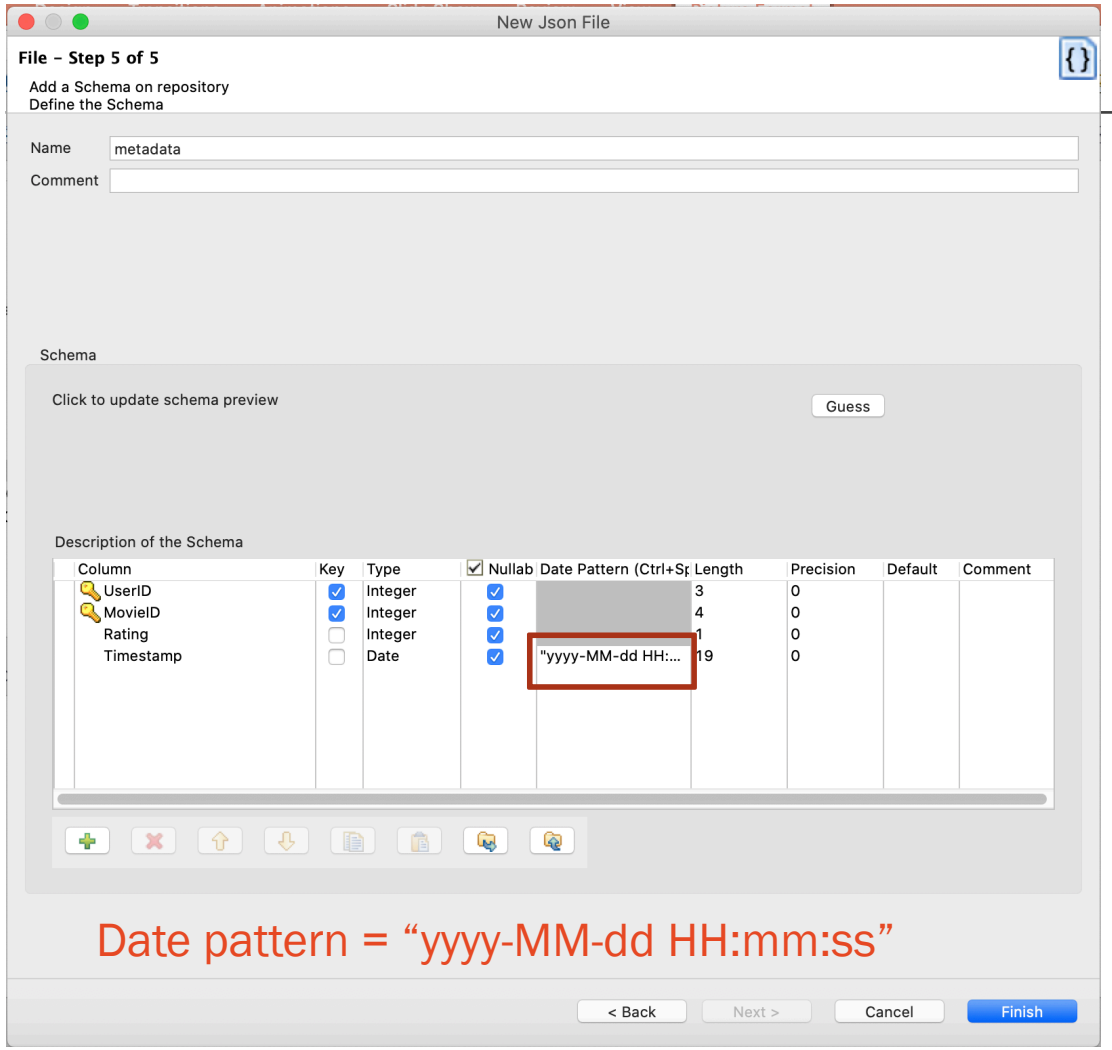
3.1 Create Metadata for movie rating (Json)(2)

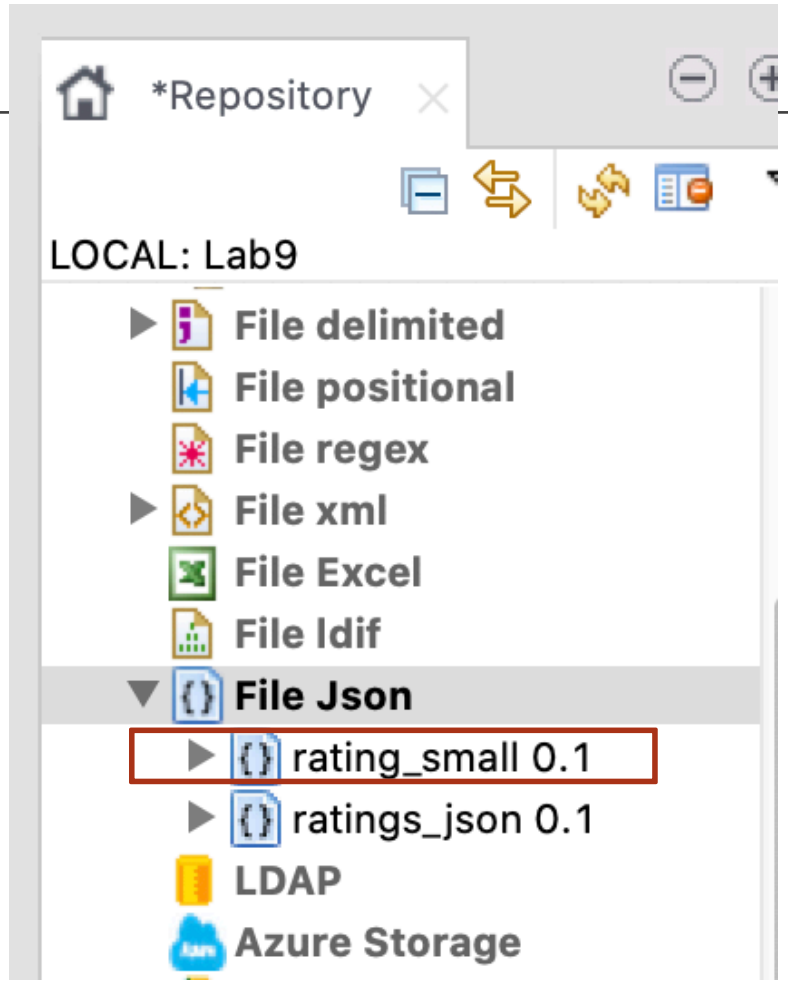




Task3

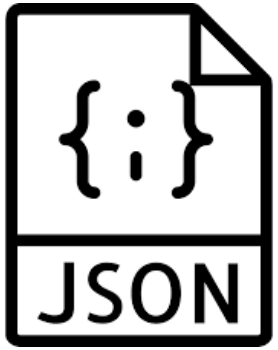
3.1 Create Metadata for movie rating (Json)(3)





Task3

3.2 Create TJob#2: Convert Json to CSV File



Json Metadata (tInputFileJSON)

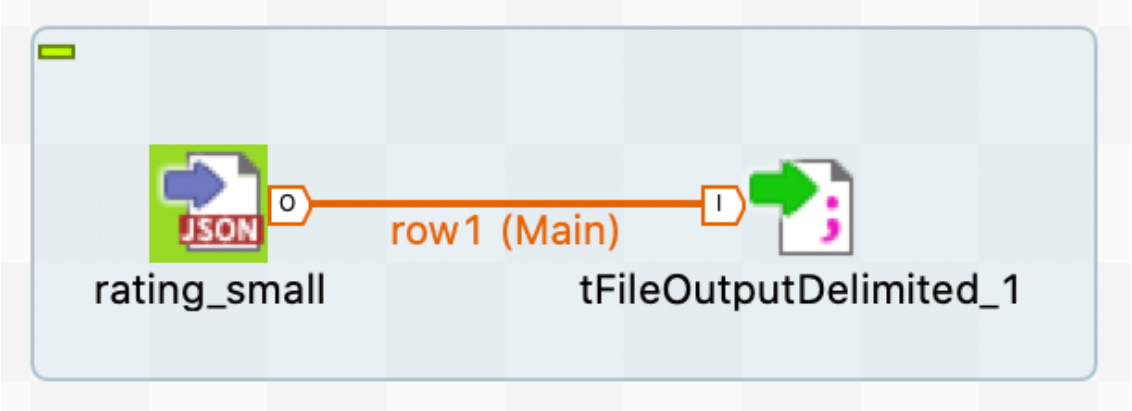
ratings.csv

Task3

3.2 Create TJob#2: Convert Json to CSV File

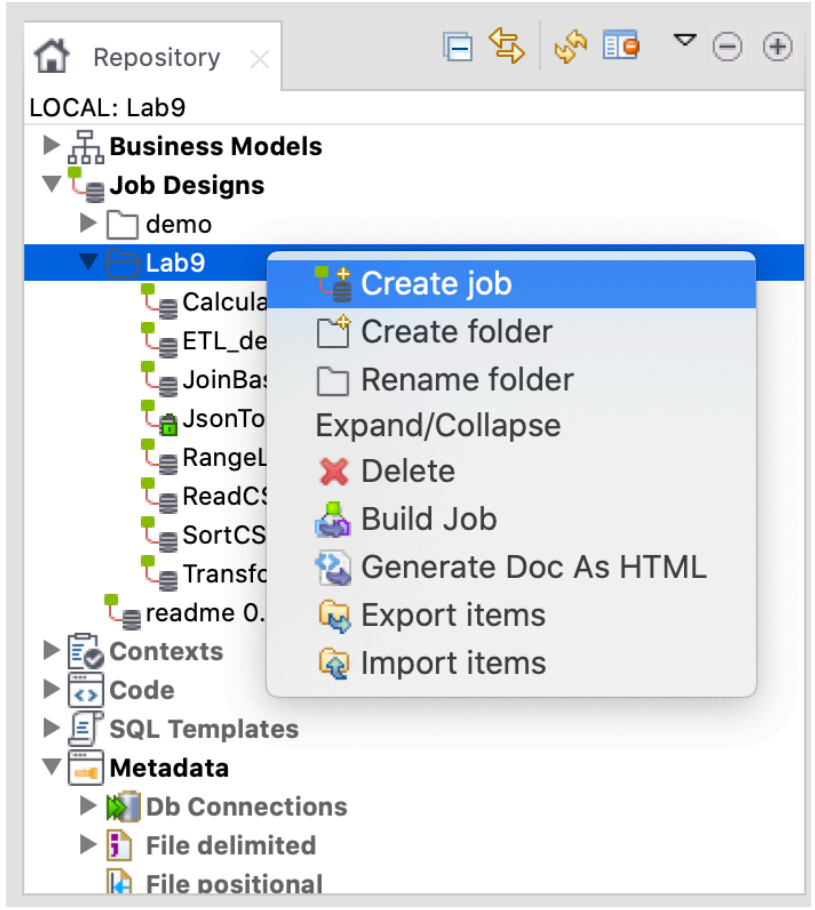
Purpose : Creating a Job to extract json data and convert to CSV file.

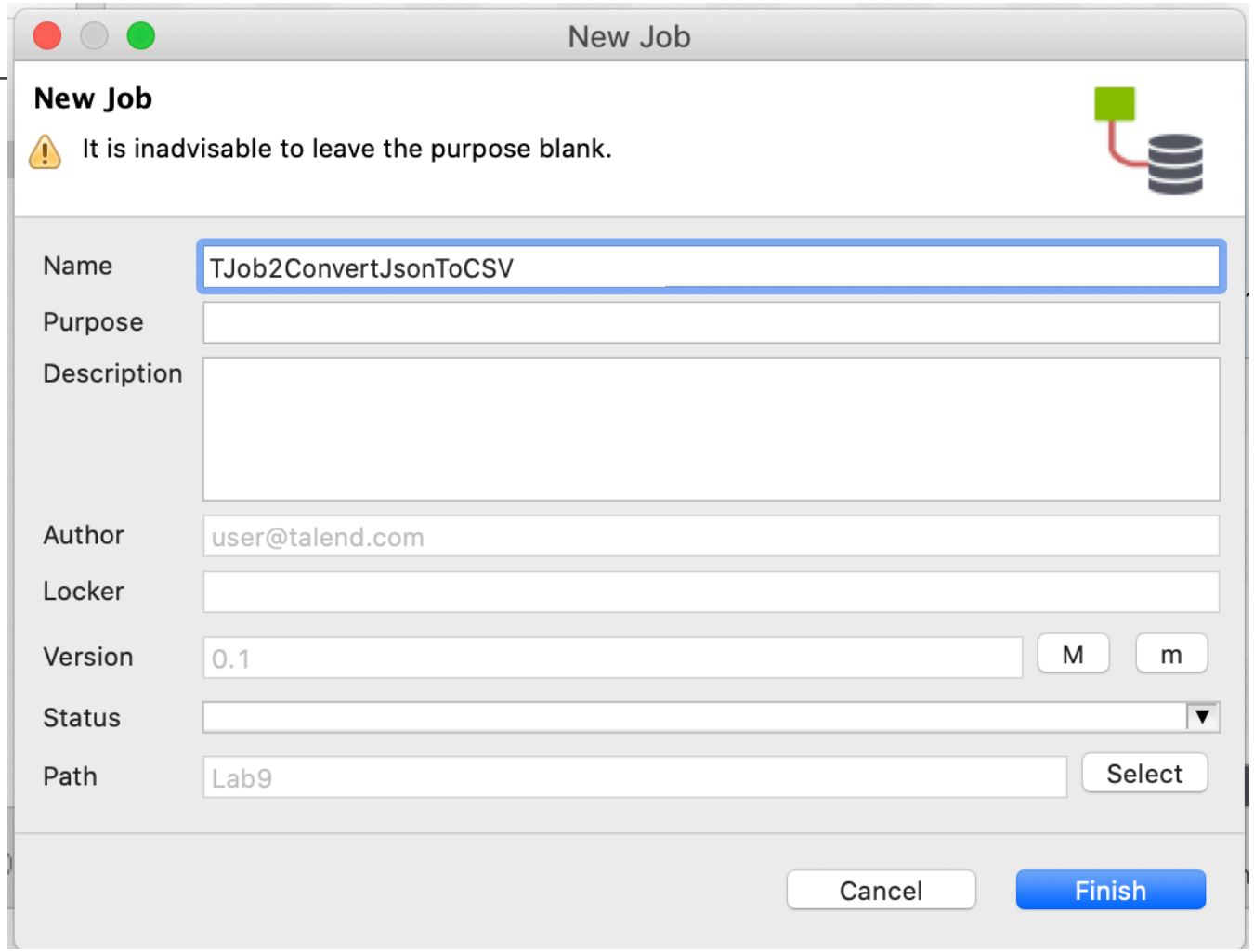
Components: tFileInputJson , tFileOutputDelimited



Task3

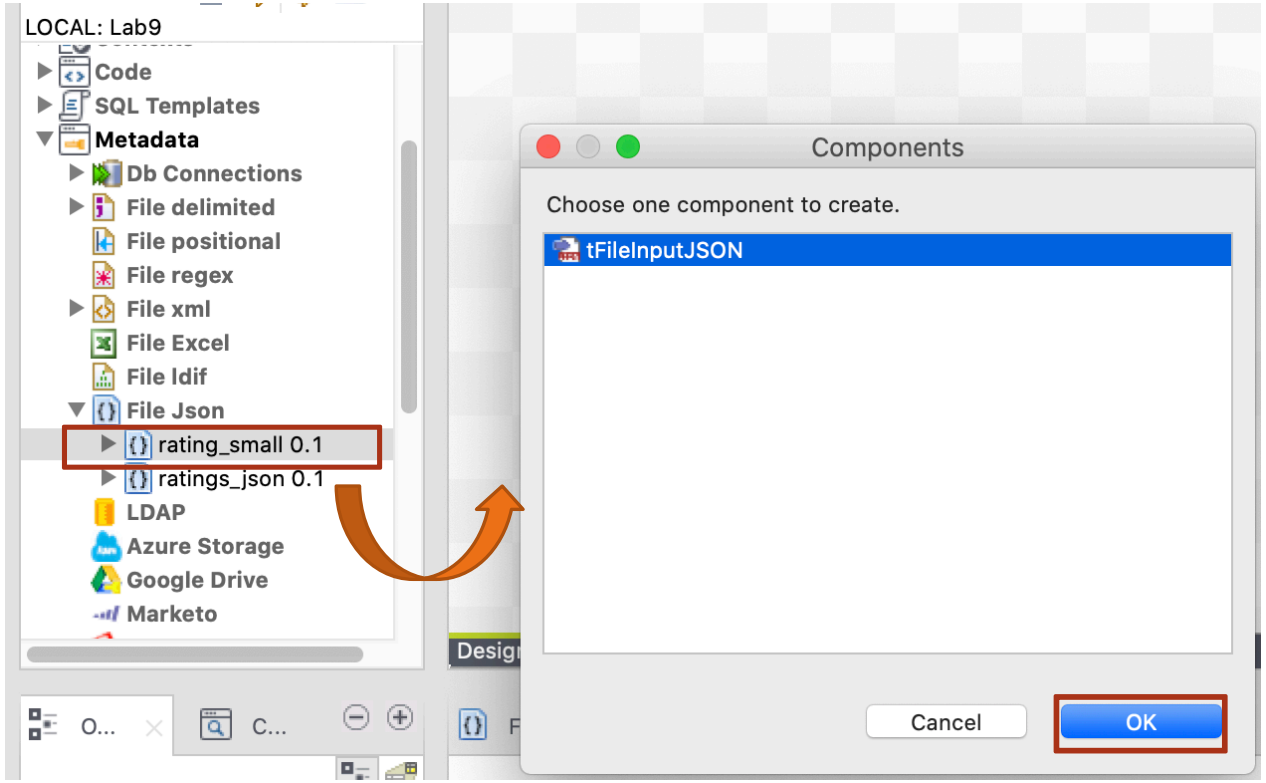
3.2 Create TJob#2: Convert Json to CSV File (1)

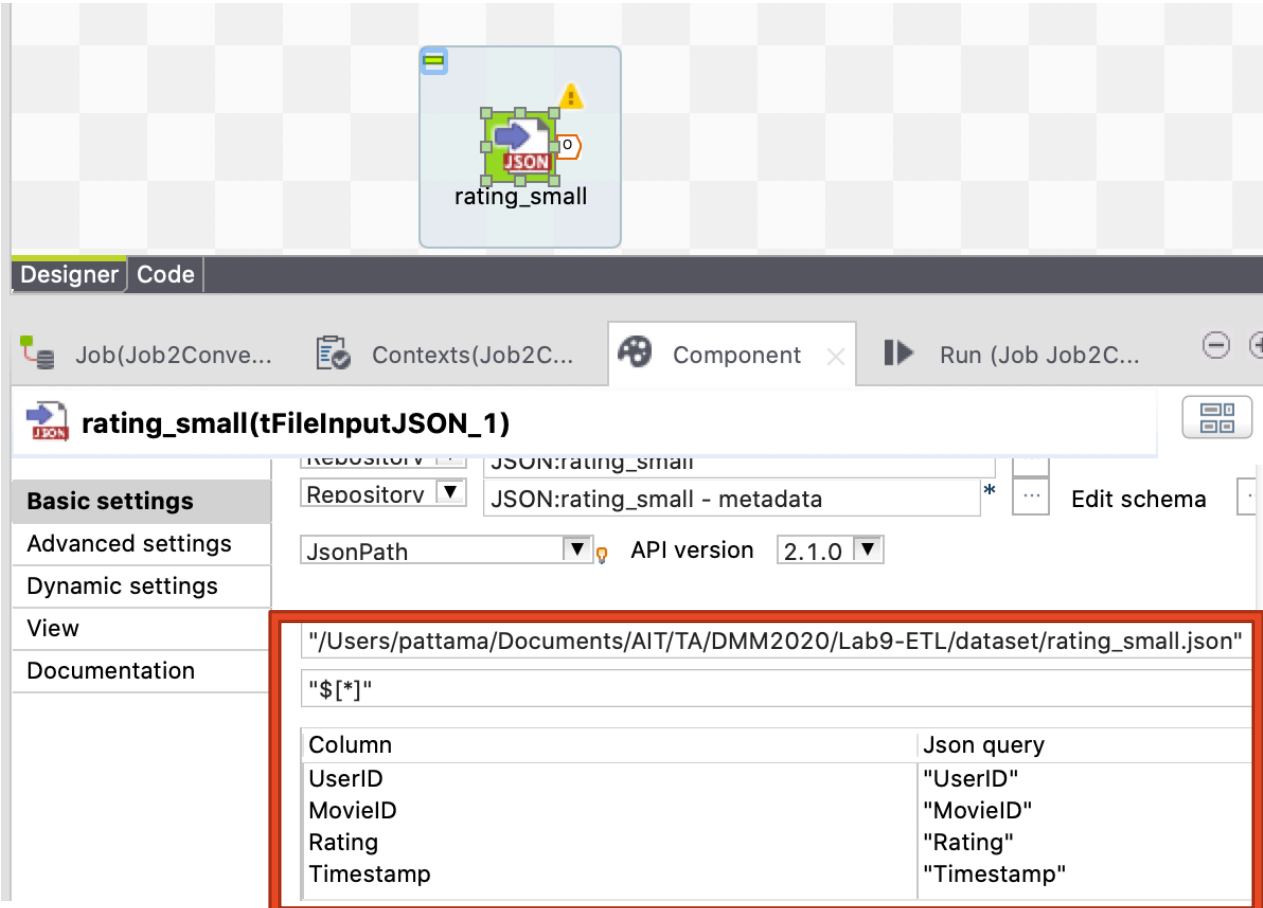




Task3

3.2 Create TJob#2: Convert Json to CSV File (2)

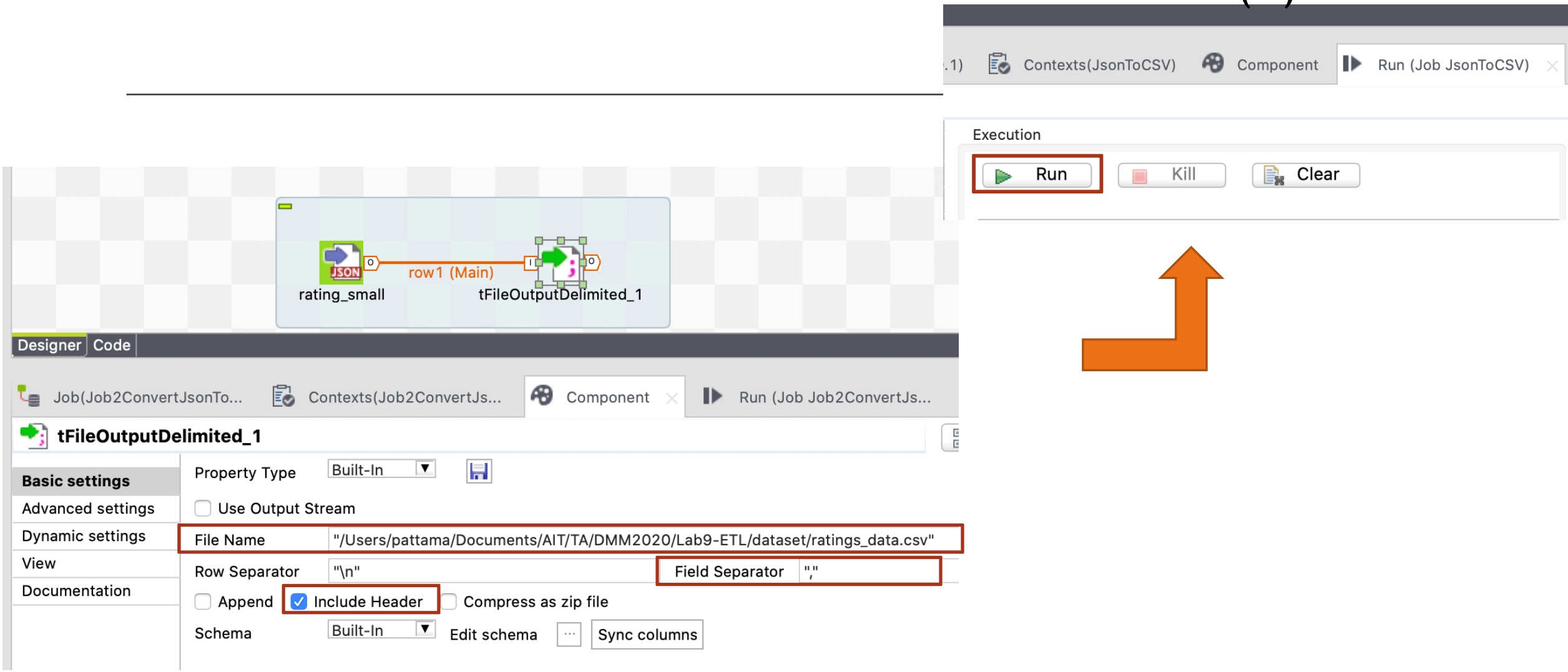




Drag ratings_json to “design workspace” area and click “OK”

