

Atelier 1 : ETL Talend et MySQL

Les objectifs de cet atelier sont les suivants :

1. **Compréhension de l'ETL (Extract, Transform, Load)** : Les participants devraient comprendre le concept général de l'ETL et son importance dans le processus d'intégration de données. Cela inclut la compréhension des différentes étapes du processus ETL, à savoir l'extraction des données à partir de sources variées, leur transformation selon les besoins métier spécifiques, et enfin leur chargement dans une base de données cible.
2. **Maîtrise de Talend** : Les participants devraient acquérir une connaissance pratique de Talend en tant qu'outil d'intégration de données. Cela implique de comprendre l'interface de Talend Studio, la manière de créer et de gérer des projets, ainsi que la manipulation des composants et des connexions pour concevoir des flux ETL efficaces.
3. **Manipulation des composants** : Les participants devraient apprendre à utiliser des composants spécifiques tels que TMap, LogRow et MySQLInput dans le cadre de la création de flux ETL. Cela comprend la compréhension des fonctionnalités de chaque composant, leur configuration appropriée et leur intégration dans des flux de données complexes.
4. **Gestion des erreurs et débogage** : Les participants devraient être capables de gérer les erreurs et les exceptions qui peuvent survenir lors de l'exécution des flux ETL. Cela implique l'utilisation de techniques de débogage telles que l'enregistrement des données à différentes étapes du processus, la gestion des exceptions et la mise en place de stratégies de récupération en cas d'échec.

1. Télécharger Talend

- https://perso.isima.fr/~jofontan/Install/TOS_DI-Win32-20200219_1130-V7.3.1.exe

2. Préparer une instance MySQL avec les données sources

Assurez-vous d'avoir le fichier SQL `gestion_livres_init.sql` dans le même répertoire que votre fichier `docker-compose.yml`.

Pour exécuter ce fichier `docker-compose.yml`, suivez ces étapes :

- a) Placez votre `docker-compose.yml` et vos fichiers SQL `gestion_livres_init.sql` dans un même répertoire.

```
CREATE DATABASE IF NOT EXISTS `gestion_livres` DEFAULT CHARACTER SET utf8;
USE `gestion_livres`;

-- Table `auteur`
DROP TABLE IF EXISTS `auteur`;
CREATE TABLE `auteur` (
  `NUMERO_A` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `NOM` varchar(450) DEFAULT NULL,
  `PRENOM` varchar(450) DEFAULT NULL,
  `DOMICILE` varchar(450) DEFAULT NULL,
  PRIMARY KEY (`NUMERO_A`)
) ENGINE=InnoDB AUTO_INCREMENT=8547586 DEFAULT CHARSET=latin1;

-- Données pour table `auteur`
INSERT INTO `auteur` VALUES
(8547, 'Chambord', 'Emilie', 'Nice'),
(52136, 'Fabiere', 'Sylvie', 'Bordeaux'),
(78545, 'TINTIN', 'Thierry', 'Clermont'),
(85478, 'Castafiore', 'Emilie', 'Paris'),
(542536, 'Dupont', 'Pierre', 'Avignon'),
(8547585, 'Momo', 'Roland', 'Toulouse');

-- Table `livre`
DROP TABLE IF EXISTS `livre`;
CREATE TABLE `livre` (
  `NUMERO_L` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `NUMERO_A` int(10) unsigned NOT NULL,
  `TITRE` varchar(128) DEFAULT NULL,
  `PRIXT` float DEFAULT NULL,
  PRIMARY KEY (`NUMERO_L`)
) ENGINE=InnoDB AUTO_INCREMENT=11324 DEFAULT CHARSET=latin1;

-- Données pour table `livre`
INSERT INTO `livre` VALUES
(10101, 85478, 'aaaaaa', 10),
(11111, 8547, 'eeeeee', 54),
(11122, 52136, 'ffffff', 12),
(11231, 542536, 'bbbbbb', 34),
(11321, 8547585, 'eeeeeee', 29),
(11323, 8547585, 'cccccccc', 45);
```

- b) Ouvrez un terminal dans ce répertoire.
c) Exécutez la commande suivante pour démarrer les services définis dans le fichier `docker-compose.yml` :

```
version: '3.8'

services:
  mysql_db:
    image: mysql:latest
    container_name: mysql_db
    ports:
      - "3306:3306"
    environment:
      MYSQL_ROOT_PASSWORD: mysecret
      MYSQL_DATABASE: gestion_livres
    volumes:
      - ./gestion_livres_init.sql:/docker-entrypoint-initdb.d/gestion_livres_init.sql
    restart: always # Ajout de cette ligne pour redémarrer le conteneur en cas d'échec

networks:
  - mysql_network
```

```
networks:
  mysql_network:
    driver: bridge
```

```
networks:
  mysql_network:
    driver: bridge
```

- d) Taper la commande suivante pour lancer les services
docker compose up -d
- e) Pour stopper et supprimer les services taper la commande suivante :
Docker compose down