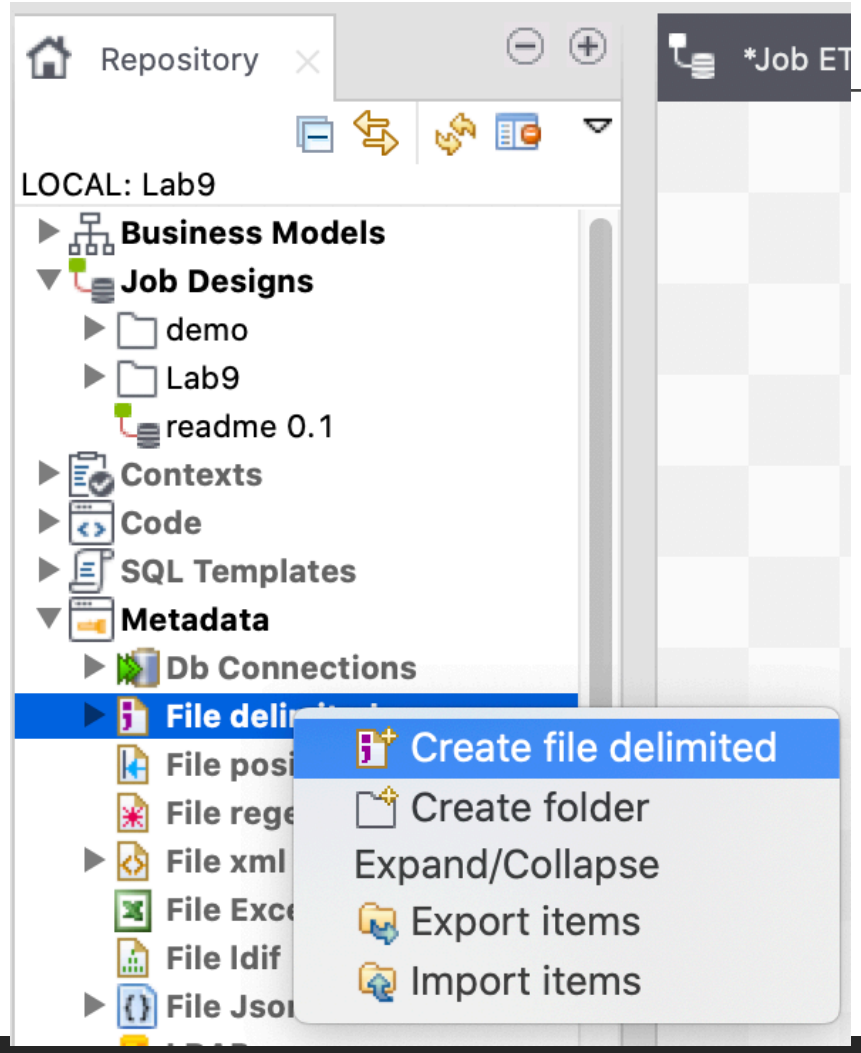
Task3

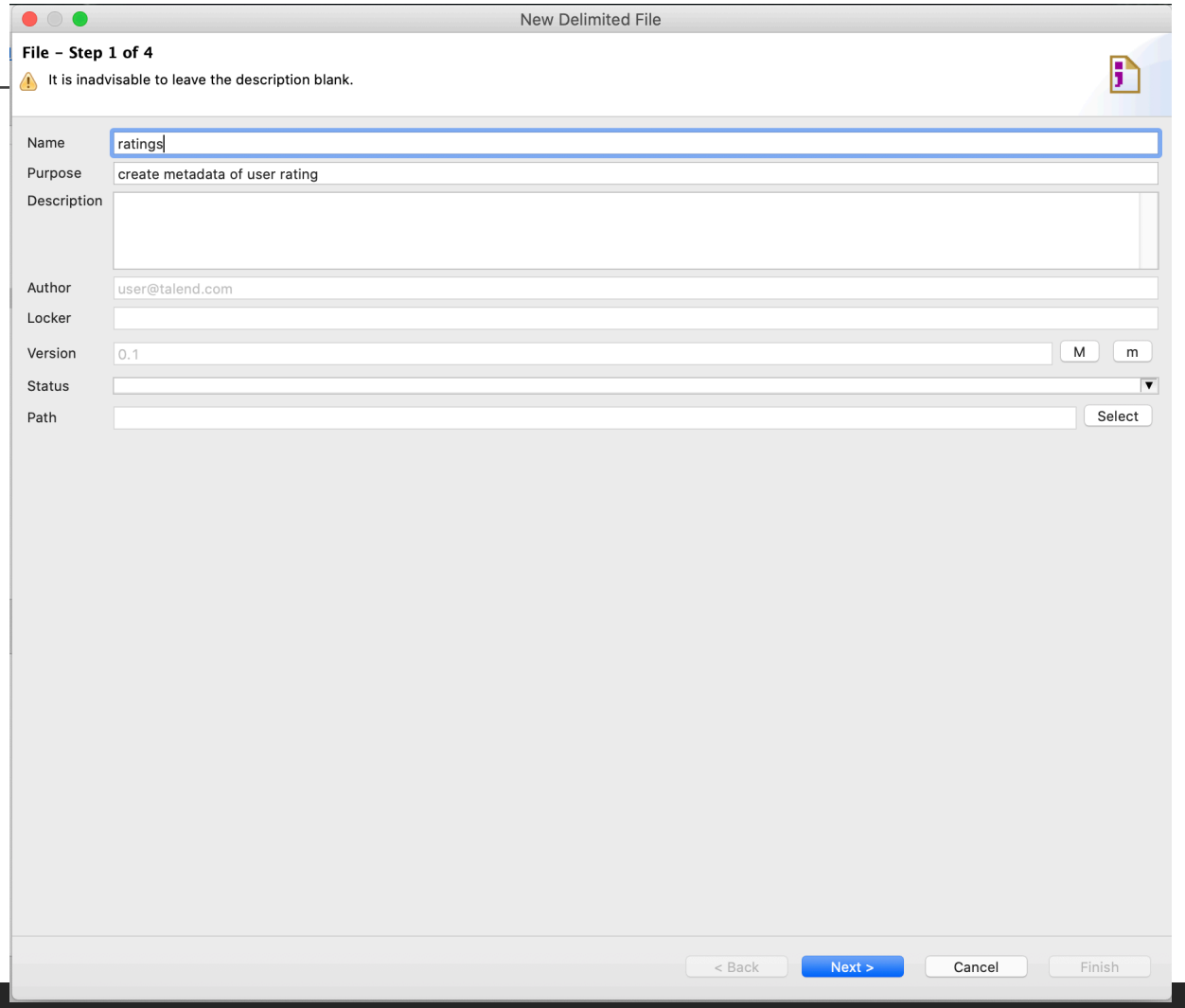
3.2 Create TJob#2: Convert Json to CSV File (3)



Task3

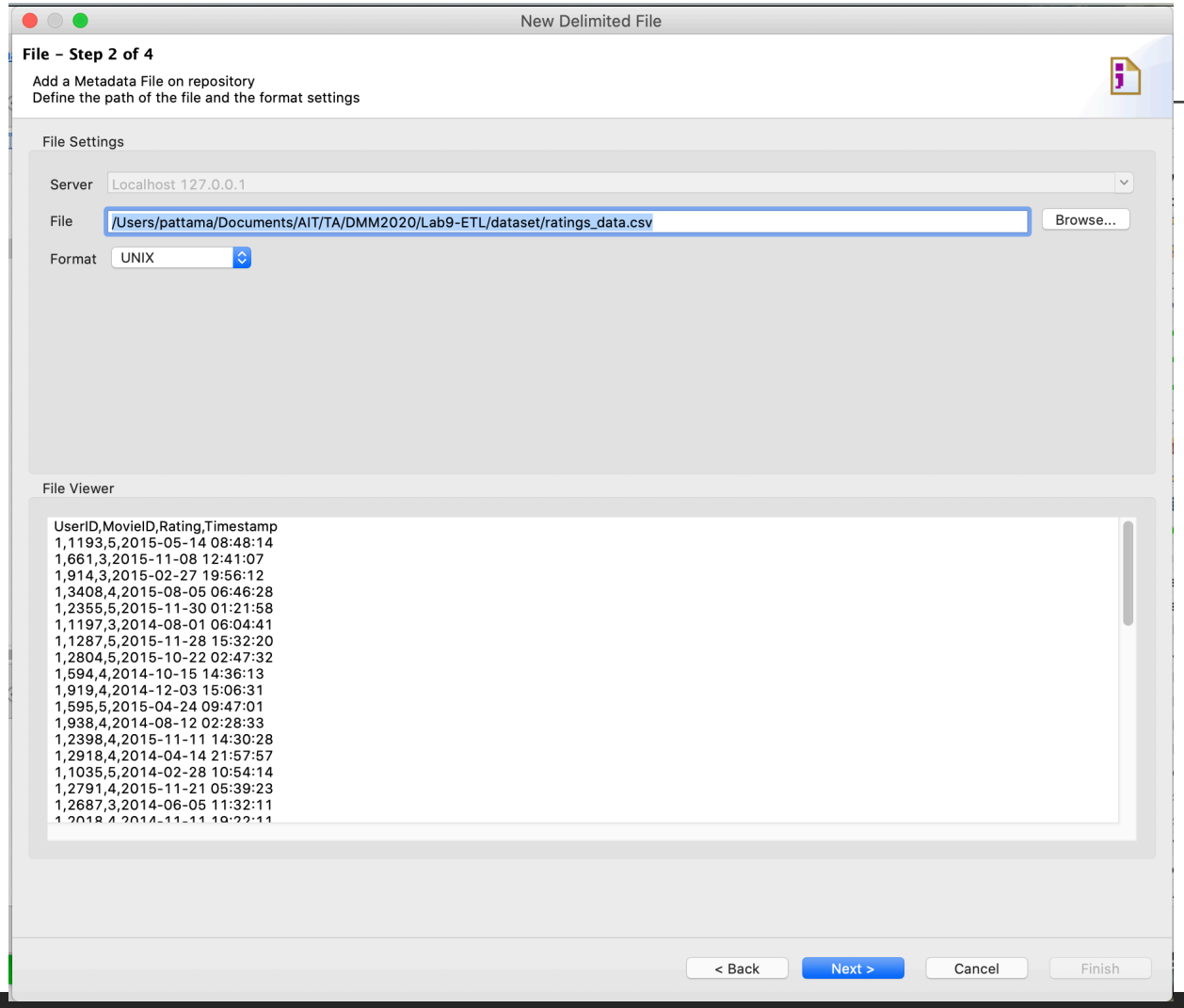
3.3 Create Metadata for movie rating (File Delimited)(1)

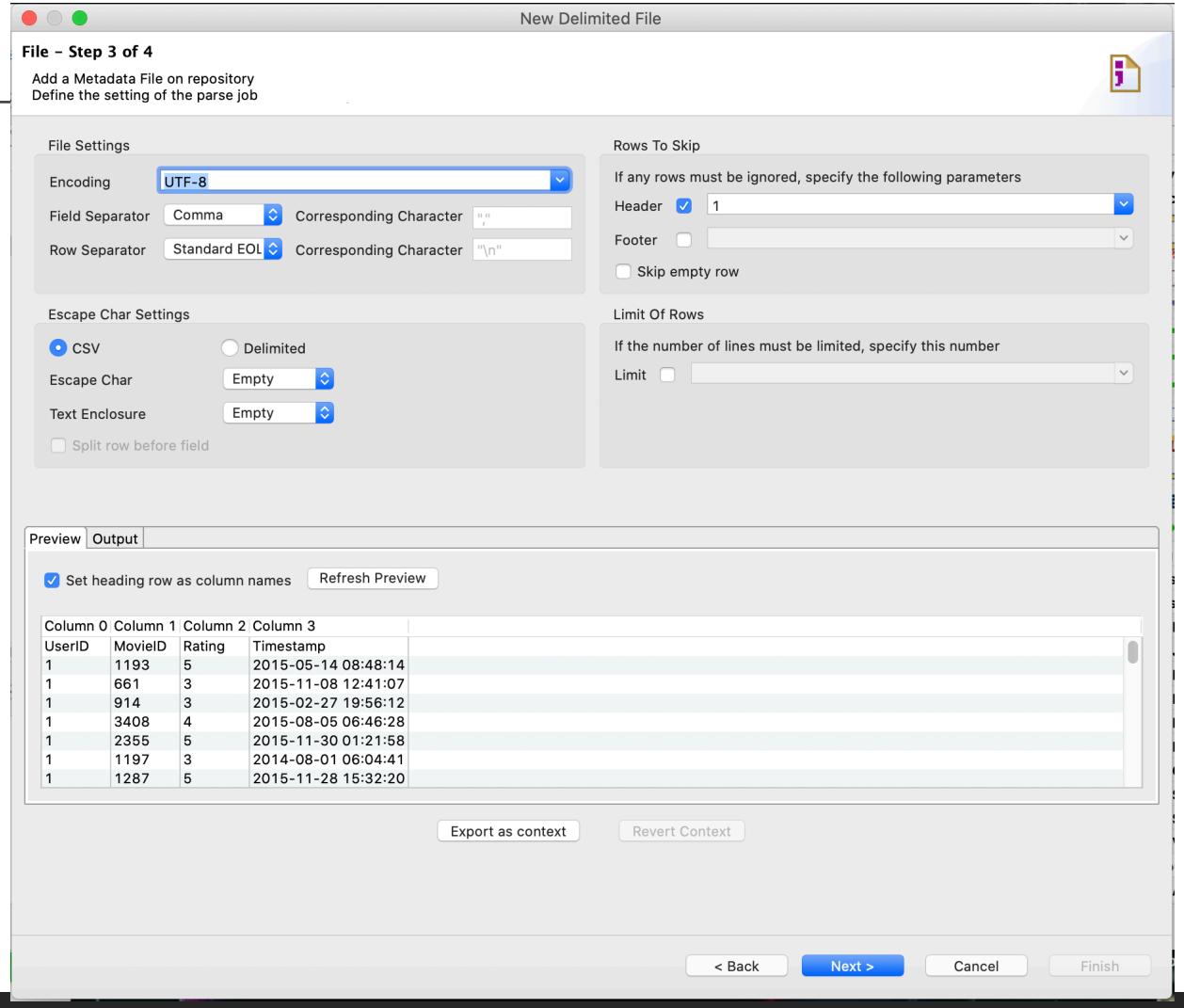




Task3

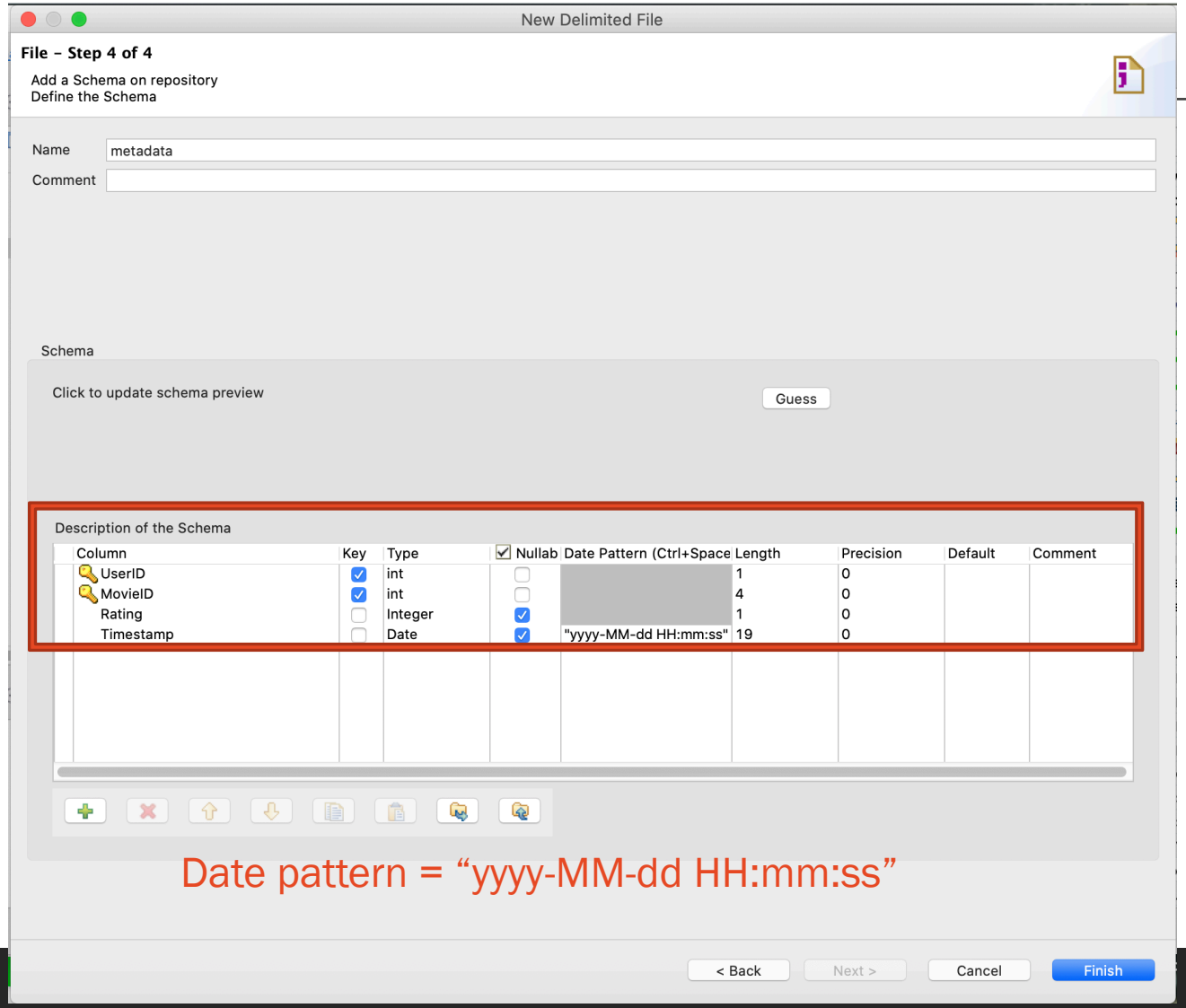
3.3 Create Metadata for movie rating (File Delimited)(2)

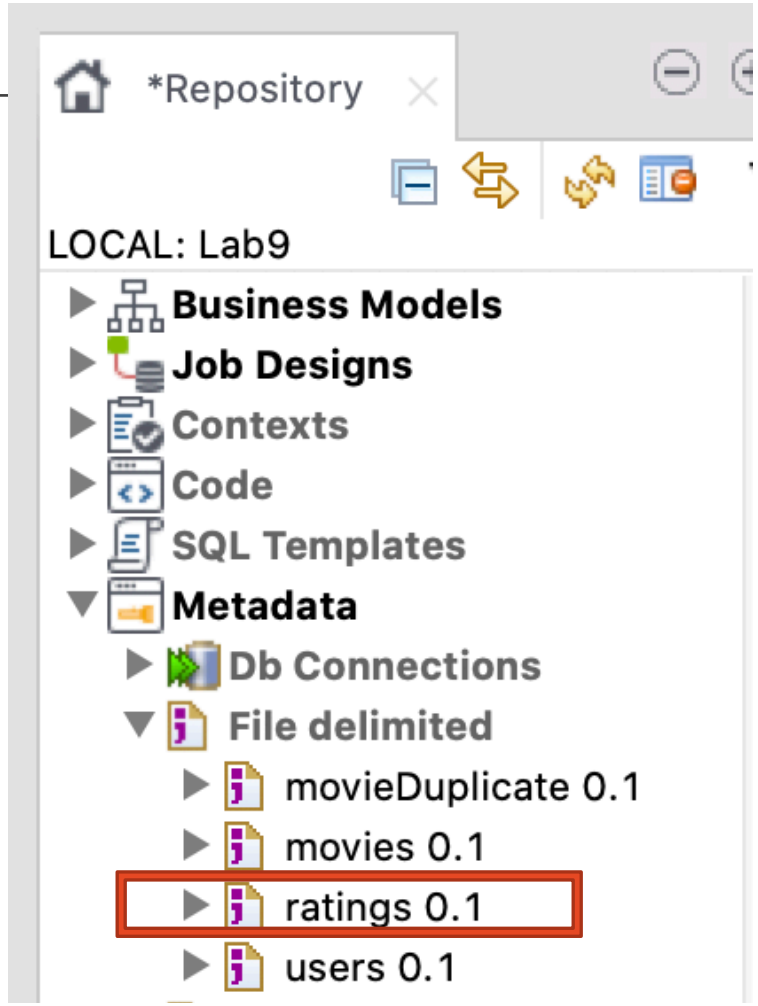




Task3

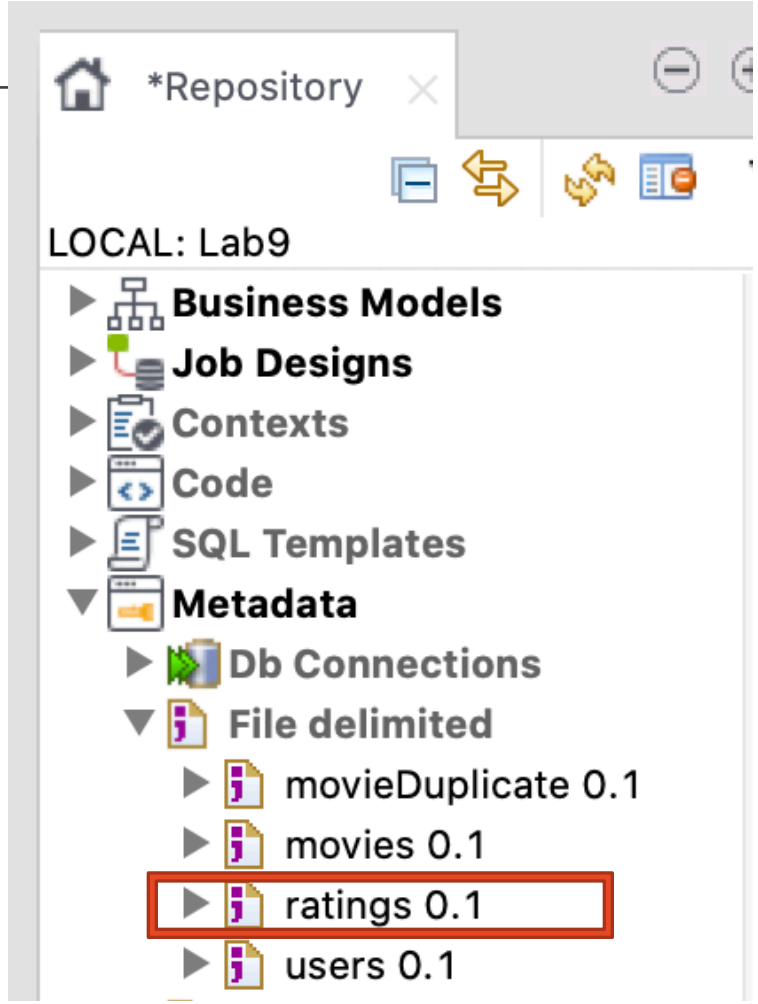
3.3 Create Metadata for movie rating (File Delimited)(3)





Output of Task3

1. Metadata of File Delimited “ratings”



The Results from Extraction Process

MovieID	Title	Year
1	Toy Story (1995)	1995
2	Jumanji (1995)	1995
3	Grumpier Old Men (1995)	1995
4	Waiting to Exhale (1995)	1995
5	Father of the Bride Part II (1995)	1995
6	Heat (1995)	1995
7	Sabrina (1995)	1995
8	Tom and Huck (1995)	1995
9	Sudden Death (1995)	1995
10	GoldenEye (1995)	1995
11	American President, The (1995)	1995
12	Dracula: Dead and Loving It (1995)	1995
13	Balto (1995)	1995
14	Nixon (1995)	1995
15	Cutthroat Island (1995)	1995
16	Casino (1995)	1995