Talend for Data Integration

Trainer : Sellami Mokhtar
mokhtar.sellami@gmail.com



```
mysql> UNLOCK TABLES;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> show tables;
+-----+
| Tables_in_gestion_livres |
+-----+
| auteur                   |
| livre                    |
+-----+
2 rows in set (0.00 sec)

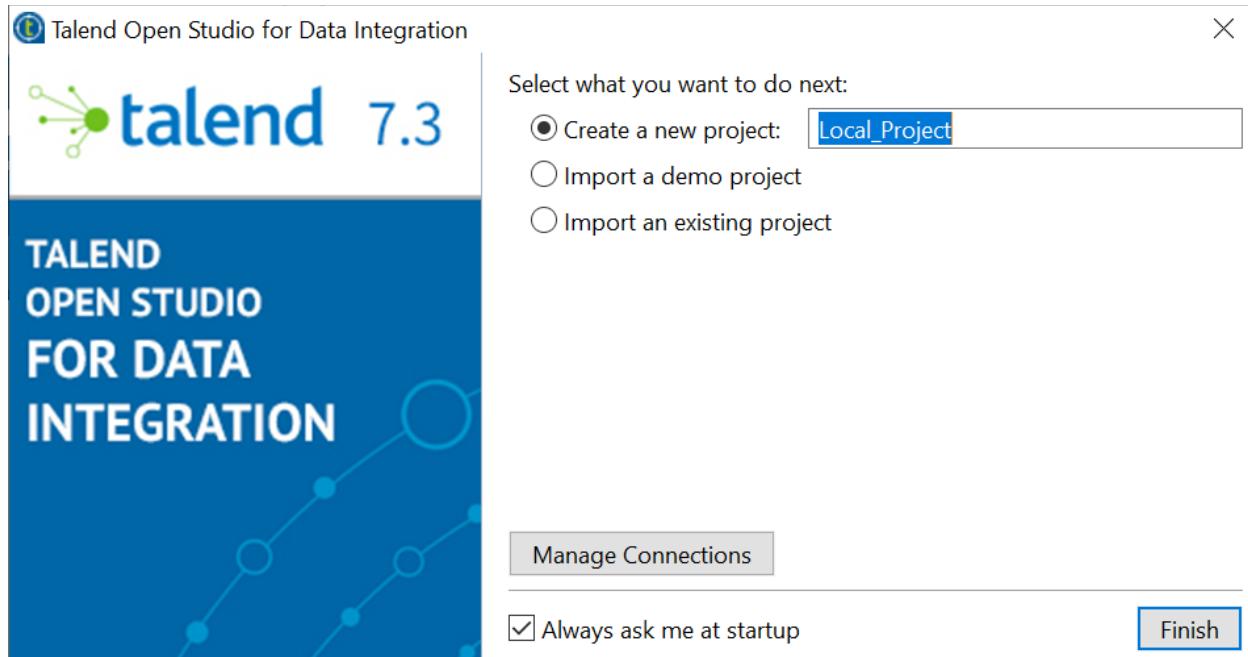
mysql> select * from livre
-> ;
+-----+
| NUMERO_L | NUMERO_A | TITRE | PRIX |
+-----+
| 10181   | 85478   | aaaaaa | 10 |
| 11111   | 8547   | eeeee | 54 |
| 11122   | 52136   | ffffff | 12 |
| 11231   | 542536  | bbbbb | 34 |
| 11321   | 8547585 | efffff | 29 |
| 11323   | 8547585 | cccddd | 45 |
+-----+
6 rows in set (0.00 sec)

mysql> select * from auteur
-> ;
+-----+
| NUMERO_A | NOM     | PRENOM | DOMICILE |
+-----+
| 8547   | Chambord | Emilie | Nice      |
| 52136  | Fablier  | Sylvie | Bordeaux  |
| 78545  | TINTIN   | Thierry | Clermont  |
| 85478  | Castafiore | Emilie | Paris      |
| 542536  | Dupont   | Pierre | Avignon   |
| 8547585 | Homo    | Roland | Toulouse  |
+-----+
6 rows in set (0.00 sec)

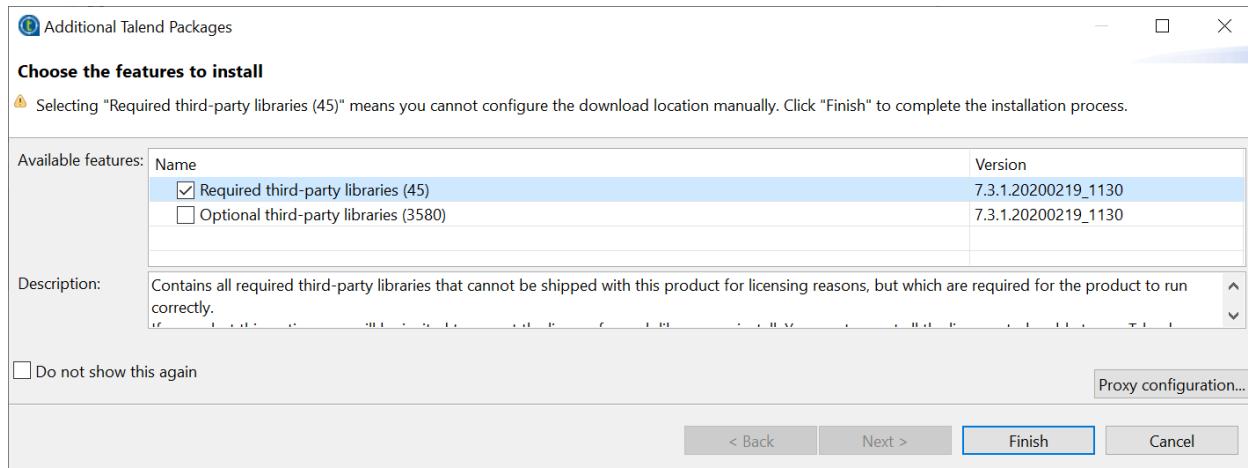
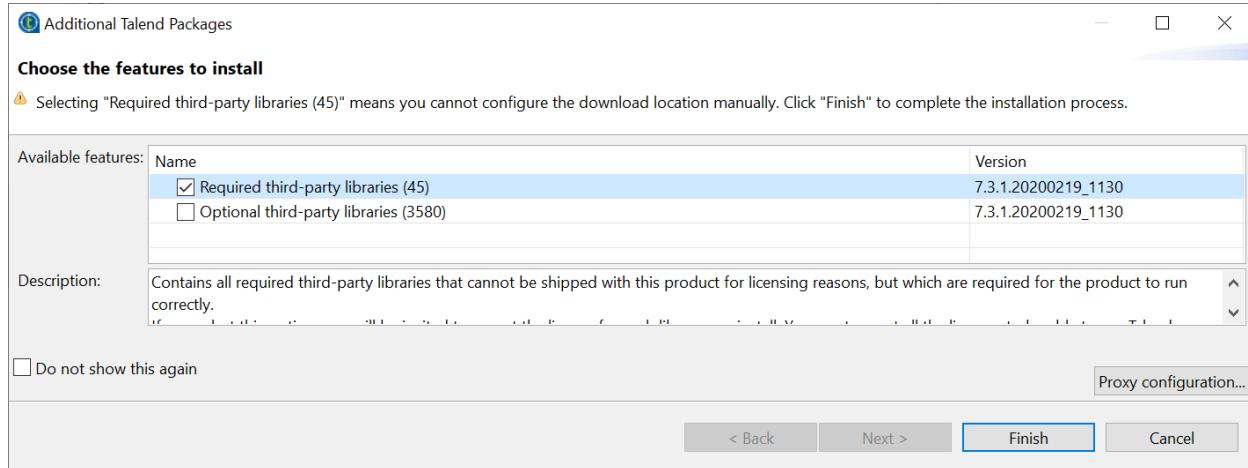
mysql>
```