Movies

UserID	Gender	BirthDate	Occupation	Zipcode
1	Female	2019-05-10	10	48067
2	Male	1964-11-10	16	70072
3	Male	1995-06-27	15	55117
4	Male	1975-11-07	7	02460
5	Male	1995-05-06	20	55455
6	Female	1970-02-27	9	55117
7	Male	1985-06-05	1	06810
8	Male	1995-05-27	12	11413
9	Male	1995-03-13	17	61614
10	Female	1985-07-29	1	95370
11	Female	1995-10-01	1	04093
12	Male	1995-01-21	12	32793
13	Male	1975-06-18	1	93304
14	Male	1985-06-09	0	60126
15	Male	1995-01-02	7	22903

Users

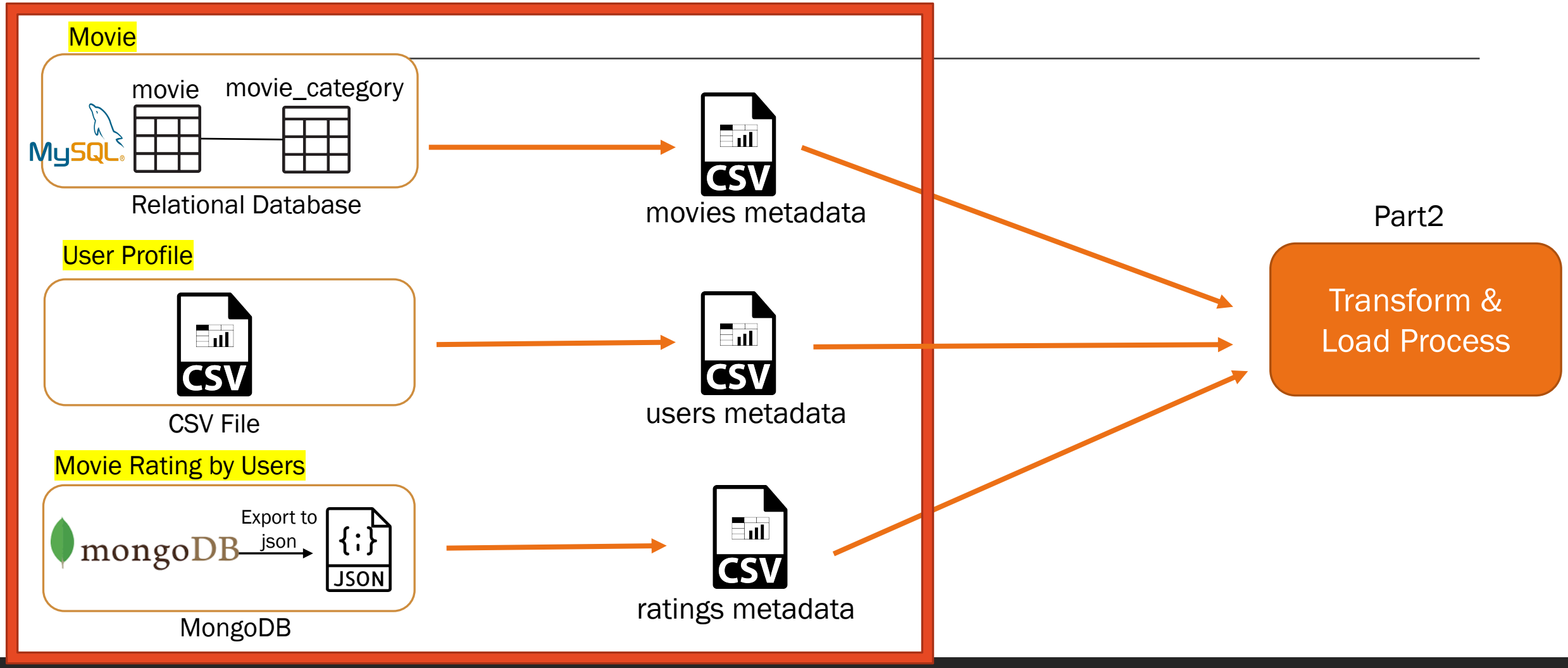
UserID	MovieID	Rating	Timestamp
1	1193	5	2015-05-14 08:48:14
1	661	3	2015-11-08 12:41:07
1	914	3	2015-02-27 19:56:12
1	3408	4	2015-08-05 06:46:28
1	2355	5	2015-11-30 01:21:58
1	1197	3	2014-08-01 06:04:41
1	1287	5	2015-11-28 15:32:20
1	2804	5	2015-10-22 02:47:32
1	594	4	2014-10-15 14:36:13
1	919	4	2014-12-03 15:06:31
1	595	5	2015-04-24 09:47:01
1	938	4	2014-08-12 02:28:33
1	2398	4	2015-11-11 14:30:28
1	2918	4	2014-04-14 21:57:57
1	1035	5	2014-02-28 10:54:14
1	2791	4	2015-11-21 05:39:23

Ratings

File Delimited Metadata

Part 2

Extraction Process in Part1 (Recap)



ETL Process

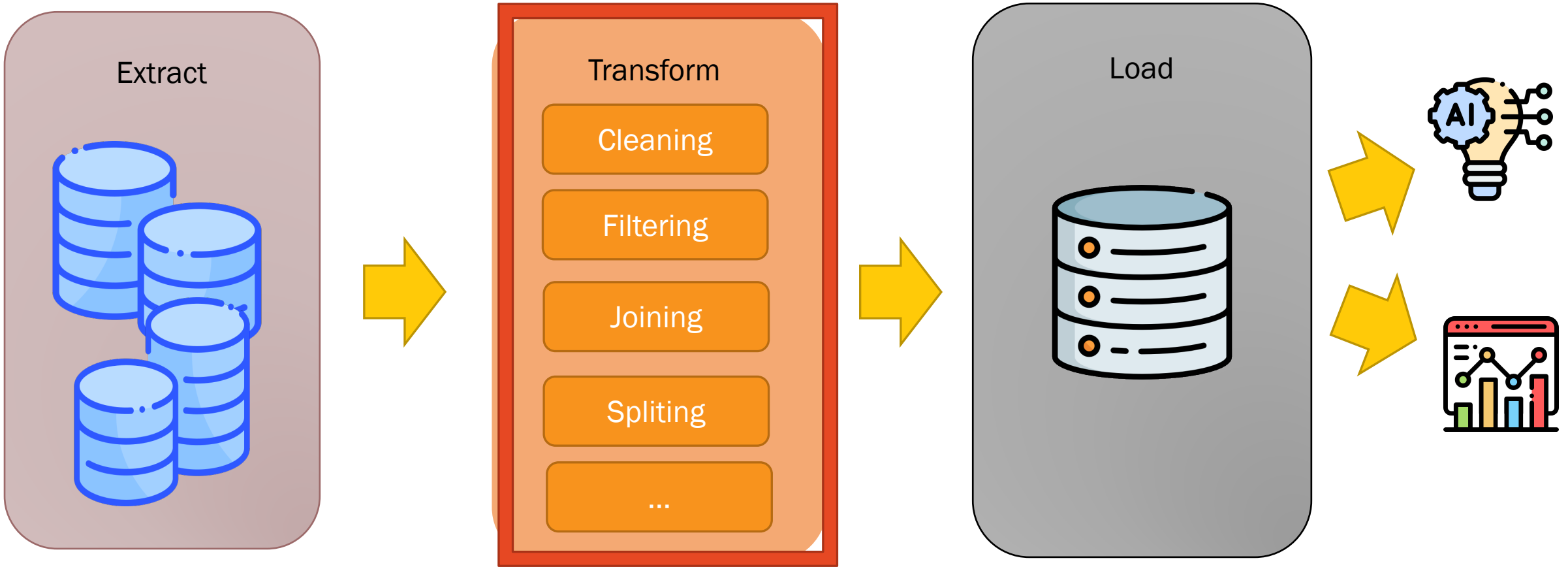
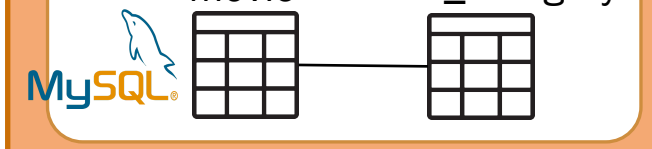


Image source: https://www.flaticon.com

Movie Rating Prediction

Input

Movie



Relational Database

User Profile

CSV File

Movie Rating by Users

MongoDB

ETL Process

Extract

Step1: get movie data from RDB and convert data to CSV file
Step2: get user profile in CSV format
Step3: get movie rating by users from MongoDB
3.1 export data to JSON file
3.2 convert Json to CSV file

Transform

Step.4 Prepare data for desired datasets
4.1 Transform movie category data
4.2 Remove Duplicate Records
4.3 Dealing with Missing Values
4.4 Integrate 3 datasets
4.5 Convert birthdate to Age
4.6 Convert Gender to numeric
4.7 Filter age >= 7 and Gender = 1|2
4.8 Merge output of 4.1 and 4.7

Load

Step.5 Write output CSV files

Talend

Task1 for step 1
- 1.1 Create Metadata for DB Connection
- 1.2 Create TJob#1
- 1.3 Create Metadata for movie

Task2 for step 2
- Create Metadata for user profile

Task3 for step 3
- 3.1 Create Metadata for movie rating (Json)
- 3.2 Create TJob#2
- 3.3 Create Metadata for movie rating (File Delimited)

Task4 for step 4.1
- 4.1 Create TJob#3
- 4.2 Create Metadata for movie category

Task5 for step 4.2-5
- Create TJob#4

Task4 for step 4.1

Step4.1: Transform movie category data

Task4:

4.1 Create TJob#3: Transform movie category data

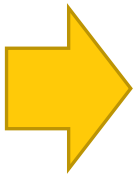
4.2 Create Metadata for movie category (File Delimited)

Task4

4.1 Create TJob#3: Transform movie category data

MovieID	Category
1	Animation
1	Children's
1	Comedy
2	Adventure
2	Children's
2	Fantasy
3	Comedy
3	Romance
4	Comedy
4	Drama
5	Comedy

movie category table



- Change movie category to numeric values for prediction model

MovieID	Animation	Children's	....	Romance
1	1	1	...	0
2	0	1	...	0
3	0	0	...	1

Desired output of movie category

Task4

4.1 Create TJob#3: Transform movie category data

MovieID	Category	Values
1	Animation	1
1	Children's	1
1	Comedy	1
2	Adventure	1
2	Children's	1
2	Fantasy	1
3	Comedy	1
3	Romance	1
4	Comedy	1
4	Drama	1
5	Comedy	1
6	Action	1
6	Crime	1
6	Thriller	1

movie category table

Pivot Table:

- Set value = 1 for all records in column “Values”
- Pivot “Category” column to header

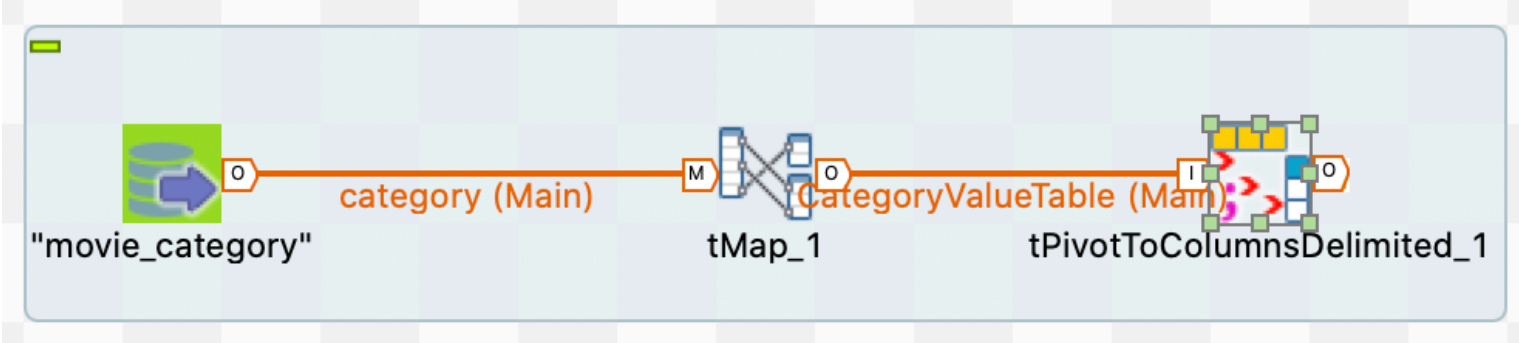
MovieID	Animation	Children’s	….	Romance
1	1	1	…	0
2	0	1	…	0
3	0	0	…	1

Task4

4.1 Create TJob#3: Transform movie category data

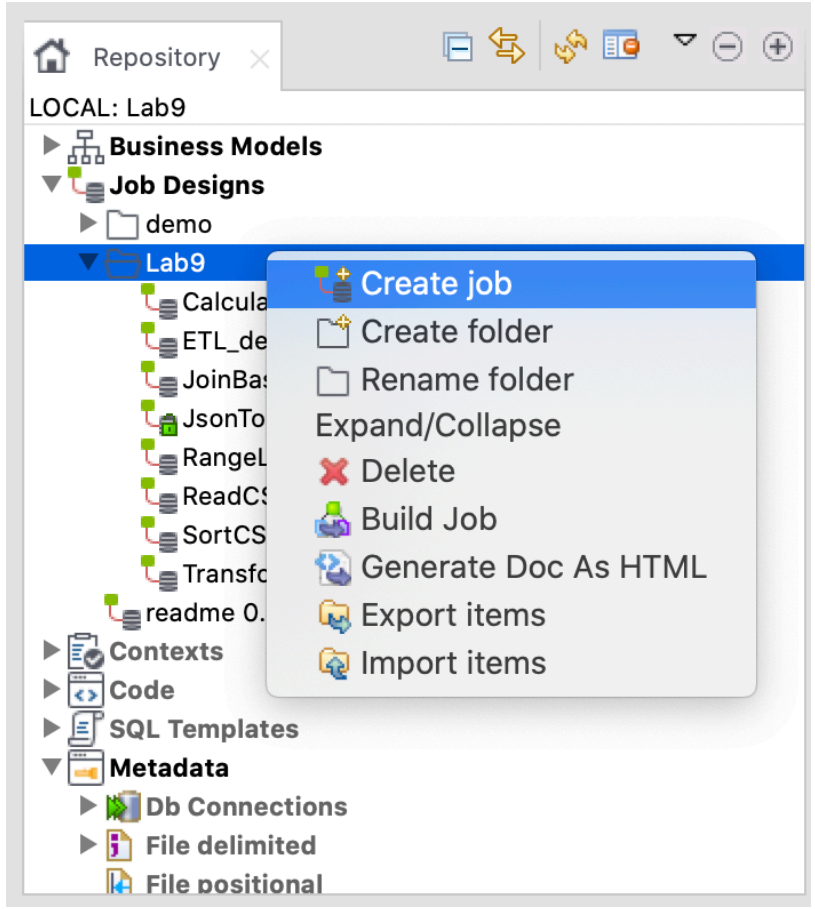
Purpose : Transform movie category data for prediction movel

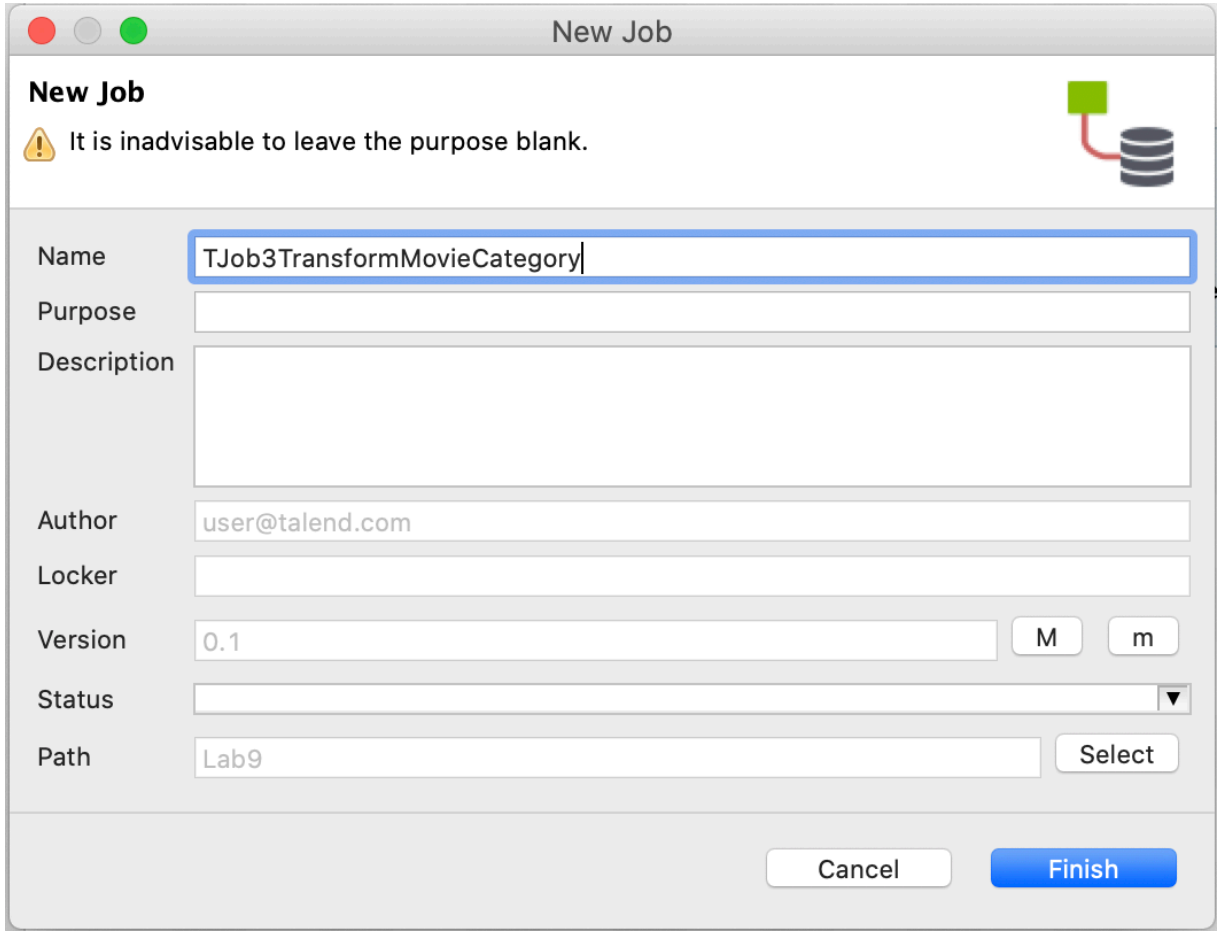
Components: tDBInput(MySQL) , tmap, tPivotToColumnsDelimited