Engine running RAM 4.59 GB CPU 0.06% Signed in v4.27.2

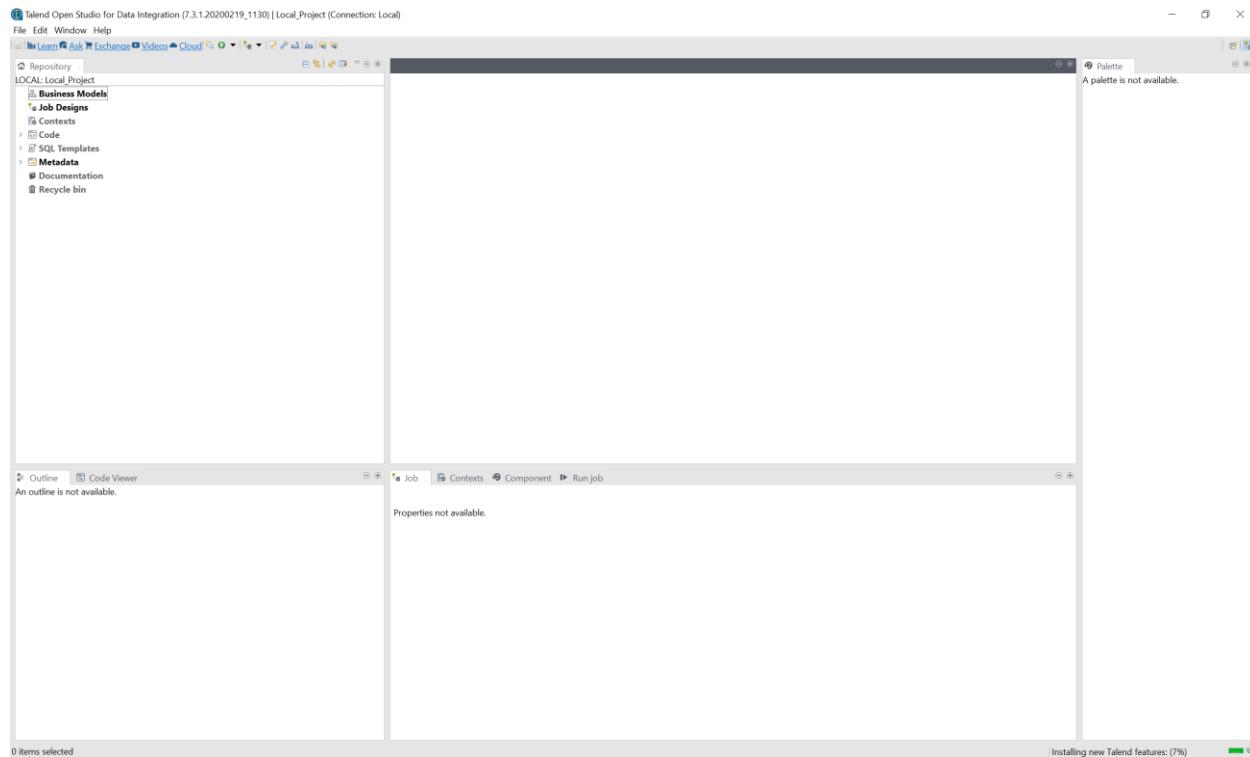
3. Créer un projet



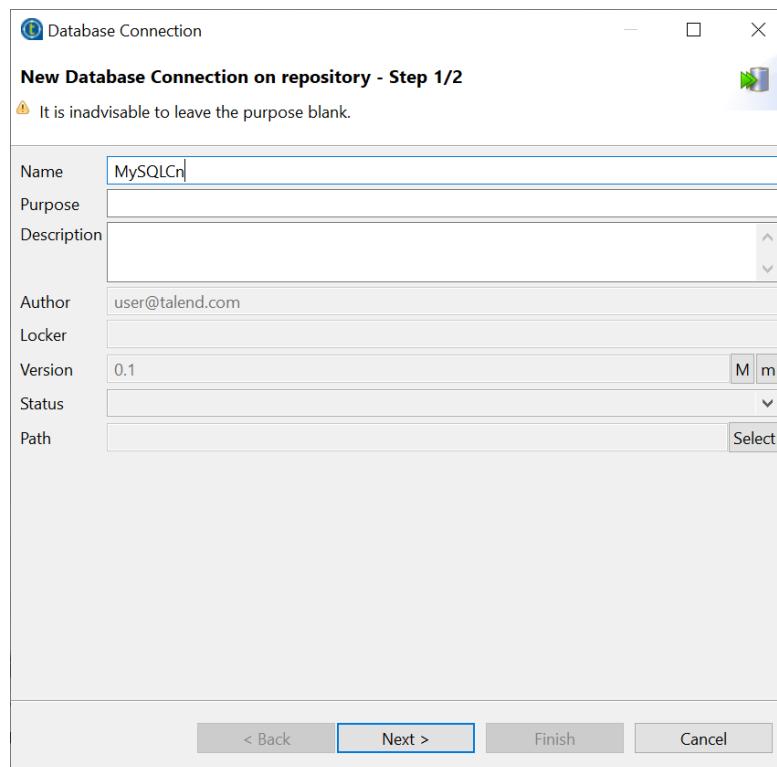
4. Installer les fonctionnalités avancées



5. Interface d'accueil de Talend



6. Création d'une connexion à une base MySQL :



Database Connection

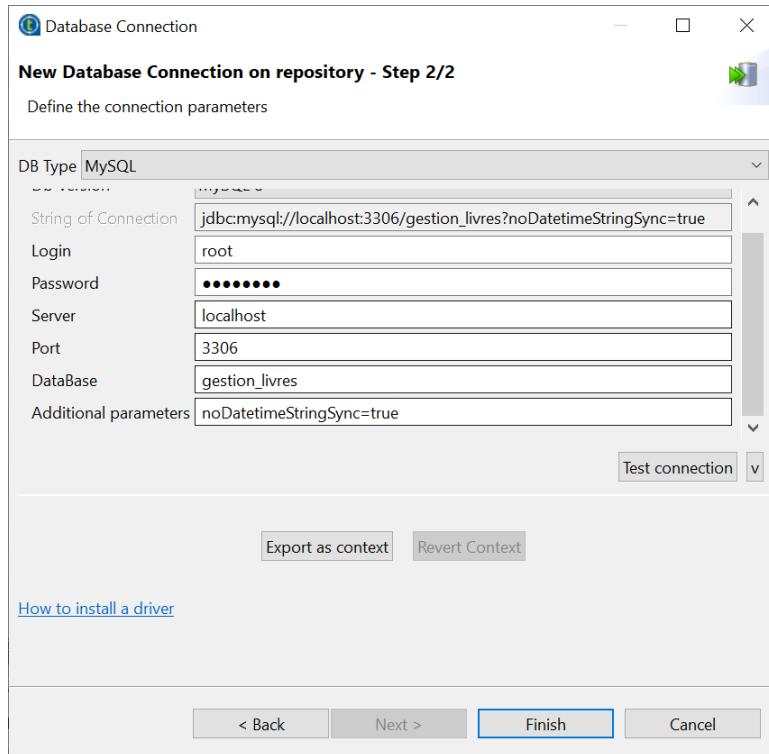
New Database Connection on repository - Step 1/2

⚠ It is inadvisable to leave the purpose blank.

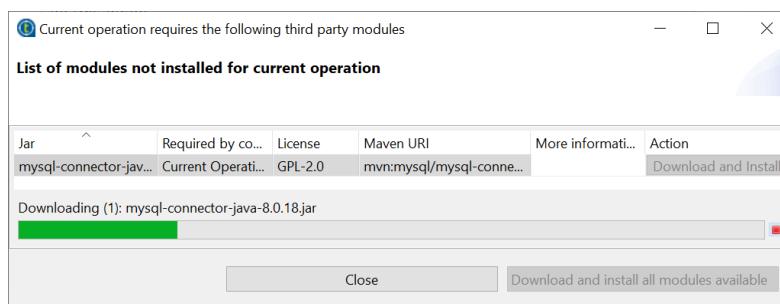
Name	MySQLCn
Purpose	
Description	
Author	user@talend.com
Locker	
Version	0.1
Status	
Path	

< Back **Next >** Finish Cancel

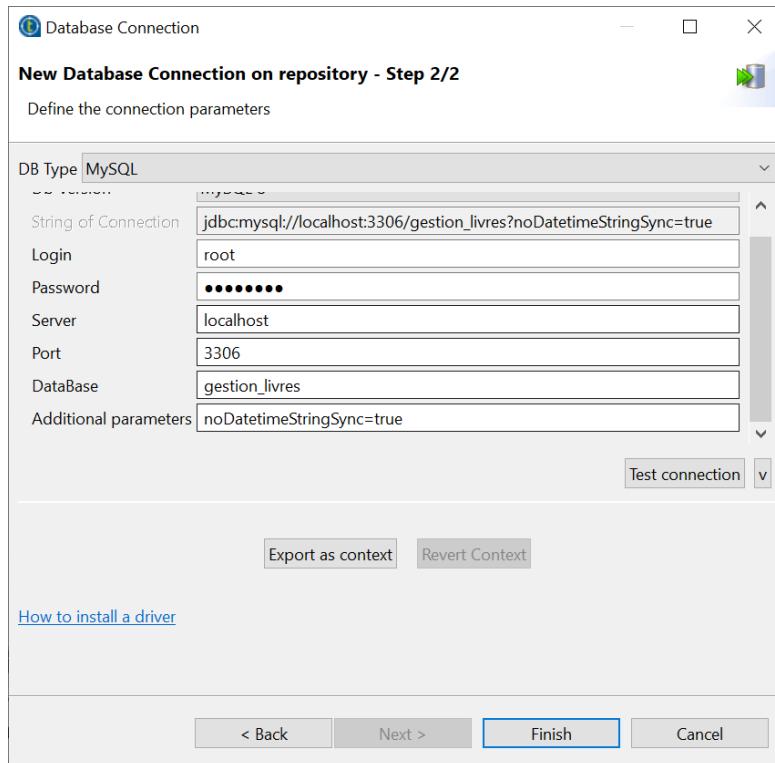
7. Configurer les paramètres de connexion à la base de données