Task4

4.1 Create TJob#3: Transform movie category data

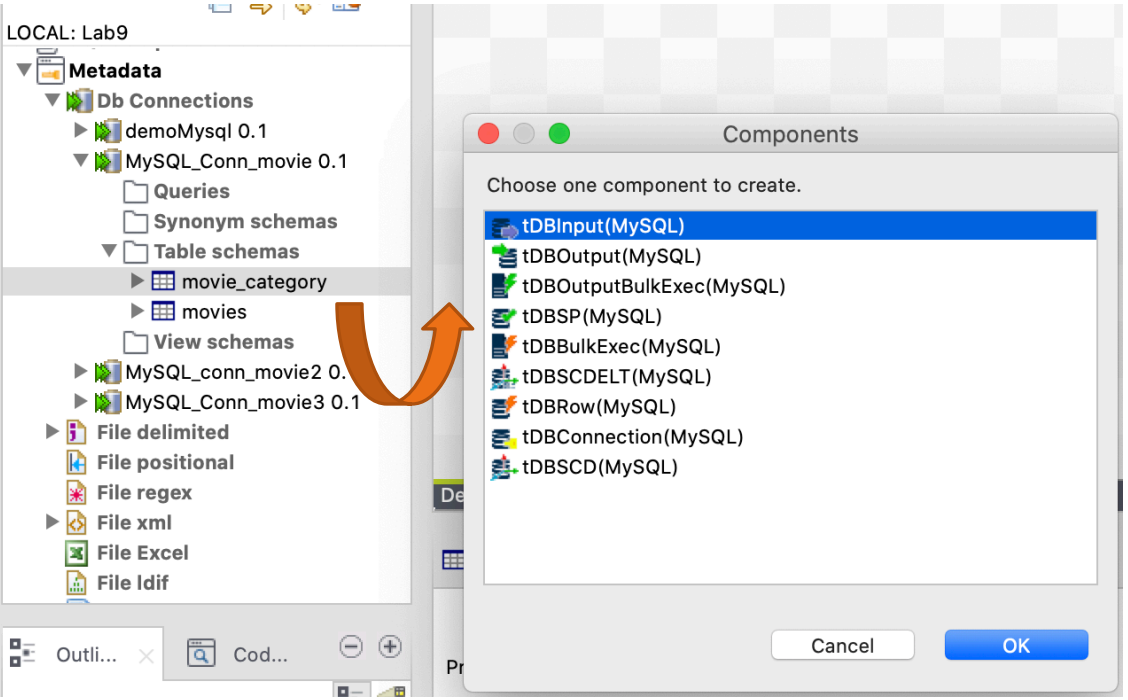


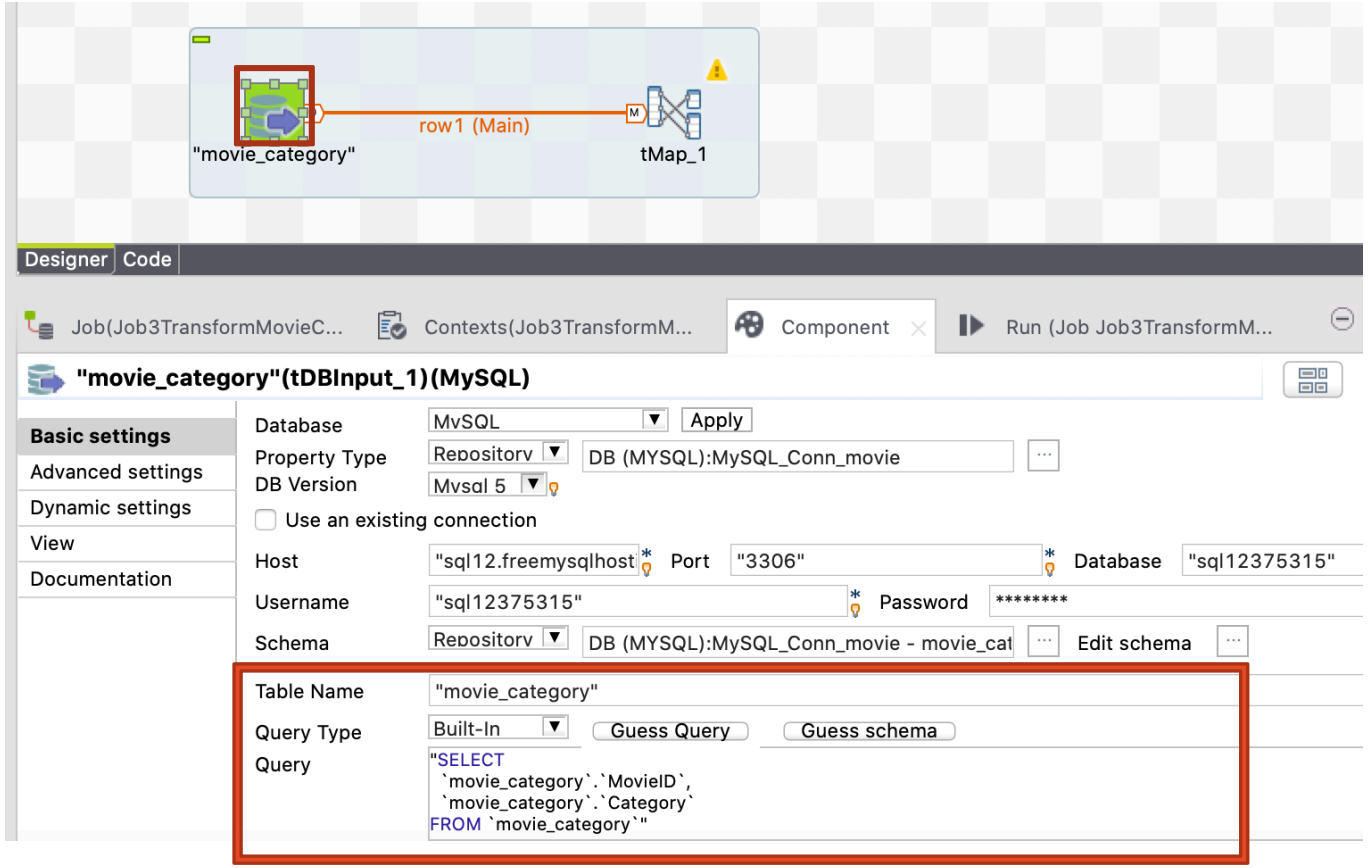


Task4

4.1 Create TJob#3: Transform movie category data

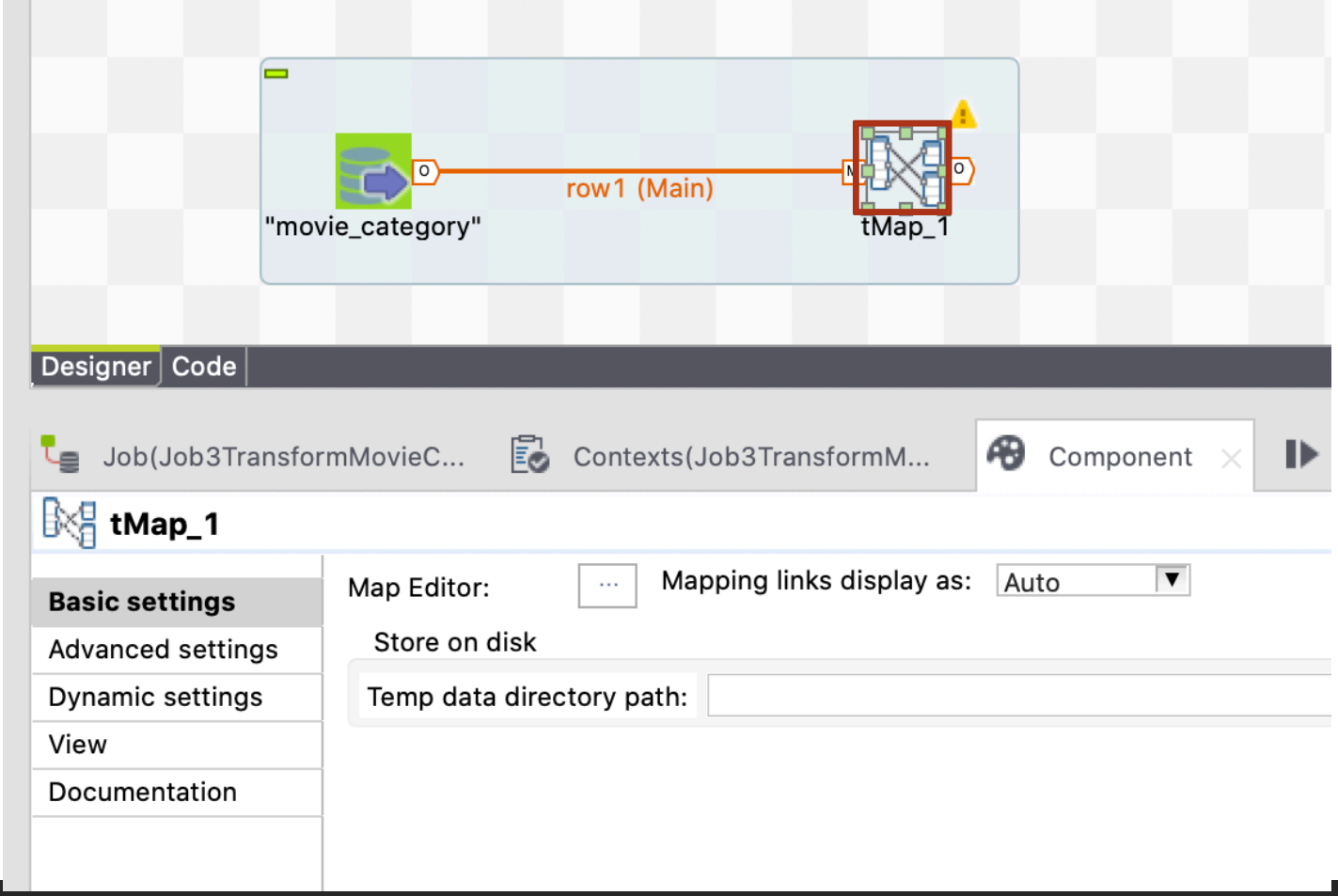
- Drag Metadata of table “movie_category” under “MySQL_Conn_movie” to Design Workspace





Task4

4.1 Create TJob#3: Transform movie category data



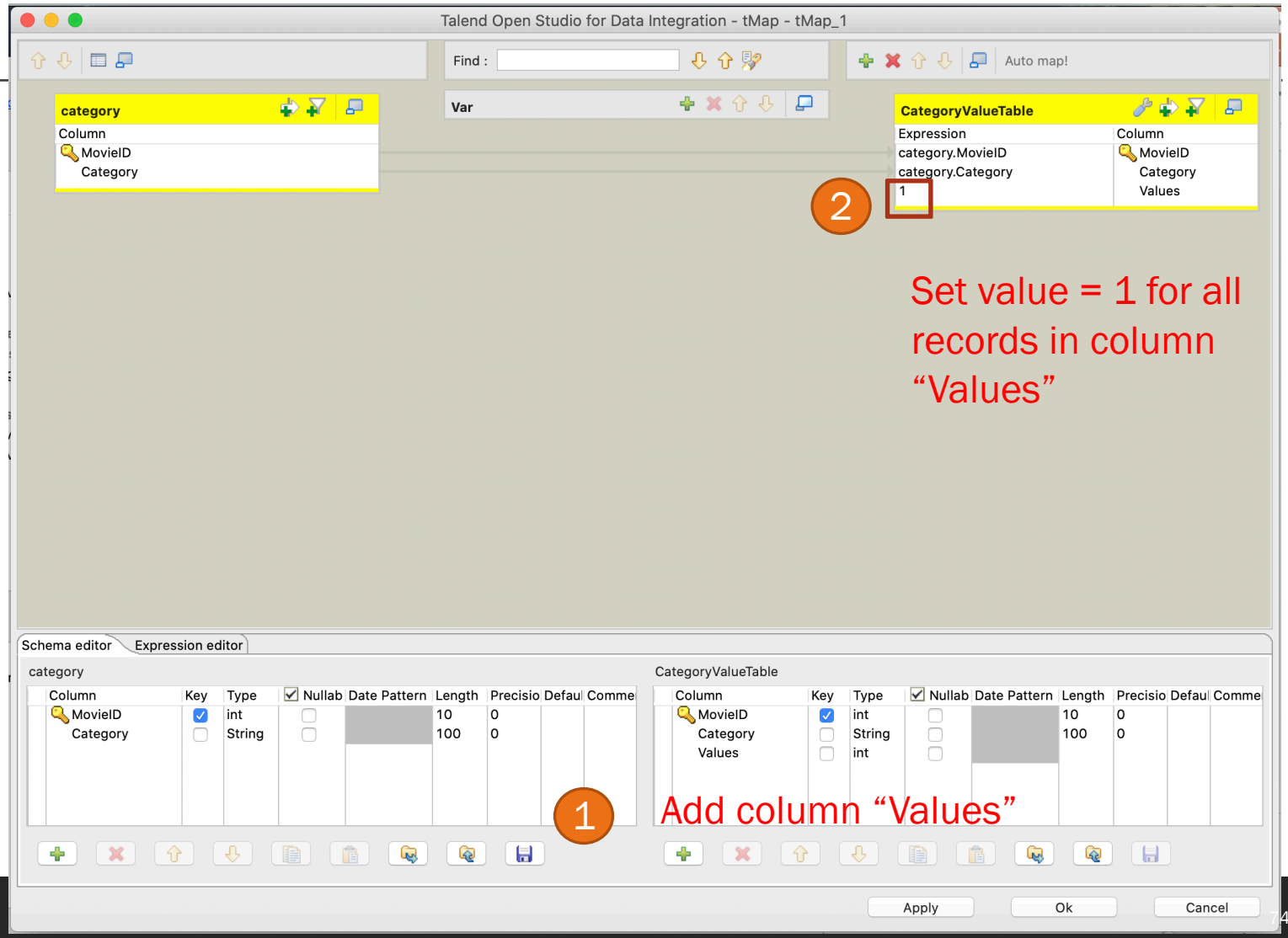
- **tMap** Component
  - get data from one or more sources
  - transforms data
  - sends the transformed data to one or more destinations.
- Double click at tMap component

Task4

4.1 Create TJob#3: Transform movie category data

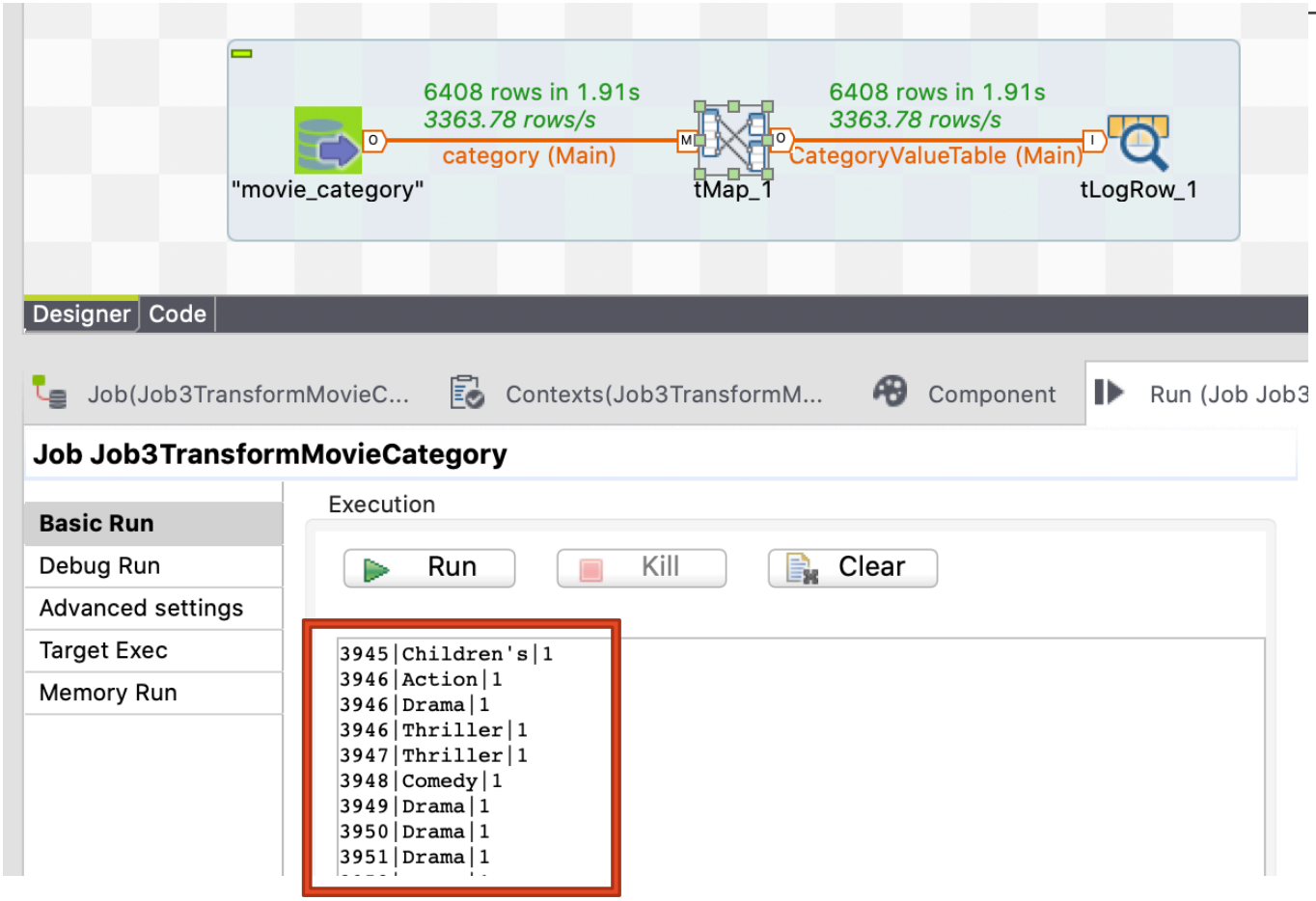
Set value = 1 for all records in column “Values”

MovieID	Category	Values
1	Animation	1
1	Children's	1
1	Comedy	1
2	Adventure	1
2	Children's	1
2	Fantasy	1
3	Comedy	1
3	Romance	1
4	Comedy	1
4	Drama	1
5	Comedy	1
6	Action	1
6	Crime	1
6	Thriller	1



Task4

4.1 Create TJob#3: Transform movie category data

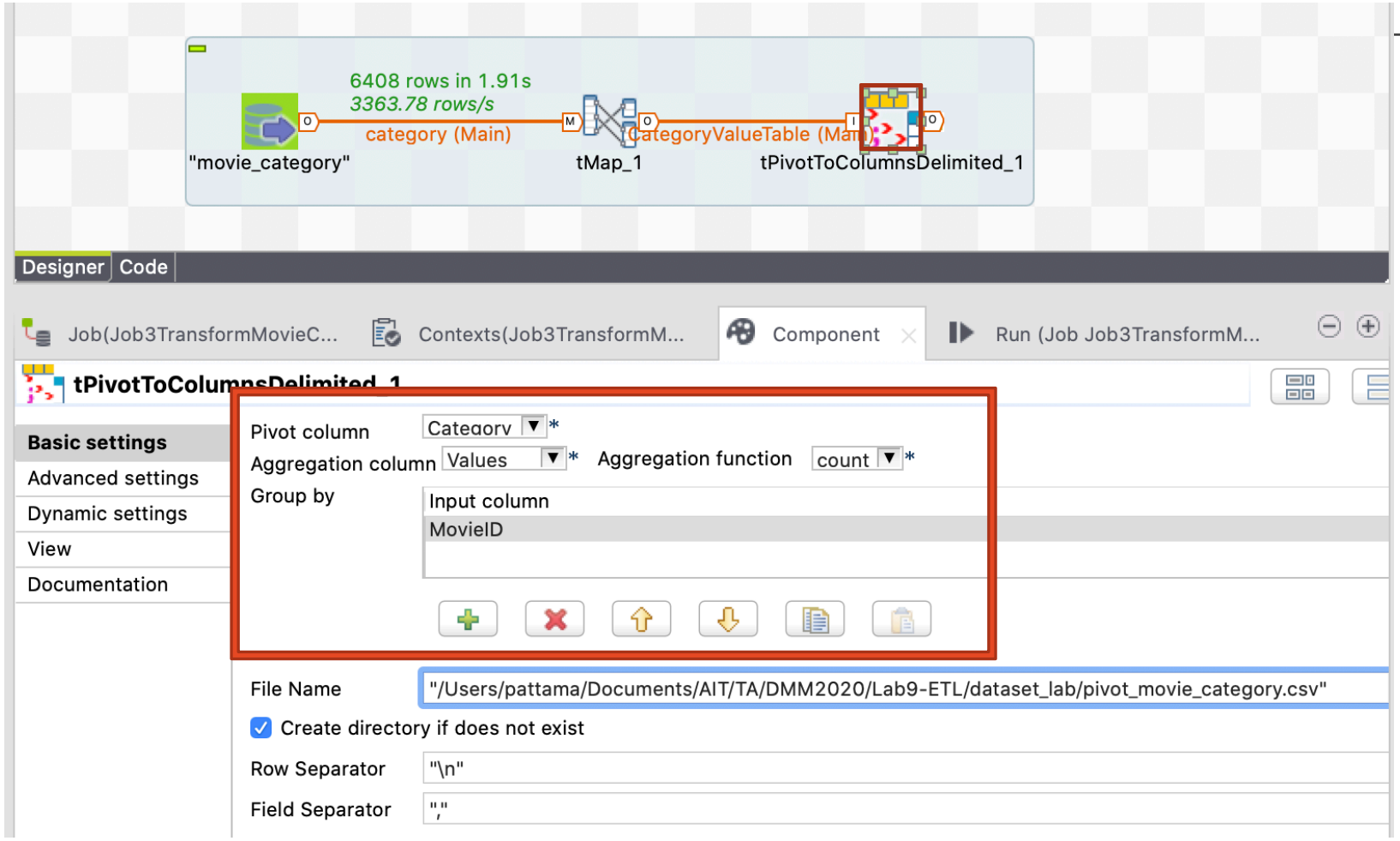


- Add tLogRow to check output
- Example Output:
  - MovieID
  - Category
  - Values

Example Output

Task4

4.1 Create TJob#3: Transform movie category data



- Pivot column: Category
- Aggregation column: Values
- Aggregation function: count
- Group by: MovieID

Task4

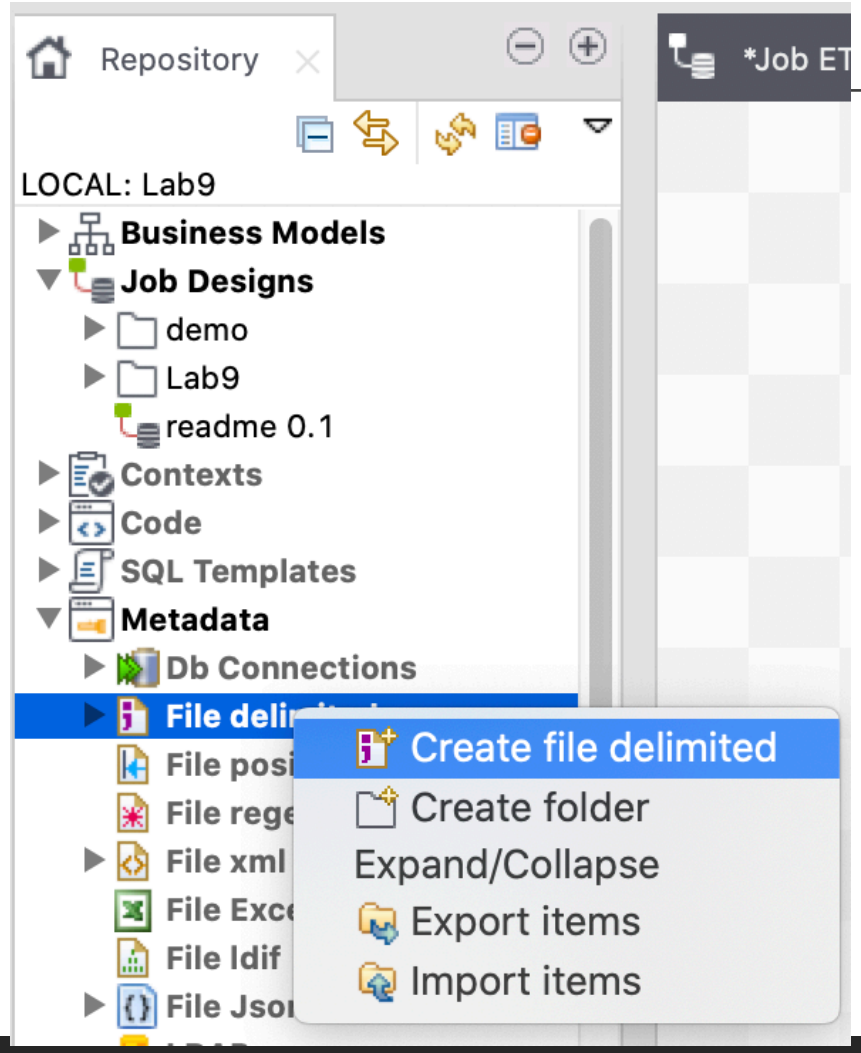
4.1 Create TJob#3: Transform movie category data

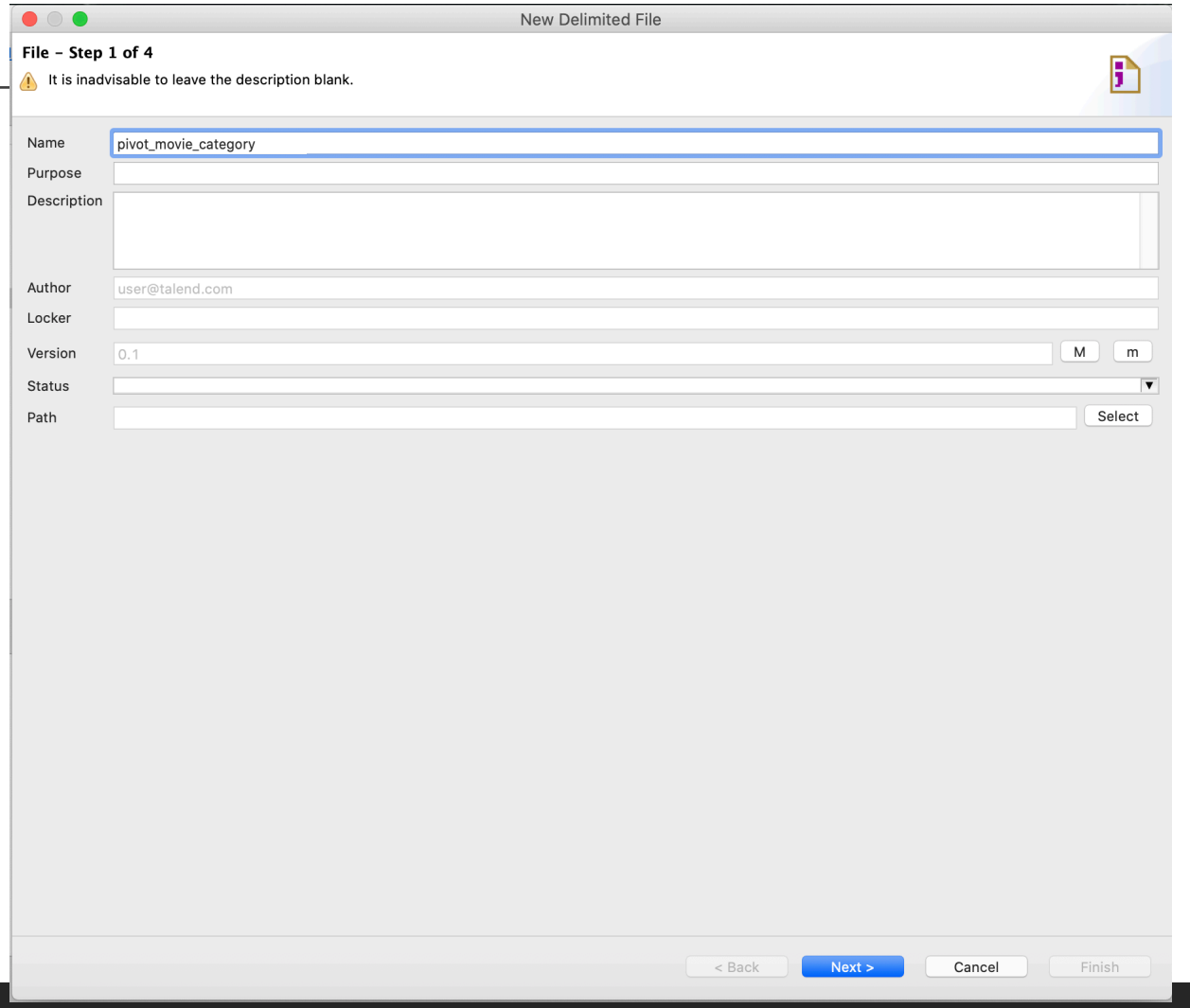
MovieID	Animation	Children's	Comedy	Adventure	Fantasy	Romance	Drama	Action	Crime	Thriller	Horror	Sci-Fi
1	1	1	1									
2		1		1	1							
3			1			1						
4			1				1					
5			1									
6								1	1	1		
7			1			1						
8		1		1								
9								1				
10				1				1		1		
11			1			1	1					
12			1								1	
13	1	1										
14							1					
15				1		1		1				
16							1			1		
17						1	1					