8. Installer les jars nécessaires

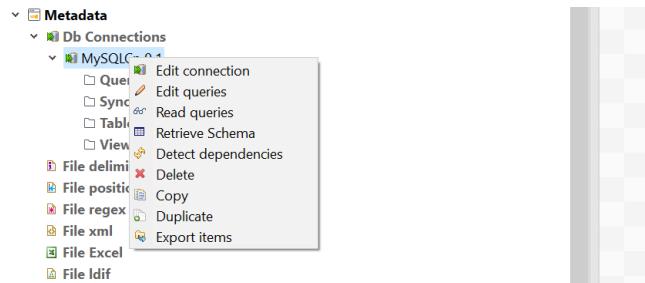


9. Tester la connexion

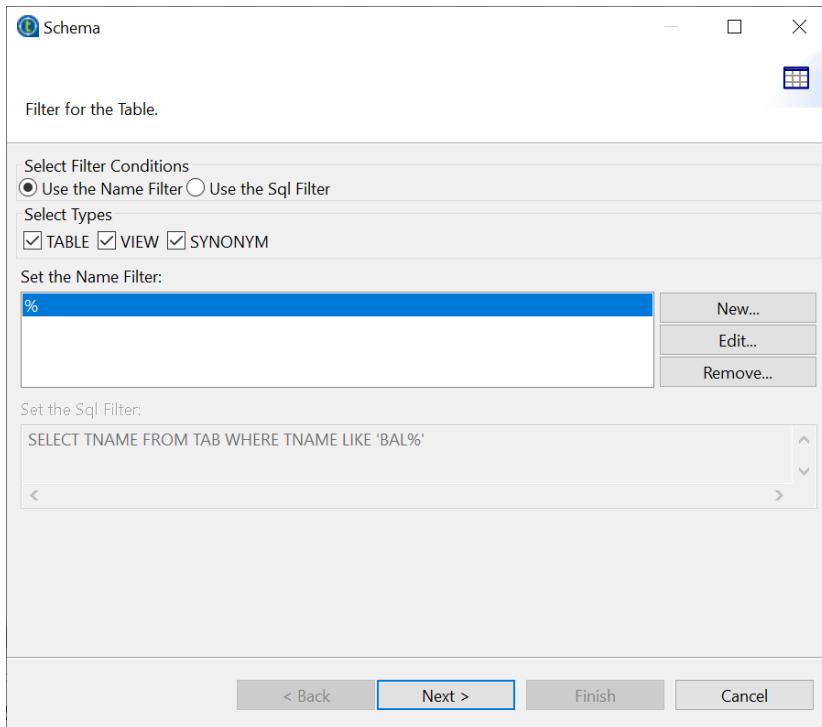


10. Création d'une connexion à une base MySQL :

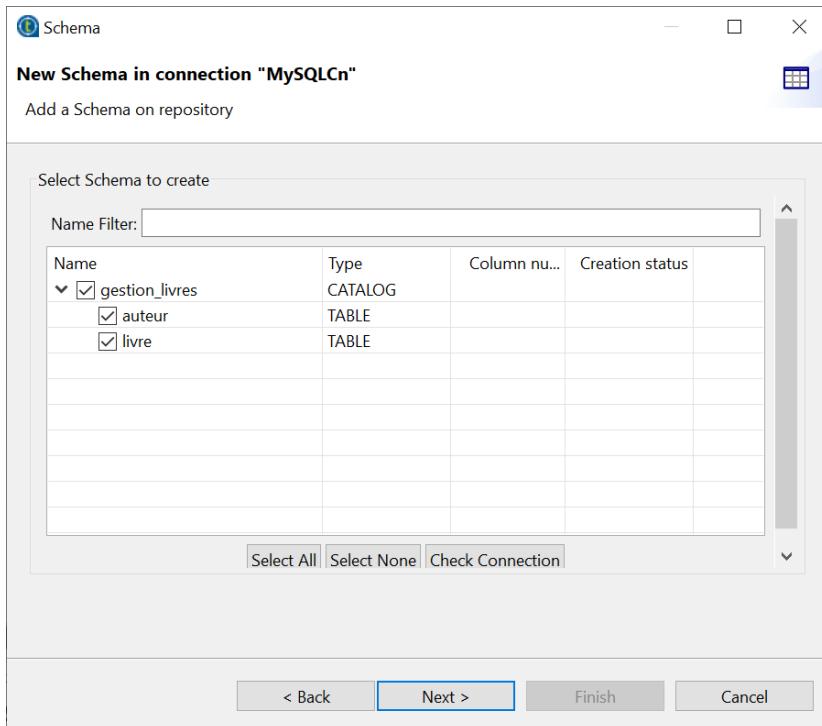