Output after pivot column

Task4

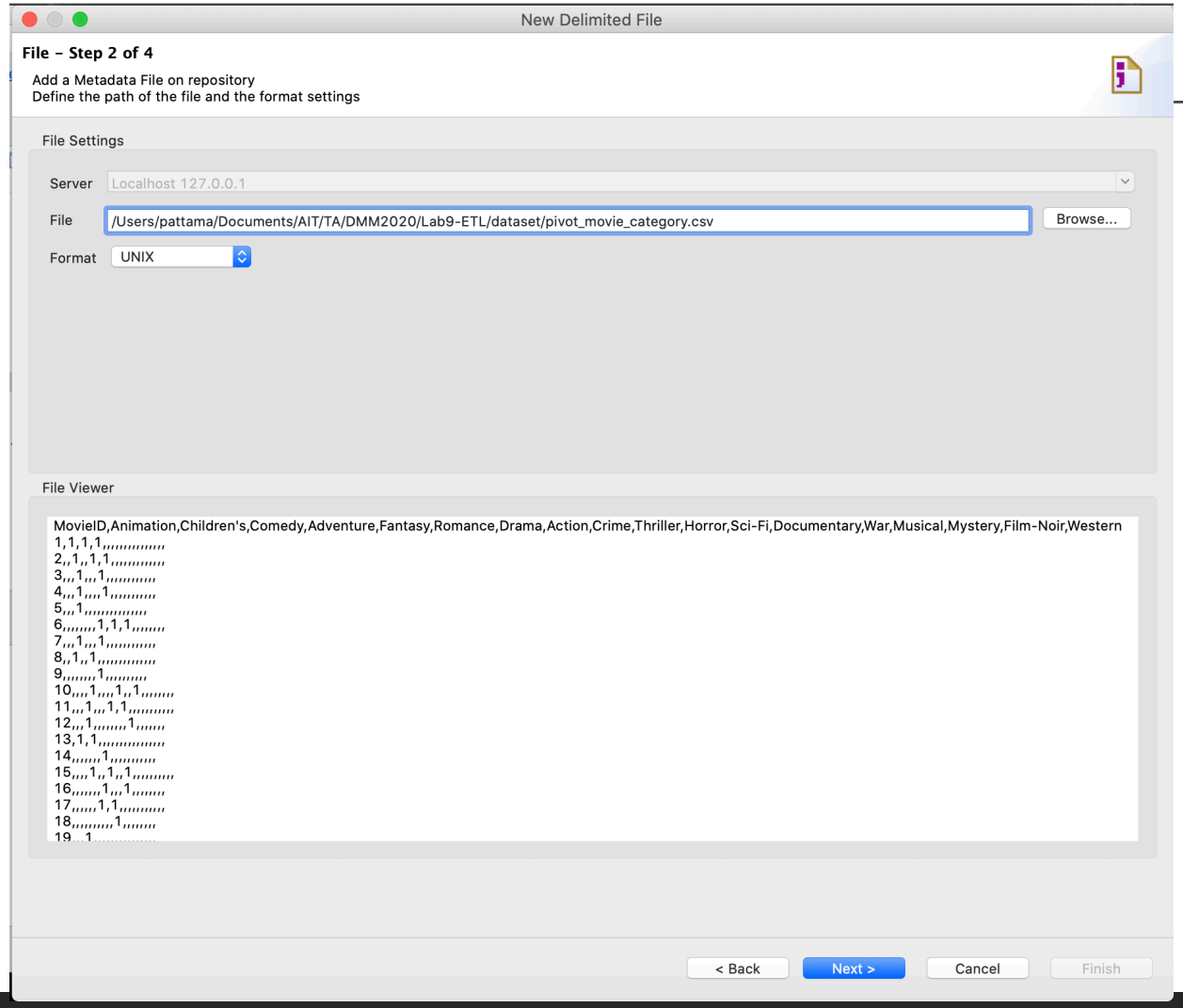
4.2 Create Metadata for movie category (File Delimited) (1)

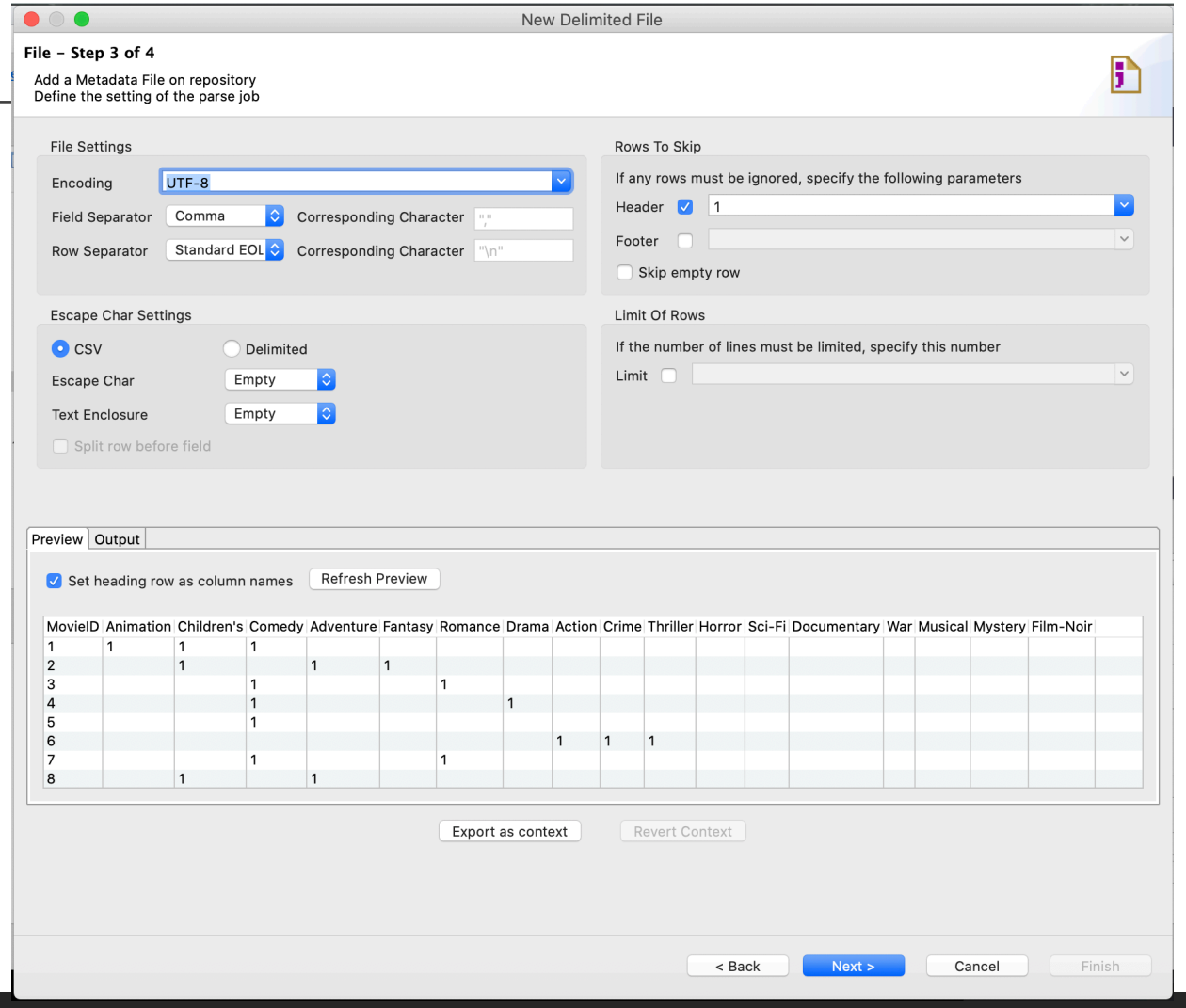




Task4

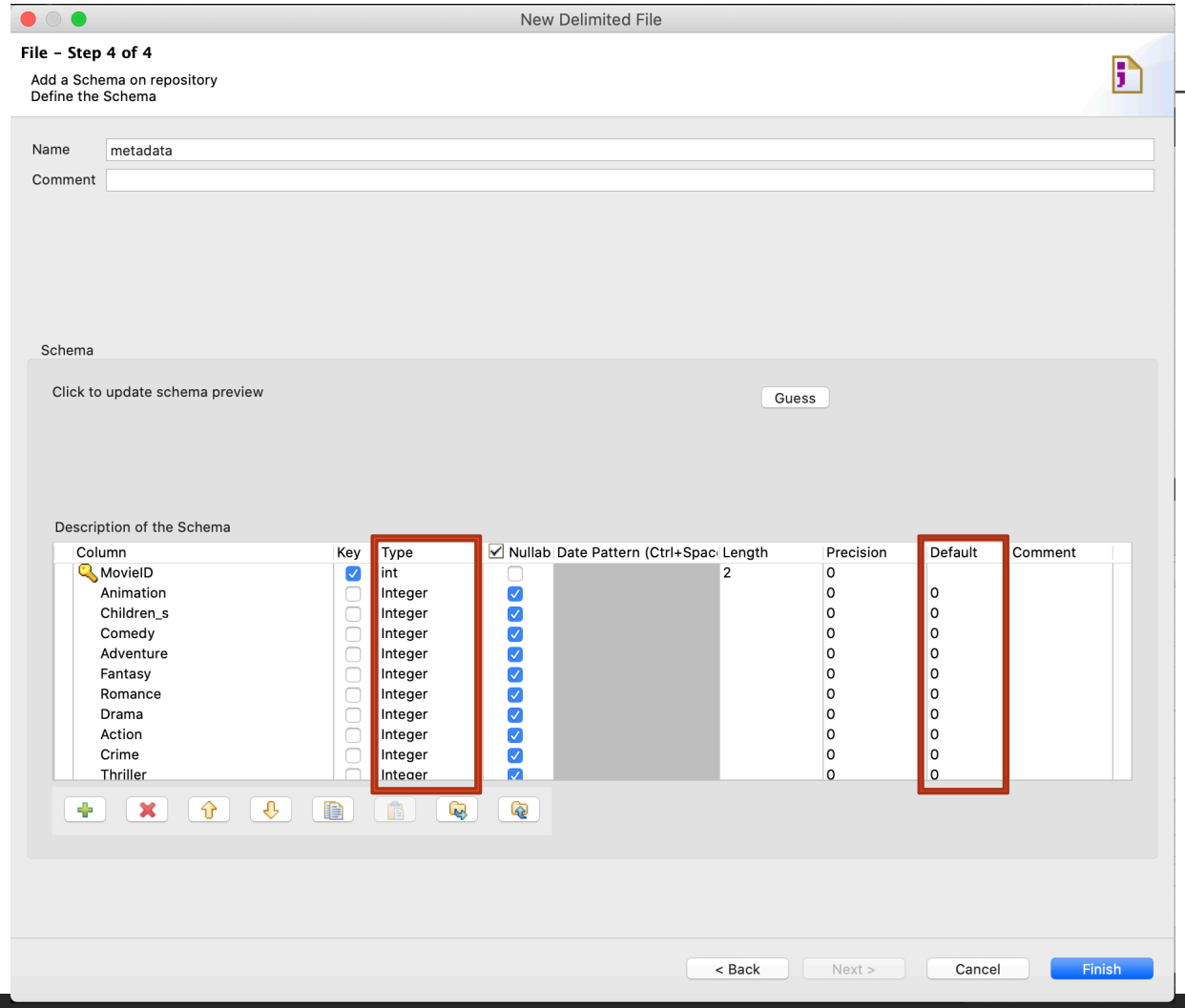
4.2 Create Metadata for movie category (File Delimited) (2)

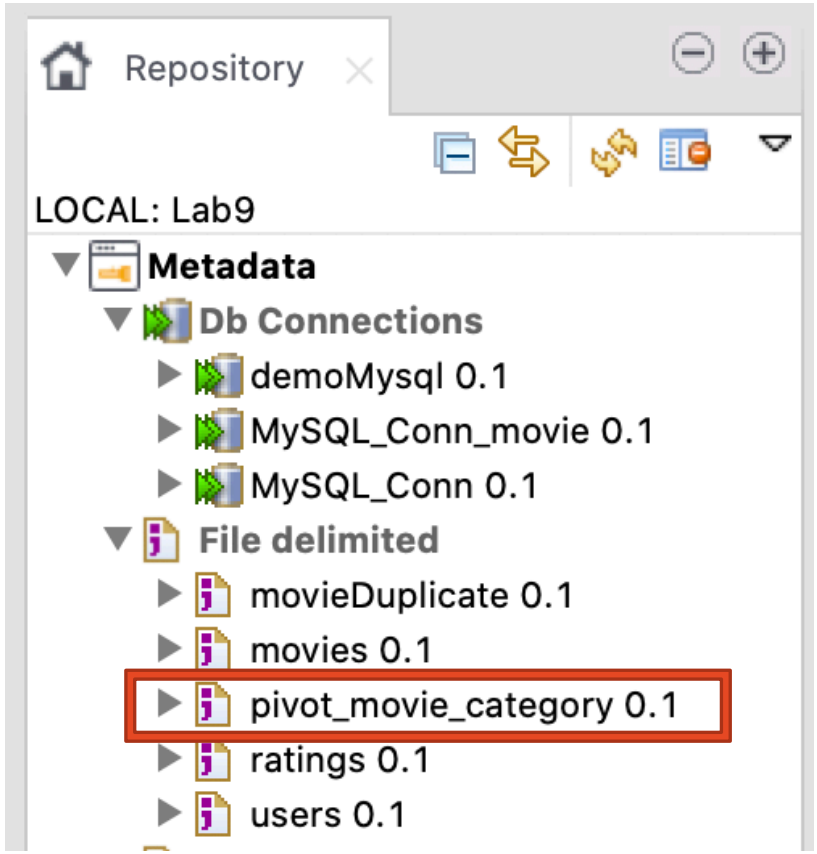




Task4

4.2 Create Metadata for movie category (File Delimited) (3)

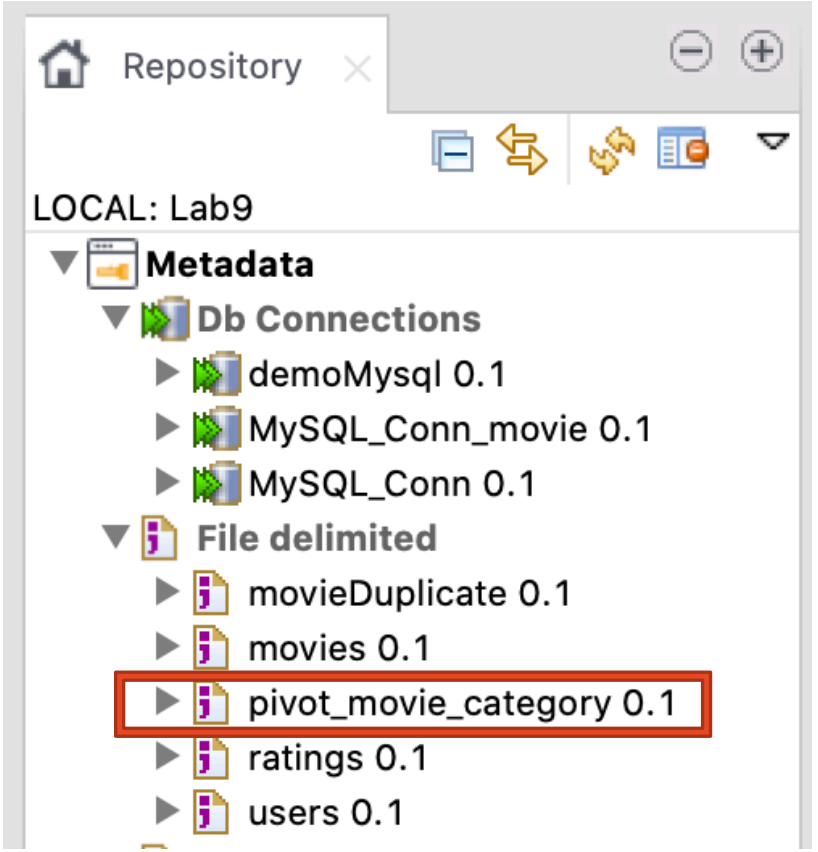




Output of Task4

1. Metadata of File Delimited “pivot_movie_category”

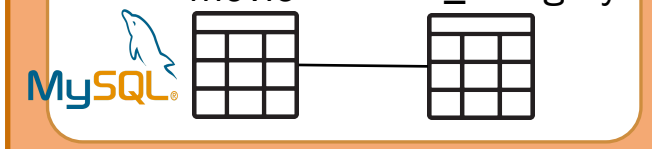
Using this metadata in Tjob#4 !!!



Movie Rating Prediction

Input

Movie



Relational Database

User Profile

CSV File

Movie Rating by Users

MongoDB

ETL Process

Extract

Step1: get movie data from RDB and convert data to CSV file
Step2: get user profile in CSV format
Step3: get movie rating by users from MongoDB
3.1 export data to JSON file
3.2 convert Json to CSV file

Transform

Step.4 Prepare data for desired datasets
4.1 Transform movie category data
4.2 Remove Duplicate Records
4.3 Integrate 3 datasets
4.4 Convert birthdate to Age
4.5 Convert Gender to numeric
4.6 Filter age >= 7 and Gender = 1|2
4.7 Merge output of 4.1 and 4.6

Load

Step.5 Write output CSV files

Talend

Task1 for step 1
- 1.1 Create Metadata for DB Connection
- 1.2 Create TJob#1
- 1.3 Create Metadata for movie

Task2 for step 2
- Create Metadata for user profile

Task3 for step 3
- 3.1 Create Metadata for movie rating (Json)
- 3.2 Create TJob#2
- 3.3 Create Metadata for movie rating (File Delimited)

Task4 for step 4.1
- 4.1 Create TJob#3
- 4.2 Create Metadata for movie category

Task5 for step 4.2-5
- Create TJob#4

Task5 for step 4.2-5

Step4.2-5: Prepare dataset for desired datasets

Task5:

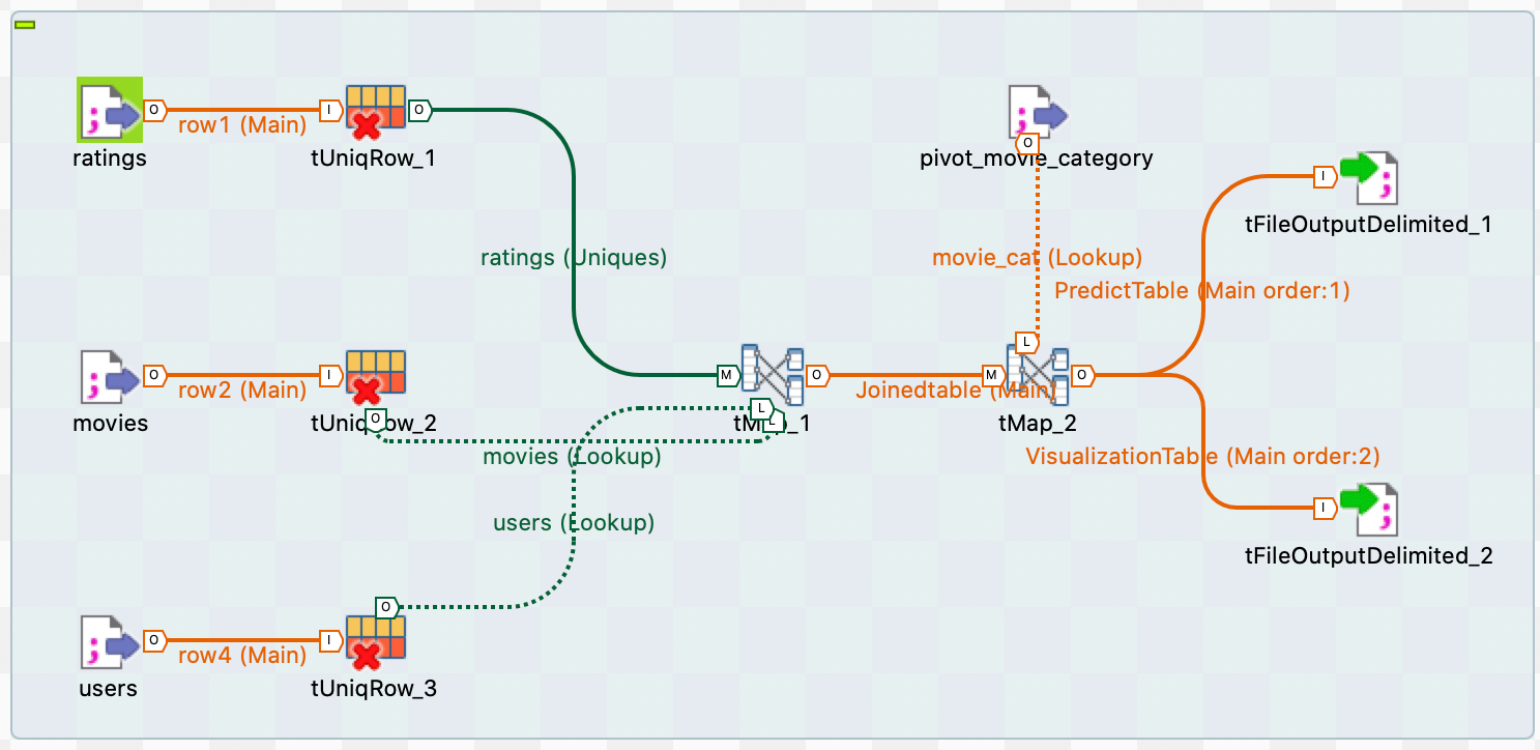
Create TJob#4

Task5

Prepare dataset for desired datasets

Purpose : Prepare dataset for desired datasets

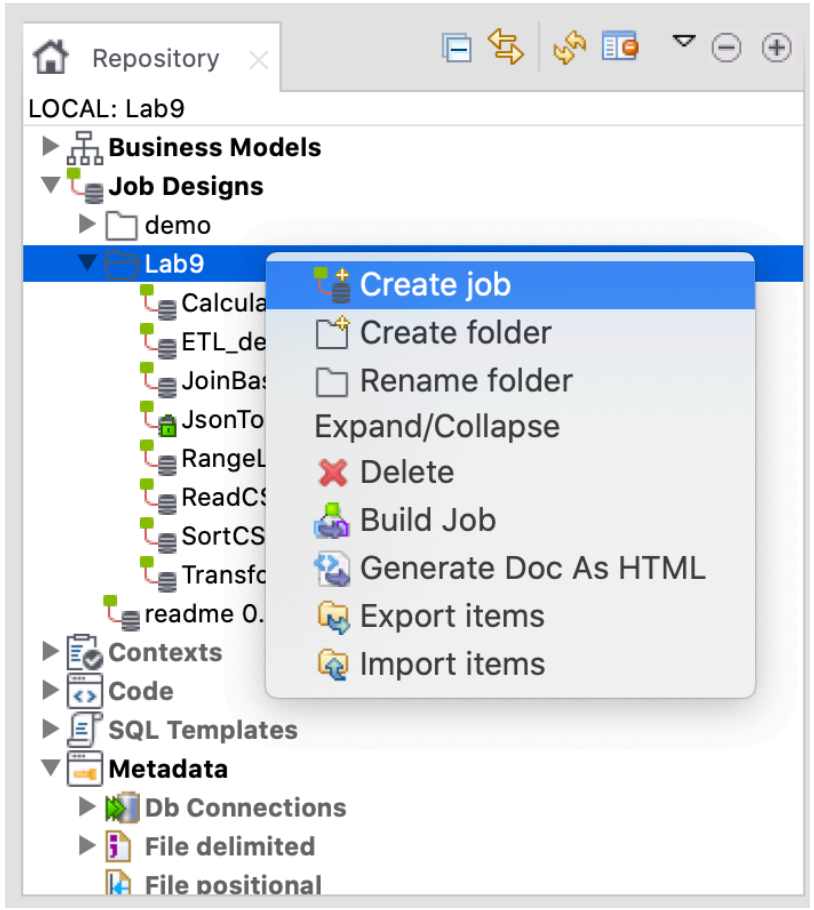
Components: tFileInputDelimited, tUniqRow, tMap, tFileOutputDelimited_1

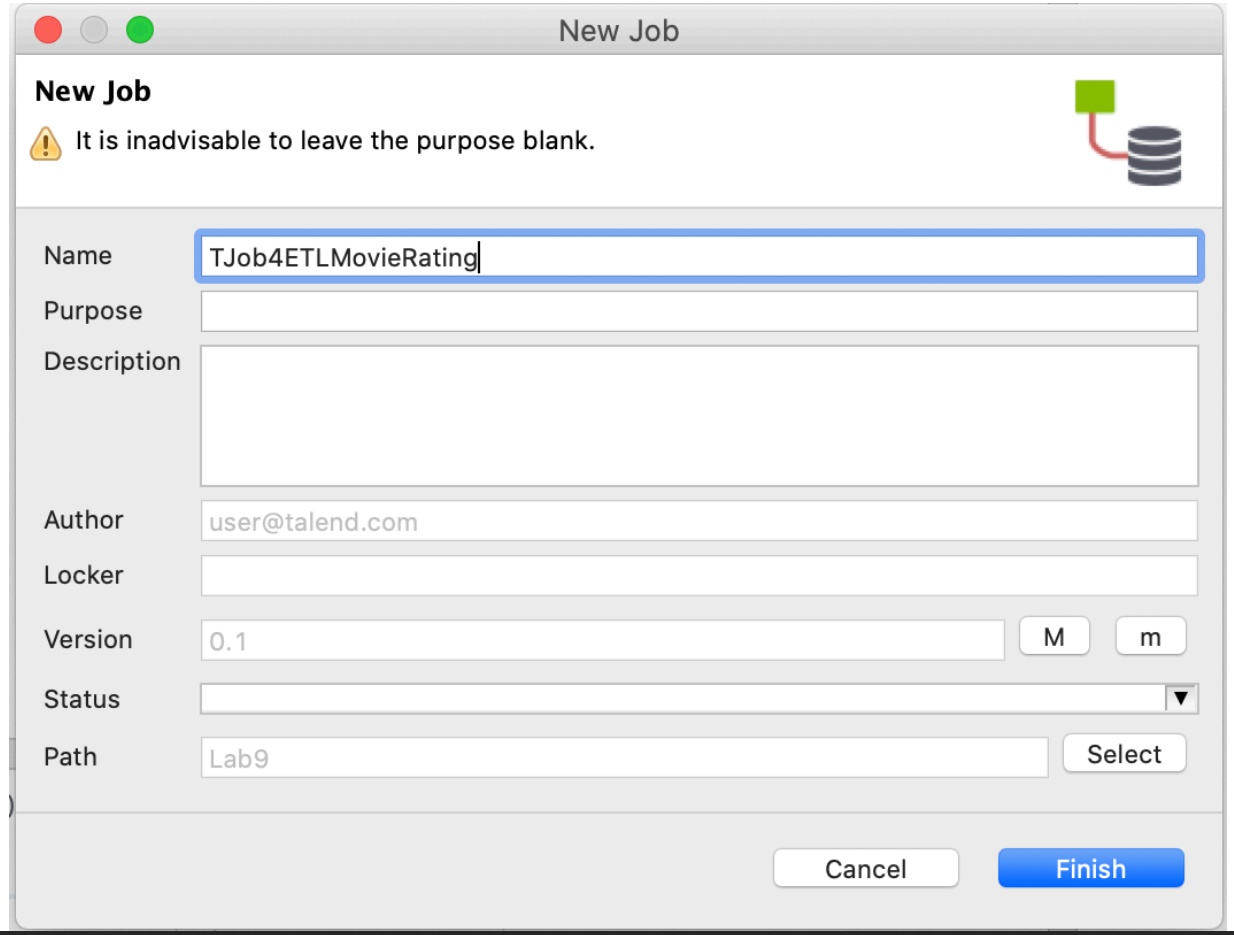


- **tUniqRow** - Compares entries and sorts out duplicate entries from the input flow.
- **tMap**
  - get data from one or more sources
  - transforms data
  - sends the transformed data to one or more destinations.

Task5

Create TJob#4: Prepare dataset for desired datasets

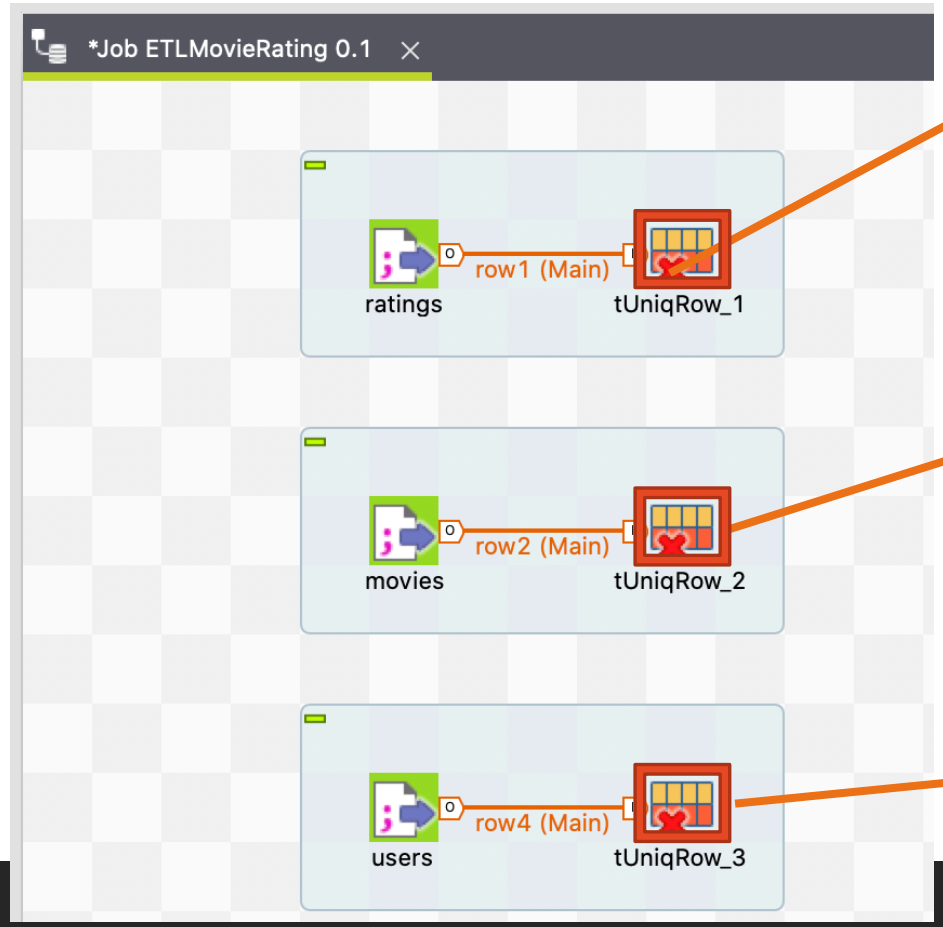


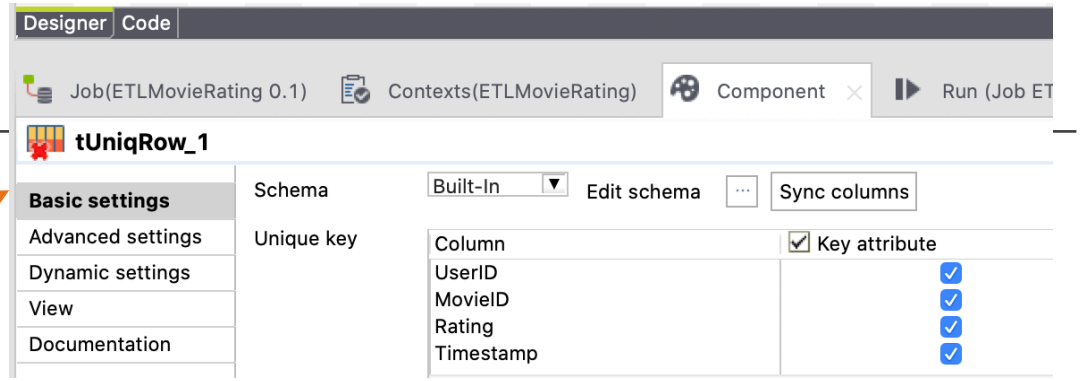


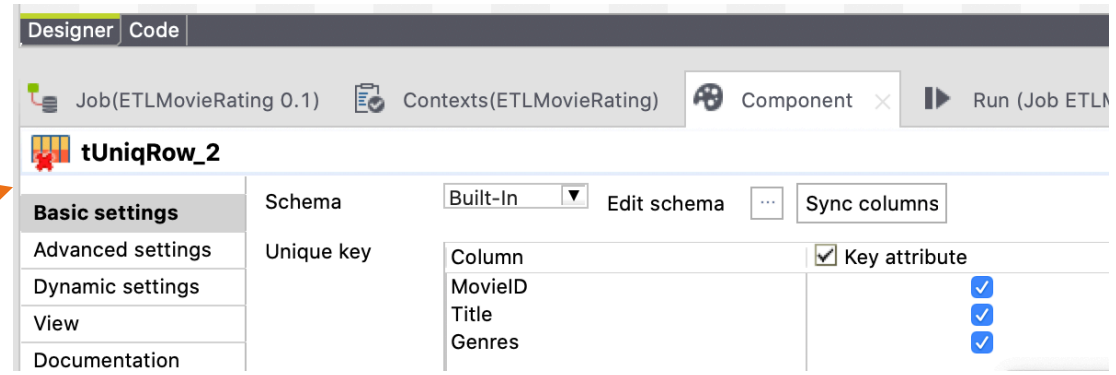
Task5

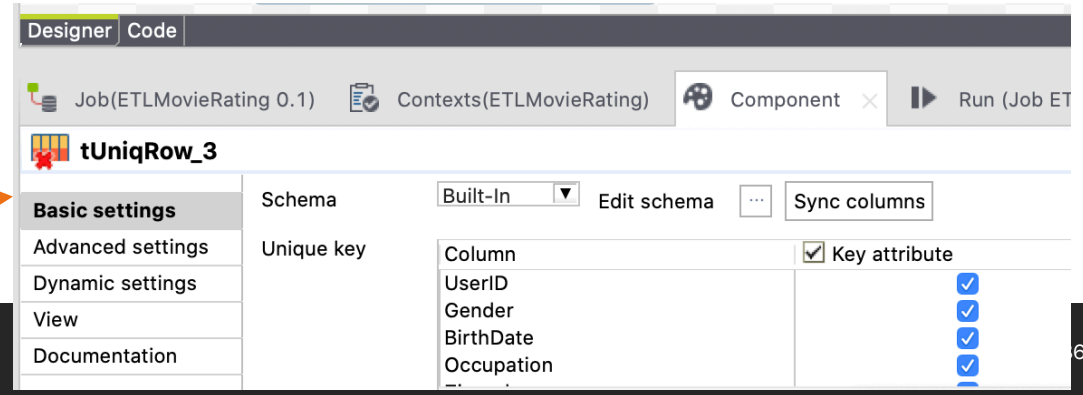
Create TJob#4: Prepare dataset for desired datasets

Step 4.2 Remove Duplicate Records





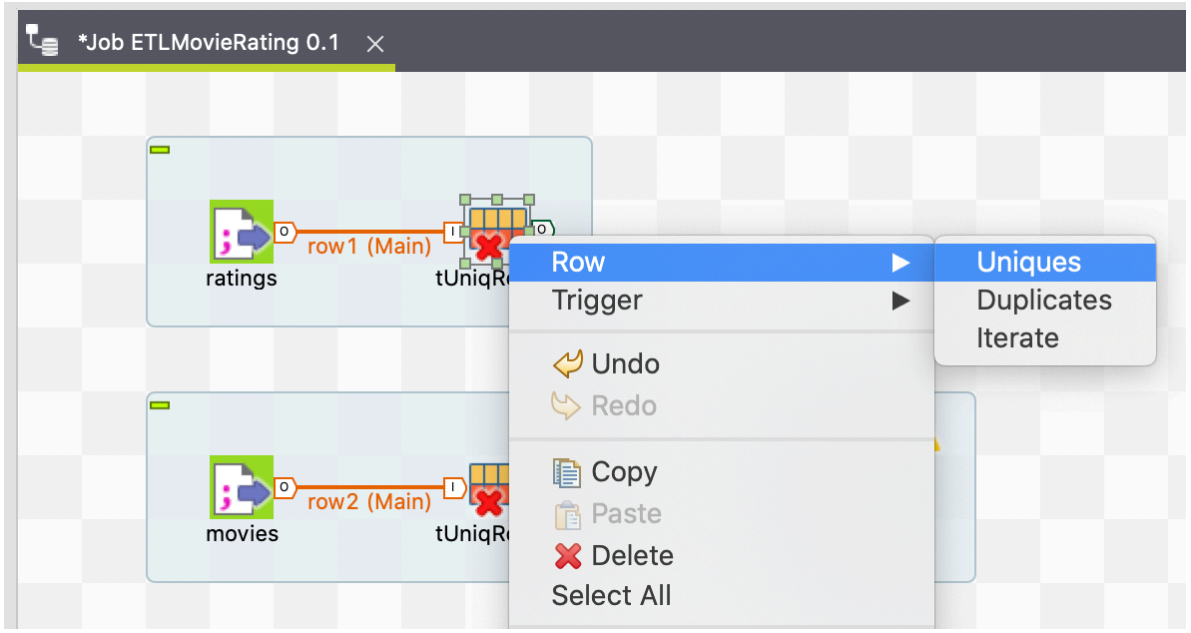


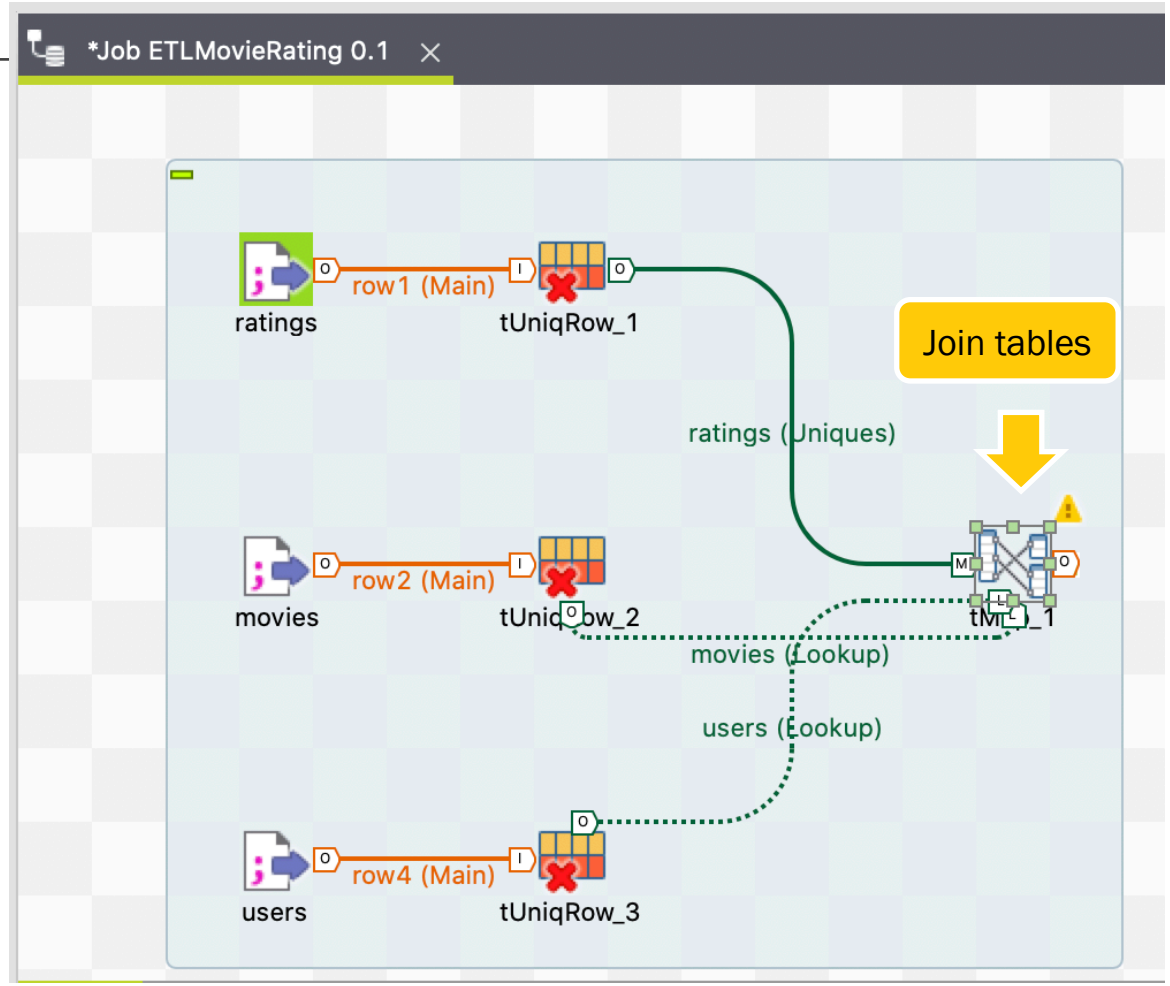


Task5

Create TJob#4: Prepare dataset for desired datasets

Step 4.3 Integrate 3 datasets





Task5

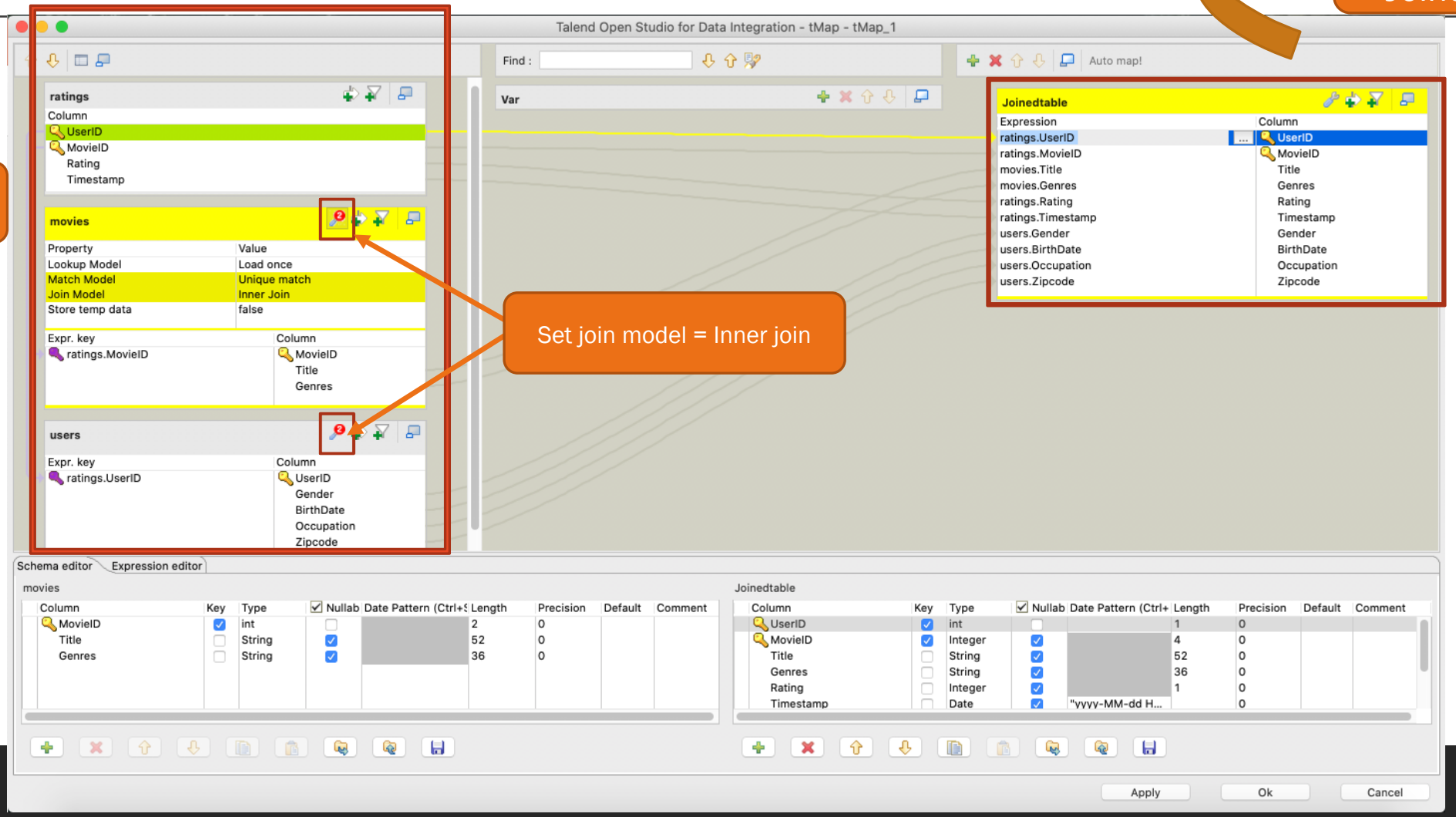
Create TJob#4: Prepare dataset for desired datasets

Step 4.3 Integrate 3 datasets

Output Table "Joinedtable"

Input Tables

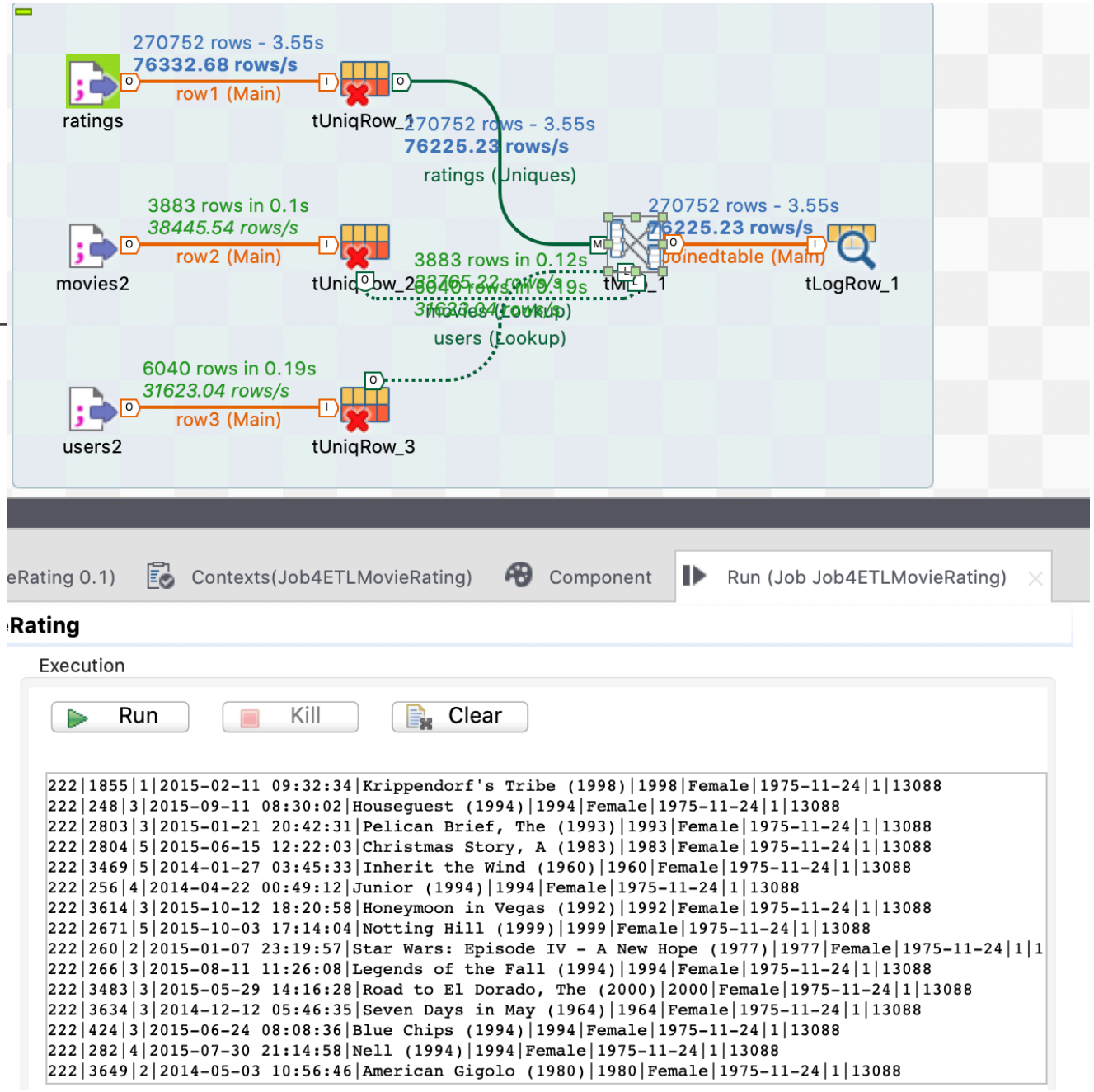
Set join model = Inner join



Task5

Create TJob#4: Prepare dataset for desired datasets

Step 4.3 Integrate 3 datasets
Output of step 4.3 is “Joinedtable”