– Récupération du schéma des tables



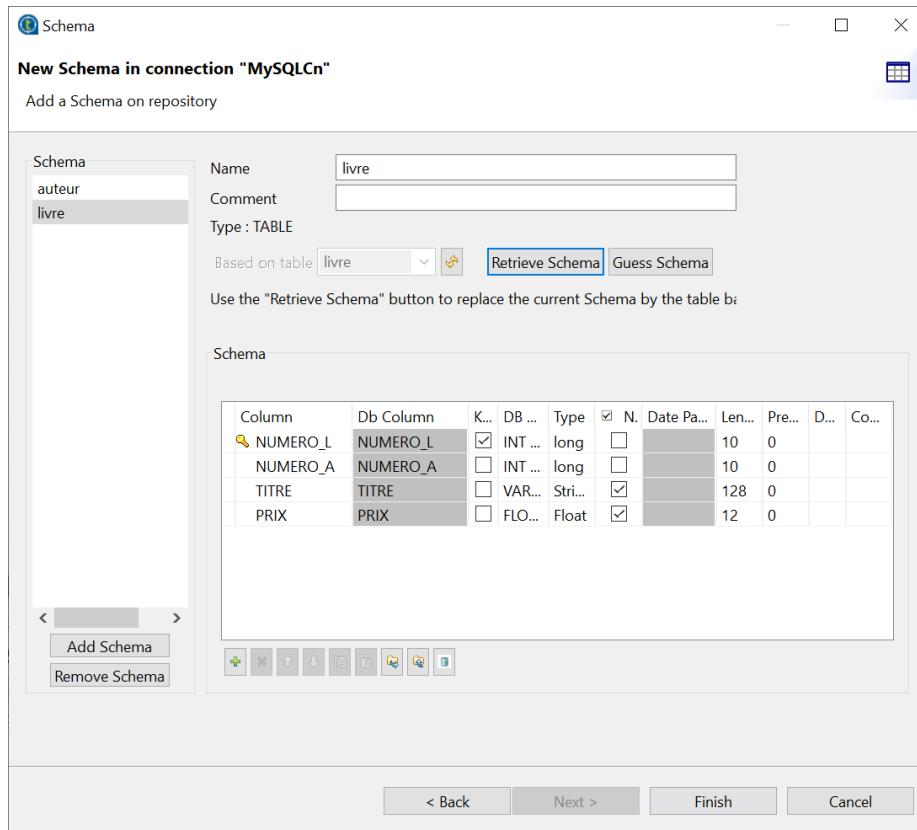
11. Cocher Table et cliquer « Next »



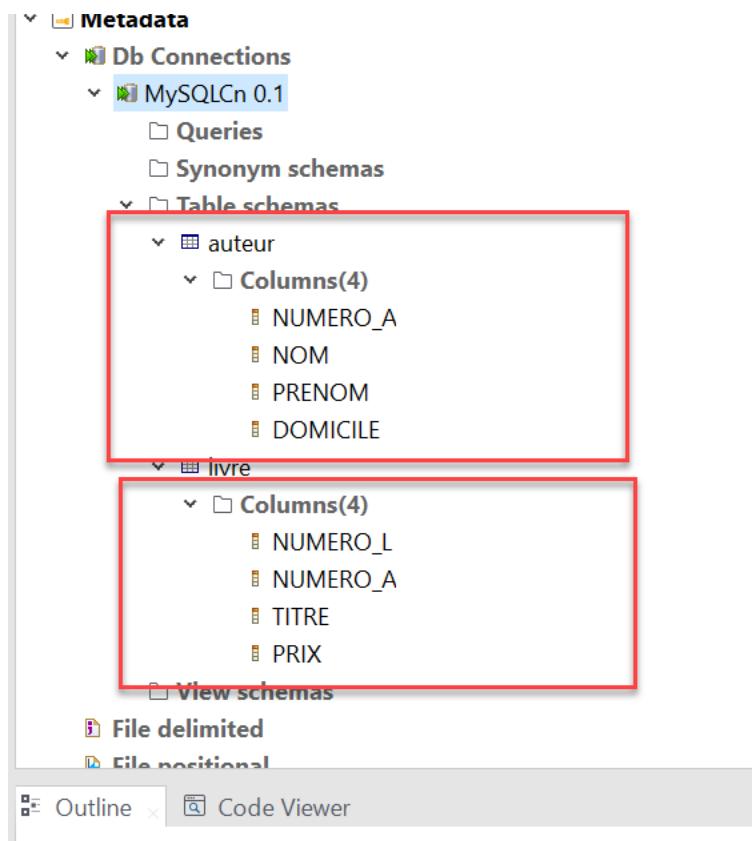
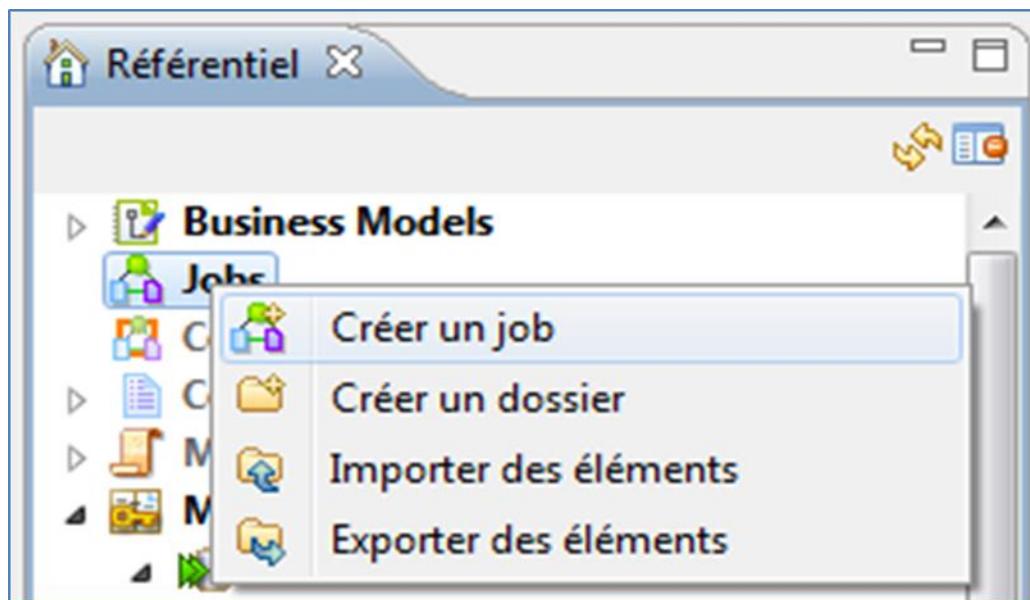
12. Sélectionner la base de données « gestion_livres »