Task5

Create TJob#4: Prepare dataset for desired datasets

UserID	Gender	BirthDate	Occupation	Zipcode
1	Female	2019-05-10	10	48067
2	Male	1964-11-10	16	70072
3	Male	1995-06-27	15	55117
4	Male	1975-11-07	7	02460
5	Male	1995-05-06	20	55455
6	Female	1970-02-27	9	55117
7	Male	1985-06-05	1	06810
8	Male	1995-05-27	12	11413
9	Male	1995-03-13	17	61614
10	Female	1985-07-29	1	95370
11	Female	1995-10-01	1	04093
12	Male	1995-01-21	12	32793
13	Male	1975-06-18	1	93304
14	Male	1985-06-09	0	60126
15	Male	1995-01-02	7	22903

Users

Step 4.4: Convert birthdate to Age
Step 4.5: Convert Gender to numeric

Transform Data

- Convert birth date to age (numeric values)

- Convert Gender to numeric values (Male=1,Female=2)

Filtering Data

- Filter age more than 7 years

Task5

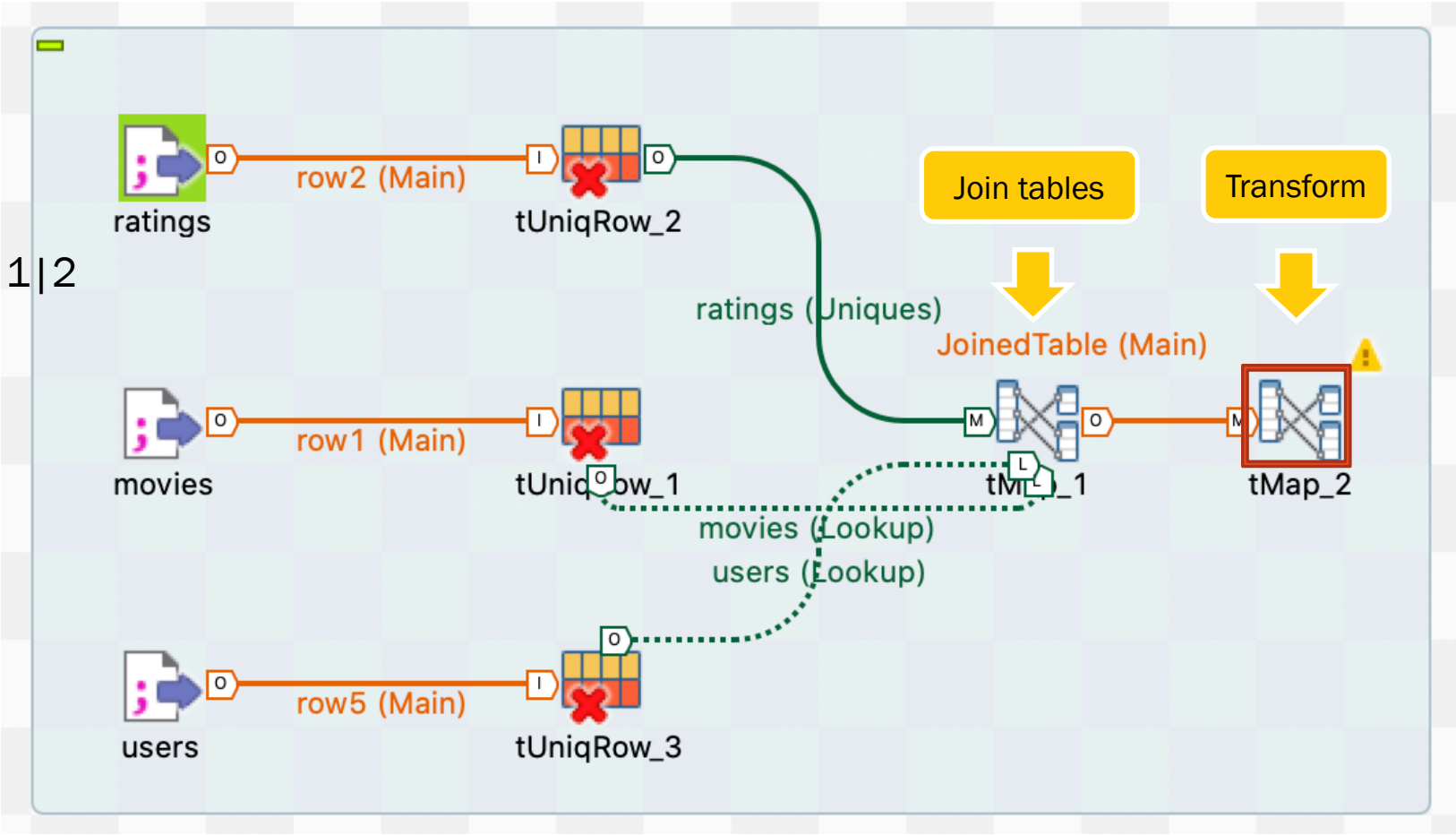
Create TJob#4: Prepare dataset for desired datasets

Transform Data:

Step 4.4: Convert birthdate to Age

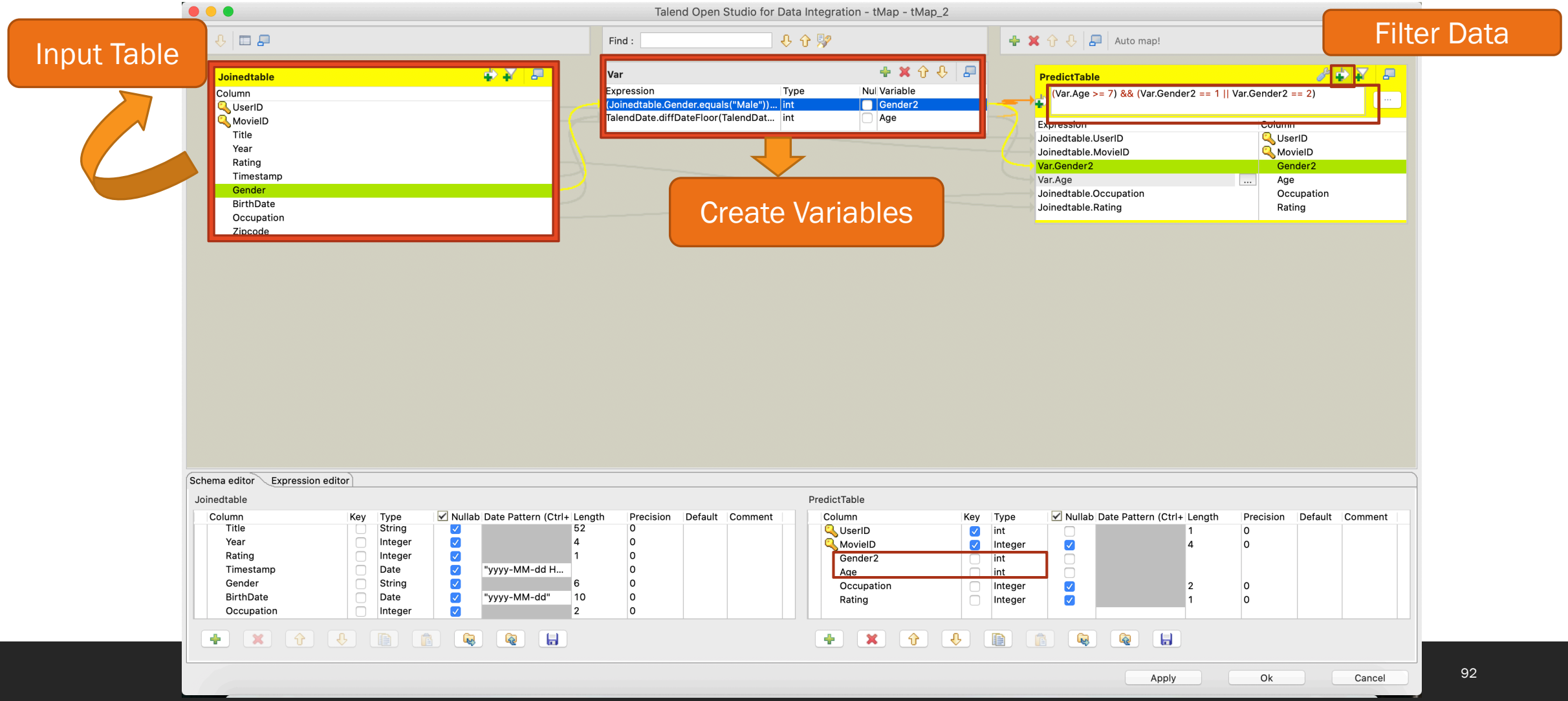
Step 4.5: Convert Gender to numeric

Step 4.6: Filter age >= 7 and Gender = 1|2



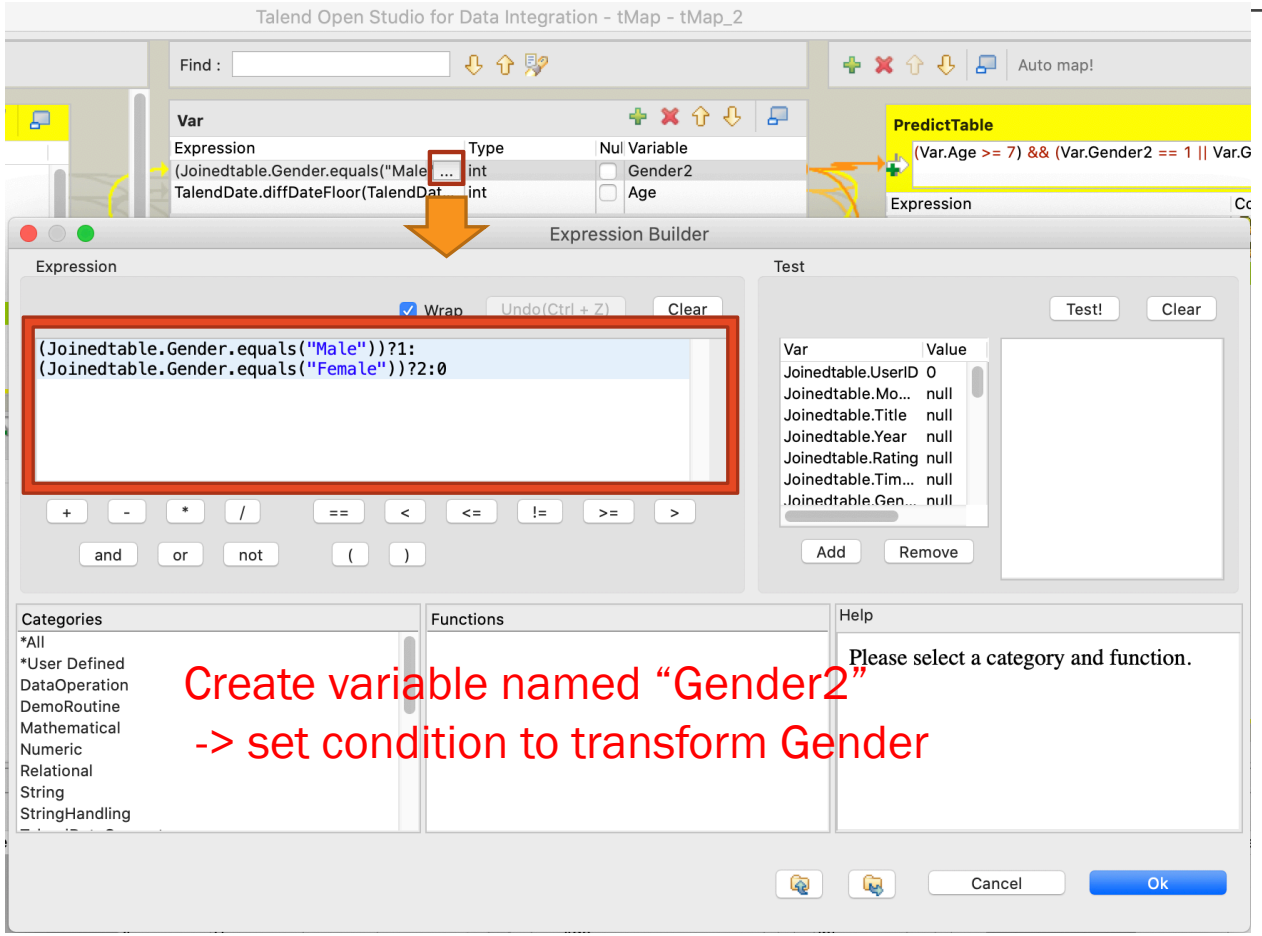
Task5

Create TJob#4: Prepare dataset for desired datasets

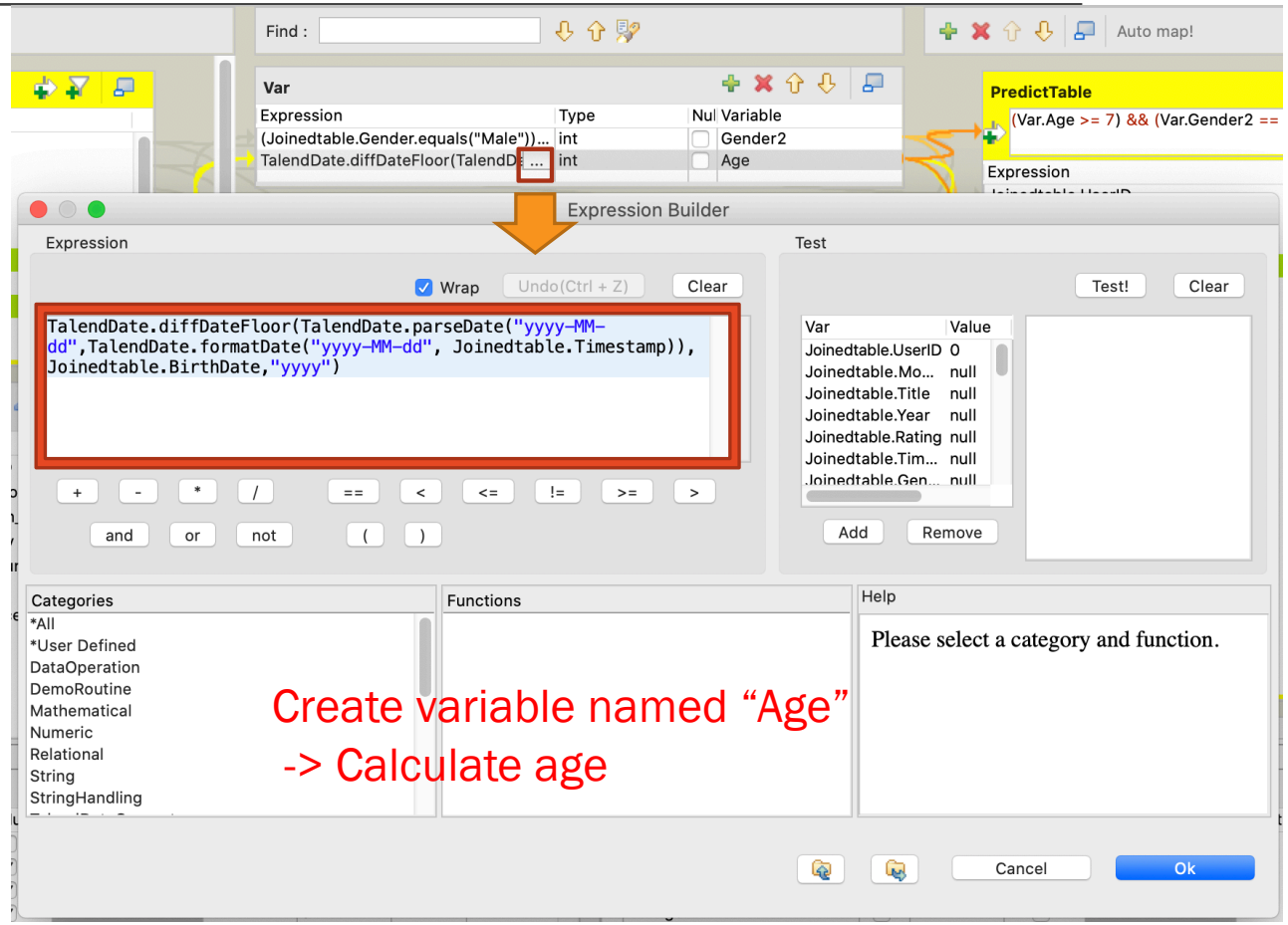


Task5

Create TJob#4: Prepare dataset for desired datasets



Create variable named “Gender2”
-> set condition to transform Gender



Create variable named “Age”
-> Calculate age

Task5

Create TJob#4: Prepare dataset for desired datasets

Create Variables

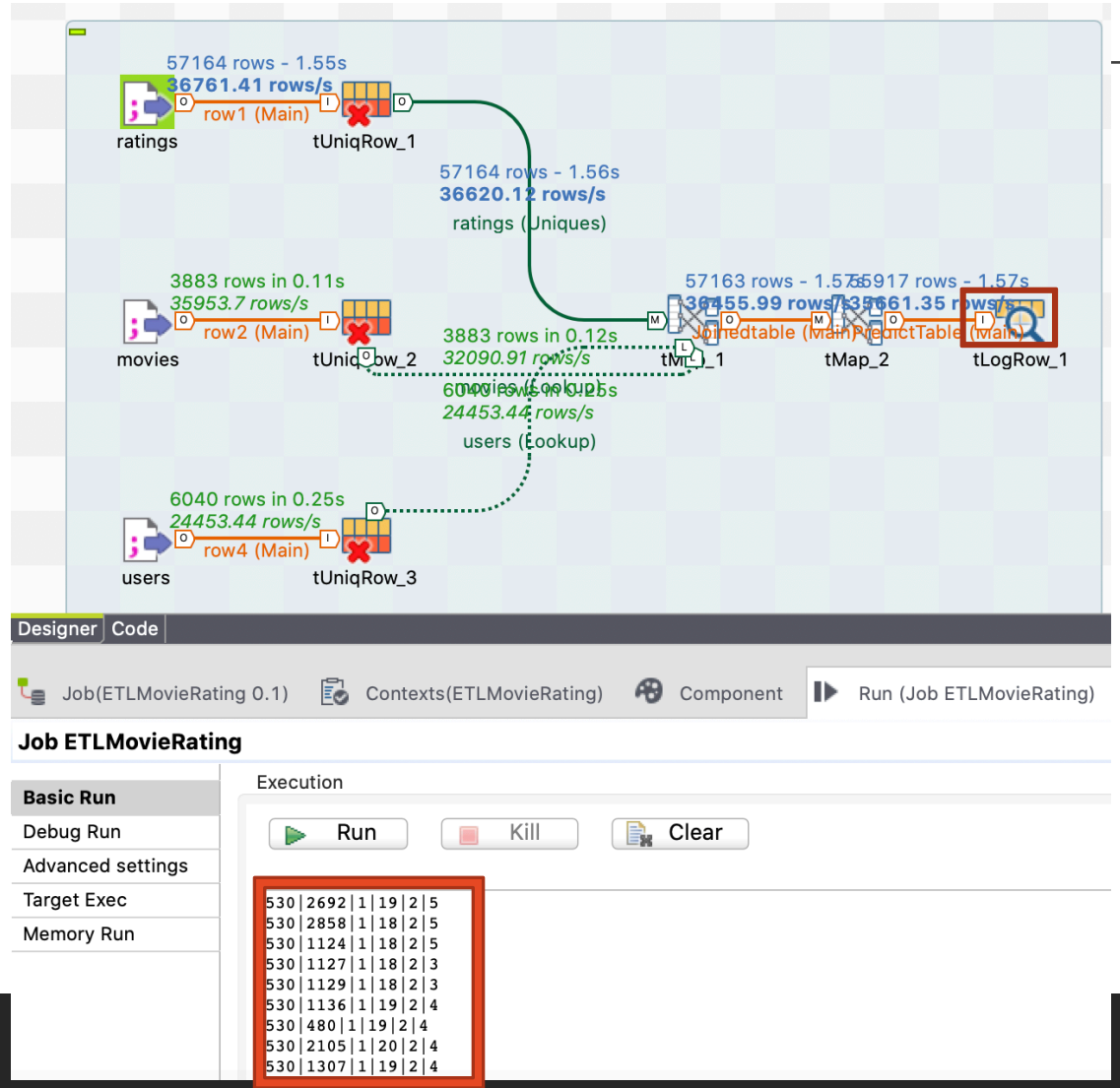
Age(int)	Call java method TalendDate.diffDateFloor(TalendDate.parseDate("yyyy-MM-dd",TalendDate.formatDate("yyyy-MM-dd", Joinedtable.Timestamp)), Joinedtable.BirthDate,"yyyy")
Gender2 (String)	IF/ELSE Statement (Joinedtable.Gender.equals("Male"))?1: (Joinedtable.Gender.equals("Female"))?2:0

Filter Data

```
(Var.Age >= 7) && (Var.Gender2 == 1 || Var.Gender2 == 2)
```

Task5

Create TJob#4: Prepare dataset for desired datasets

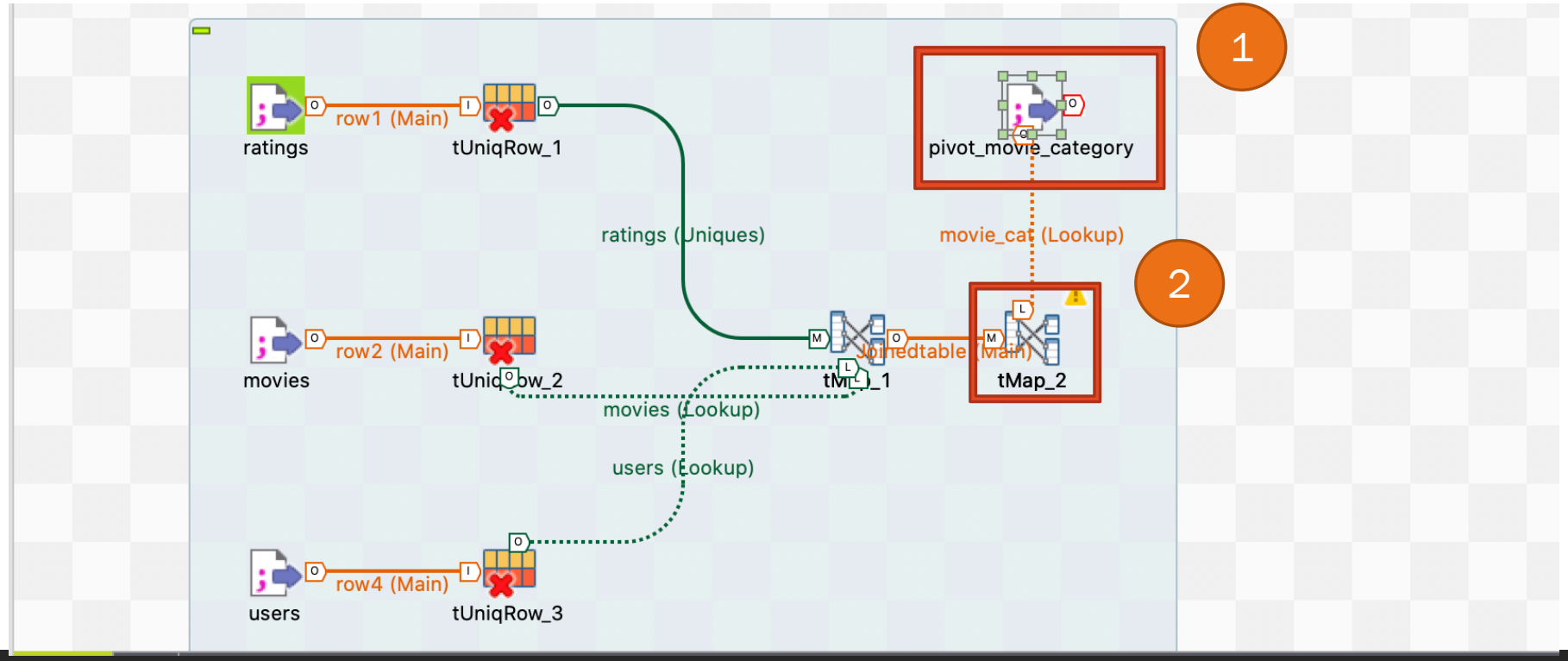


- Add tLogRow to check output
- Example Output:
  - MovieID
  - UserID
  - Gender2
  - Age
  - Occupation
  - Rating

Task5

Create TJob#4: Prepare dataset for desired datasets

- Step 4.7: Merge output of 4.1 and 4.6
  - Add File Delimited Metadata “pivot_movie_category” to merge the output and merge output at tMap_2
  - Double click at tMap_2

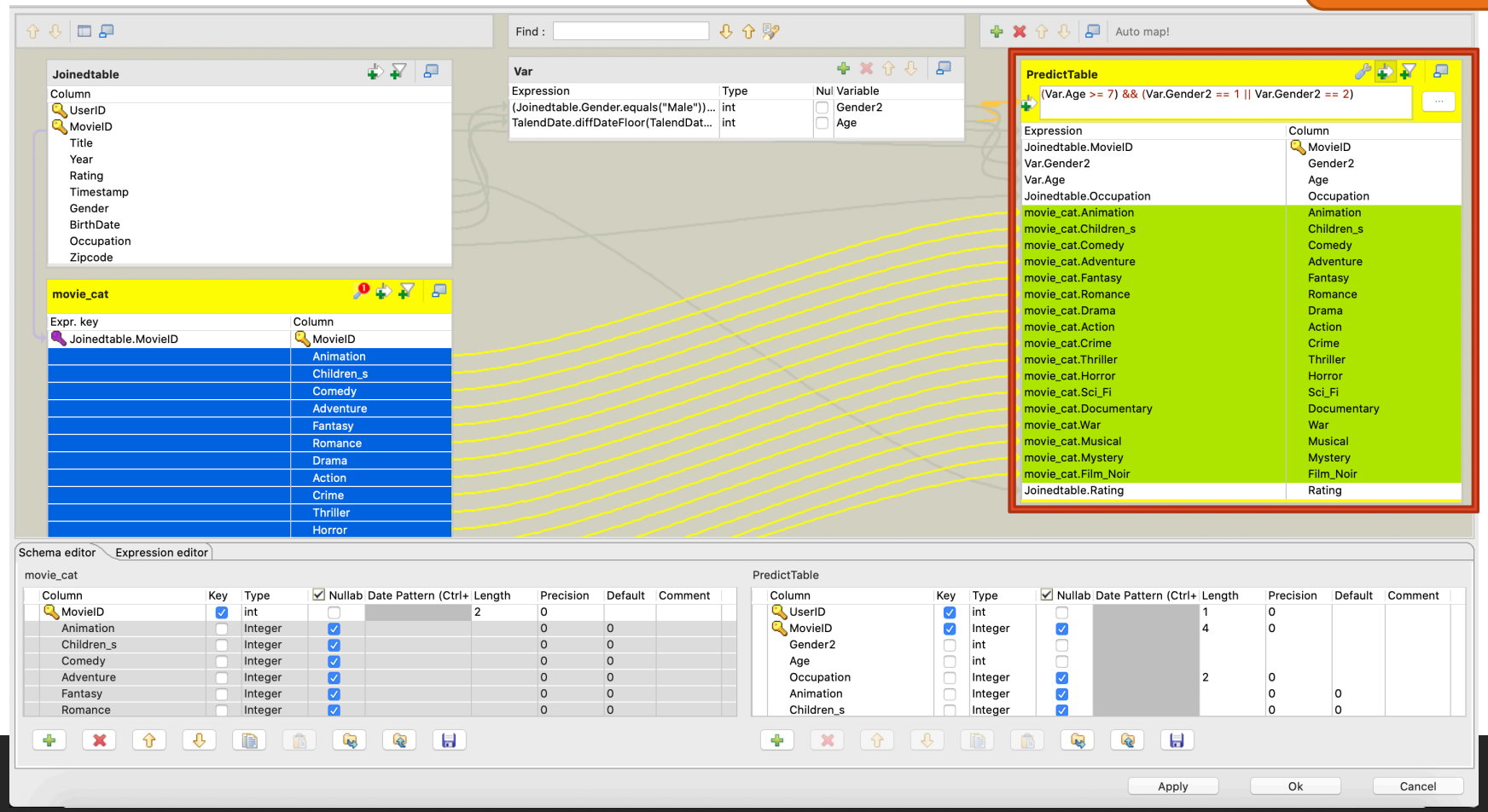


Task5

Create TJob#4: Prepare dataset for desired datasets

- Step 4.7: Merge output of 4.1 and 4.6 for Prediction Model

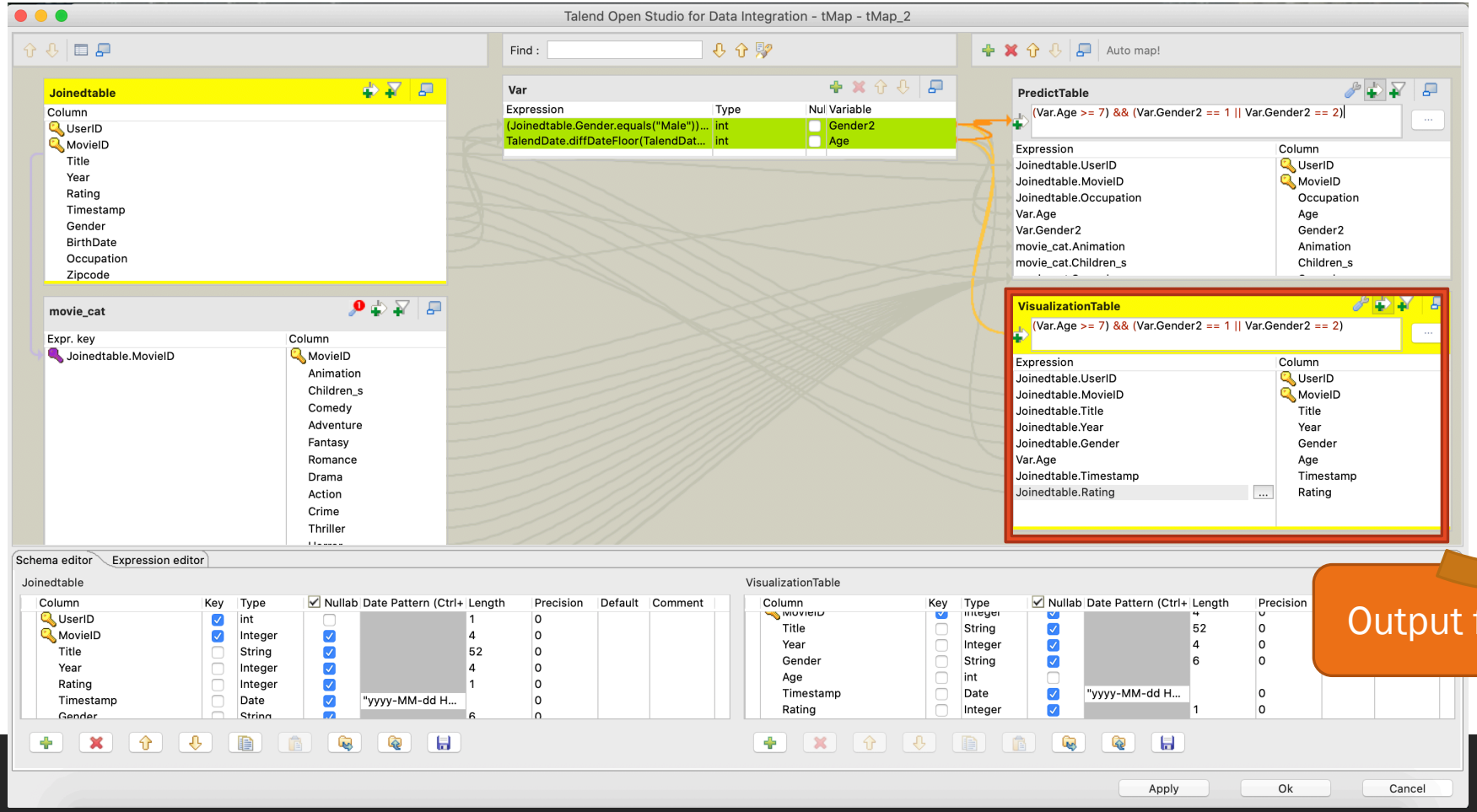
Output for Prediction Model



Task5

Create TJob#4: Prepare dataset for desired datasets

- Step 4.7: Merge output of 4.1 and 4.6 for Visualization



ETL Process

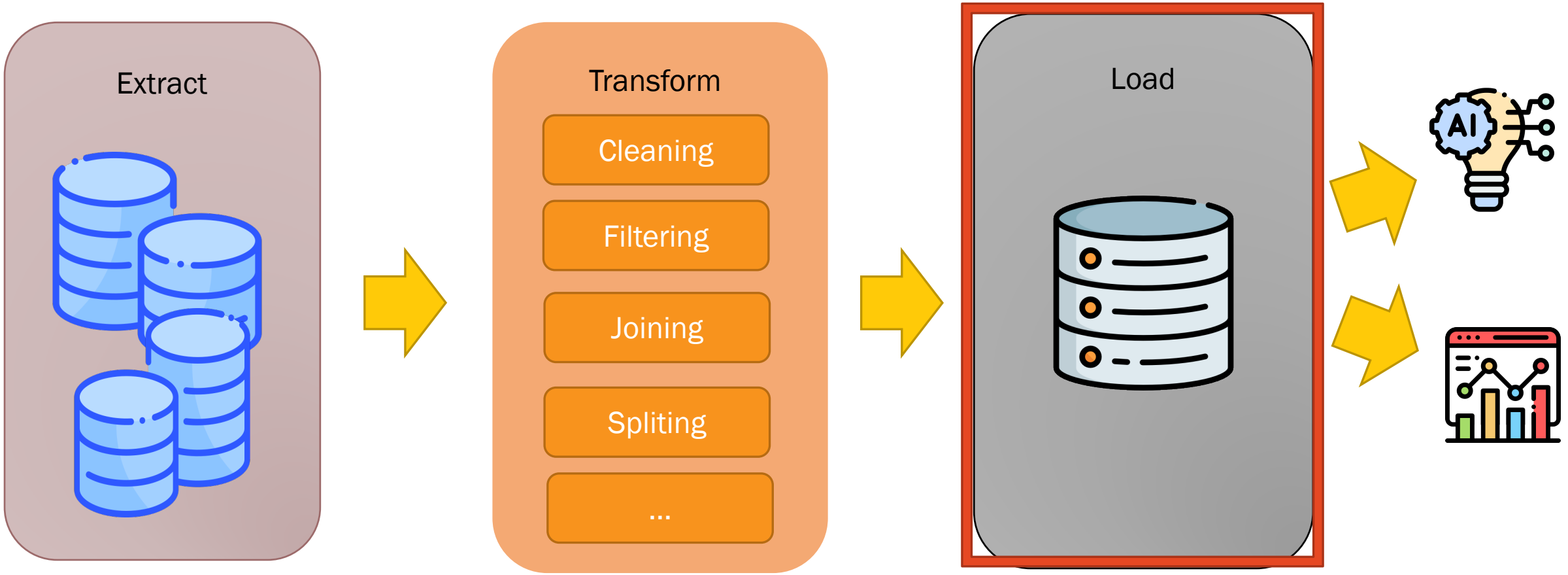
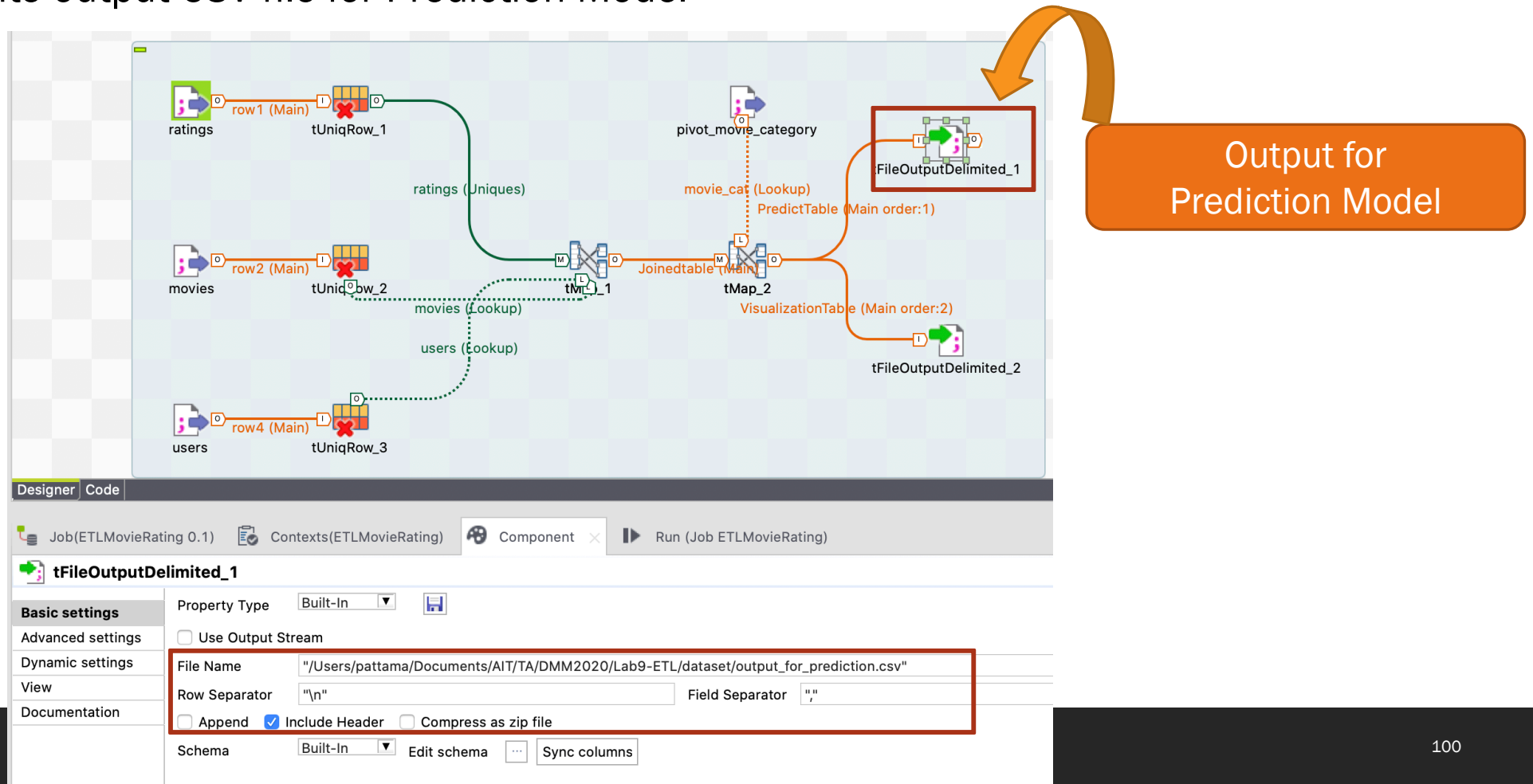


Image source: https://www.flaticon.com

Task5

Create TJob#4: Prepare dataset for desired datasets

- Step 5: Write output CSV file for Prediction Model

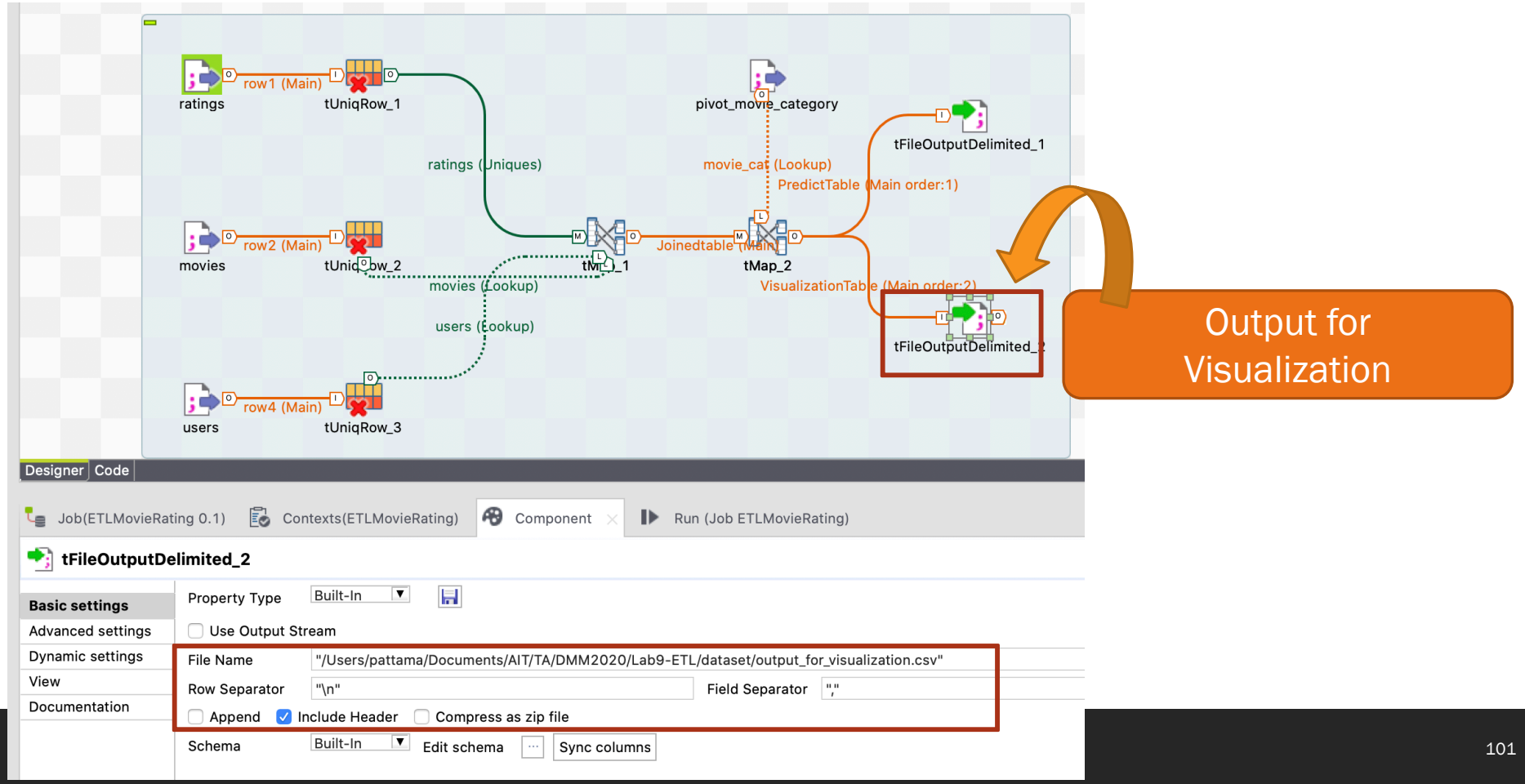


Output for Prediction Model

Task5

Create TJob#4: Prepare dataset for desired datasets

- Step 5: Write output CSV file for Visualization



Output of Task5

1. Dataset for Prediction Model “Movie Rating Prediction ”
2. Dataset for Visualization

Output of Task5

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
UserID	MovieID	Gender2	Age	action	comedy	romantic	fantacy	Fantasy	Romance	Drama	Action_1	Crime	Thriller	Horror	Sci_Fi
2	1357	1	50	0	0	0	0	0	1	1	0	0	0	0	0
2	3068	1	50	0	0	0	0	0	0	1	0	0	0	0	0
2	1537	1	50	0	0	1	0	0	0	0	0	0	0	0	0
2	647	1	49	0	0	0	0	0	0	1	0	0	0	0	0
2	2194	1	50	0	0	0	0	0	0	1	1	1	0	0	0
2	648	1	50	0	0	0	1	0	0	0	1	0	0	0	0
2	2268	1	49	0	0	0	0	0	0	1	0	1	0	0	0
2	2628	1	50	0	0	0	1	1	0	0	1	0	0	0	1
2	1103	1	50	0	0	0	0	0	0	1	0	0	0	0	0
2	2916	1	50	0	0	0	1	0	0	0	1	0	1	0	1
2	3468	1	50	0	0	0	0	0	0	1	0	0	0	0	0
2	1210	1	49	0	0	0	1	0	1	0	1	0	0	0	1
2	1792	1	50	0	0	0	0	0	0	0	1	0	1	0	0
2	1687	1	49	0	0	0	0	0	0	0	1	0	1	0	0
2	1213	1	50	0	0	0	0	0	0	1	0	1	0	0	0
2	3578	1	50	0	0	0	0	0	0	1	1	0	0	0	0
2	2881	1	50	0	0	0	0	0	0	0	1	0	1	0	0
2	3030	1	49	0	0	1	0	0	0	1	0	0	0	0	0
2	1217	1	49	0	0	0	0	0	0	1	0	0	0	0	0
2	3105	1	50	0	0	0	0	0	0	1	0	0	0	0	0
2	434	1	50	0	0	0	1	0	0	0	1	1	0	0	0
2	2126	1	49	0	0	0	0	0	0	0	1	1	1	0	0
2	3107	1	50	0	0	0	0	0	0	1	1	0	0	0	0
2	3108	1	50	0	0	1	0	0	1	1	0	0	0	0	0

Output for Prediction Model

Output of Task5

A	B	C	D	E	F	G	H	I
UserID	MovieID	Title	Year	Gender	Gender2	Age	Timestamp	Rating
2	1357	Shine (1996)	1996	Male	1	50	2015-10-18 15:24:38	5
2	3068	Verdict, The (1982)	1982	Male	1	50	2015-01-14 15:31:09	4
2	1537	Shall We Dance? (Shall We Dansu?) (1996)	1996	Male	1	50	2015-05-22 09:03:30	4
2	647	Courage Under Fire (1996)	1996	Male	1	49	2014-02-21 07:02:49	3
2	2194	Untouchables, The (1987)	1987	Male	1	50	2015-10-25 03:23:12	4
2	648	Mission: Impossible (1996)	1996	Male	1	50	2015-01-07 10:58:19	4
2	2268	Few Good Men, A (1992)	1992	Male	1	49	2014-05-27 15:12:42	5
2	2628	Star Wars: Episode I - The Phantom Menace (1	1999	Male	1	50	2015-11-03 14:11:03	3
2	1103	Rebel Without a Cause (1955)	1955	Male	1	50	2015-06-07 05:36:56	3
2	2916	Total Recall (1990)	1990	Male	1	50	2015-09-19 18:07:05	3
2	3468	Hustler, The (1961)	1961	Male	1	50	2015-08-17 09:42:45	5
2	1210	Star Wars: Episode VI - Return of the Jedi (198	1983	Male	1	49	2014-02-09 14:01:03	4
2	1792	U.S. Marshalls (1998)	1998	Male	1	50	2015-07-02 00:47:44	3
2	1687	Jackal, The (1997)	1997	Male	1	49	2014-05-11 10:57:57	3
2	1213	GoodFellas (1990)	1990	Male	1	50	2014-11-28 21:19:25	2
2	3578	Gladiator (2000)	2000	Male	1	50	2015-06-24 00:51:45	5
2	2881	Double Jeopardy (1999)	1999	Male	1	50	2014-12-29 21:27:23	3
2	3030	Yojimbo (1961)	1961	Male	1	49	2014-10-09 16:27:37	4
2	1217	Ran (1985)	1985	Male	1	49	2014-03-12 16:41:31	3
2	3105	Awakenings (1990)	1990	Male	1	50	2015-10-17 05:59:39	4
2	434	Cliffhanger (1993)	1993	Male	1	50	2015-10-03 23:12:06	2
2	2126	Snake Eyes (1998)			1	49	2014-02-04 15:38:02	3

Output for Visualization

References

1. https://www.talend.com/resources/discovering-talend-studio/
2. https://www.tutorialspoint.com/talend/index.htm


Thank you.