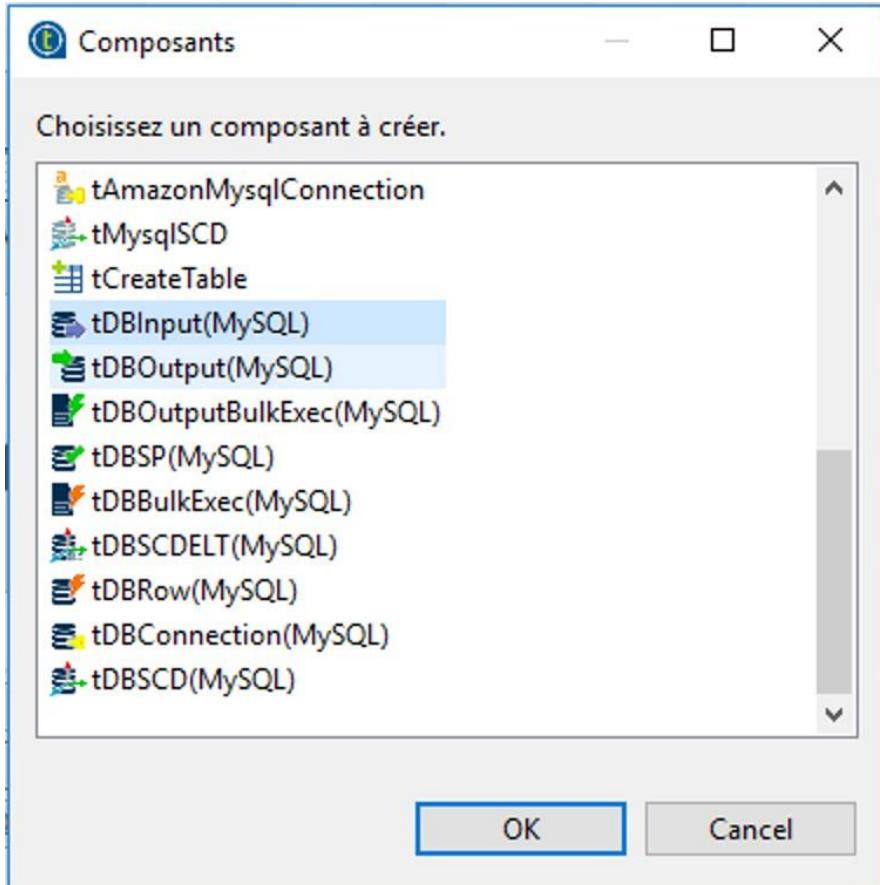
13. Choisir les colonnes désirées en cas de besoins et cliquer « Finish »

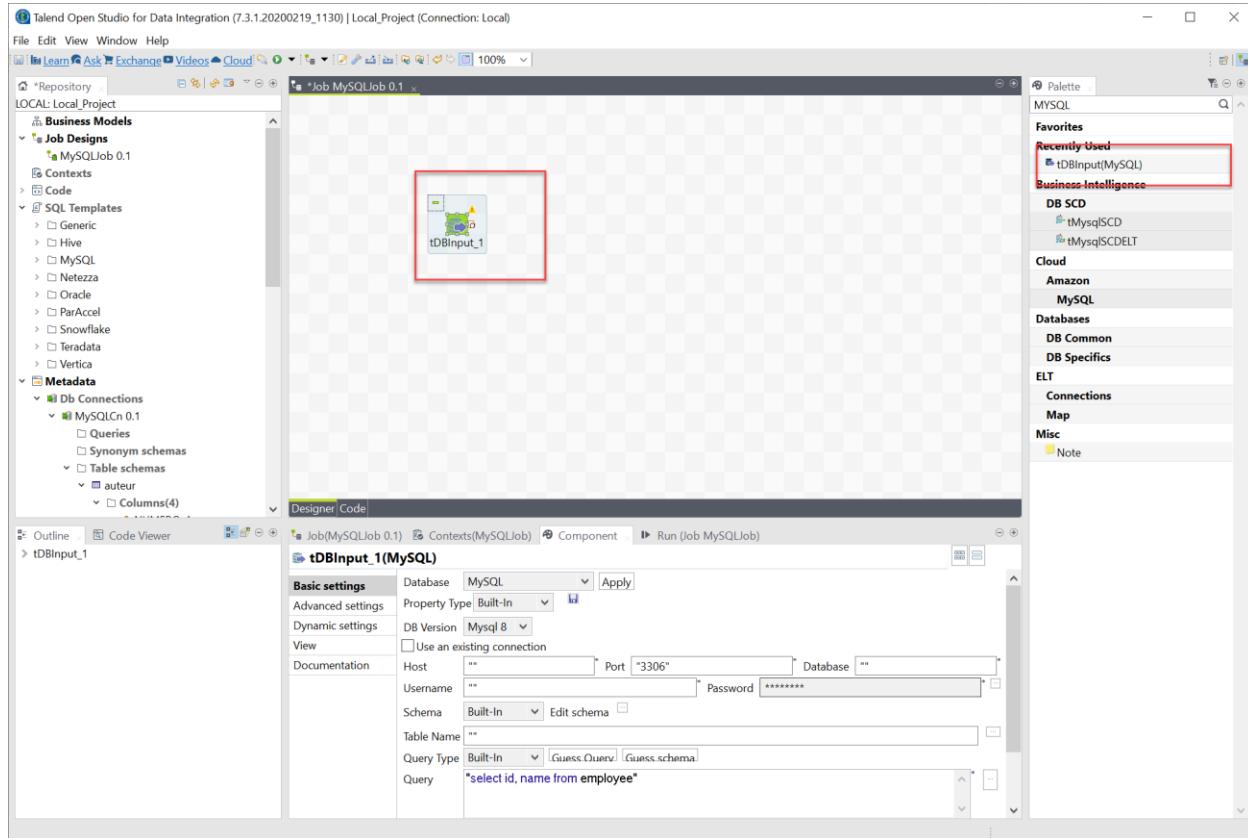


14. Vérifier les schémas de chaque table

**15. Création d'un premier job :****16. Création d'un premier job : ajout de la connexion**

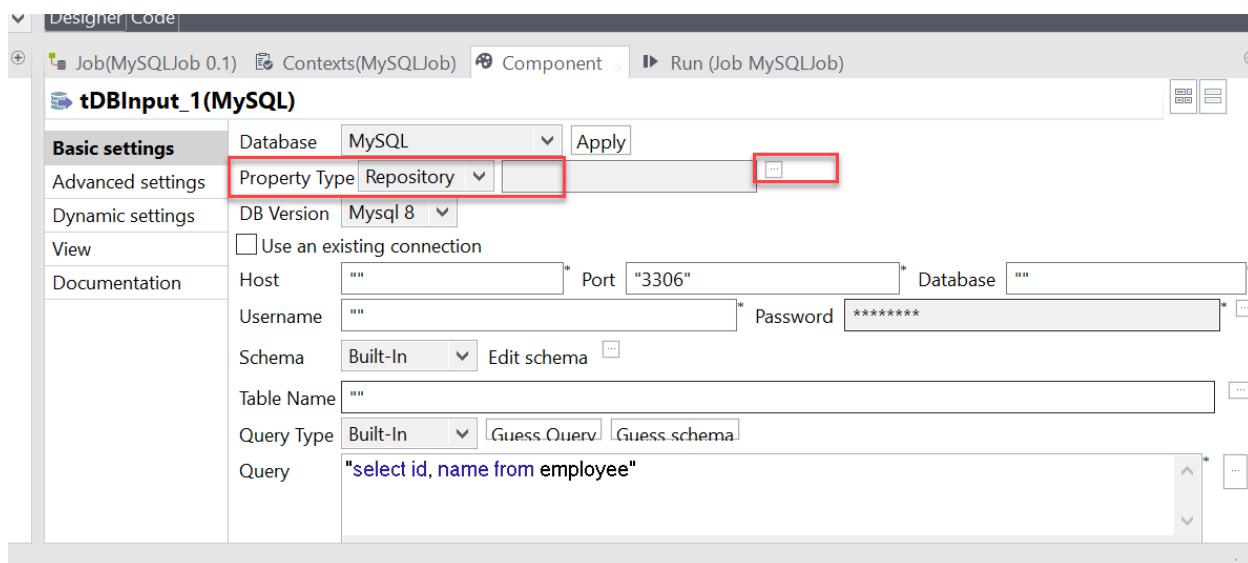
– Faire glisser le tDBInput (MySQL) dans la partie Design avec la table Auteur



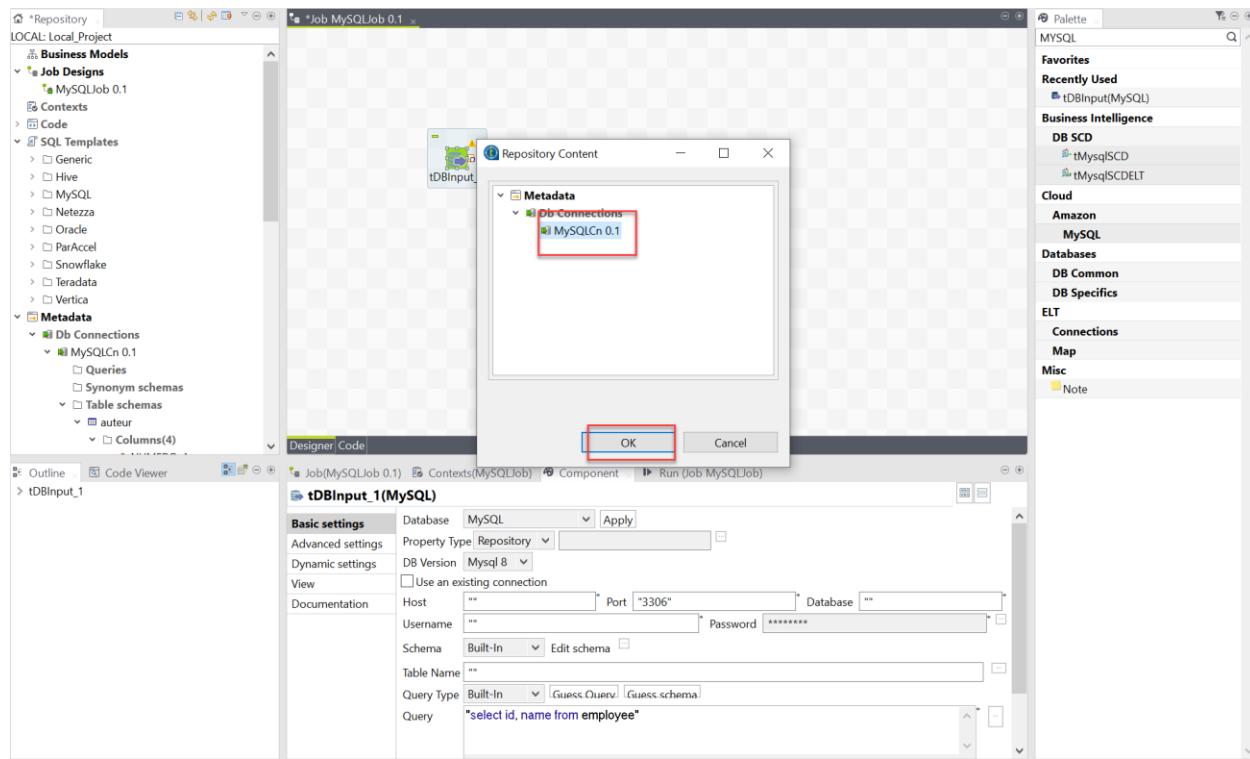


17. Configurer le composant tDBInput(MYSQL)

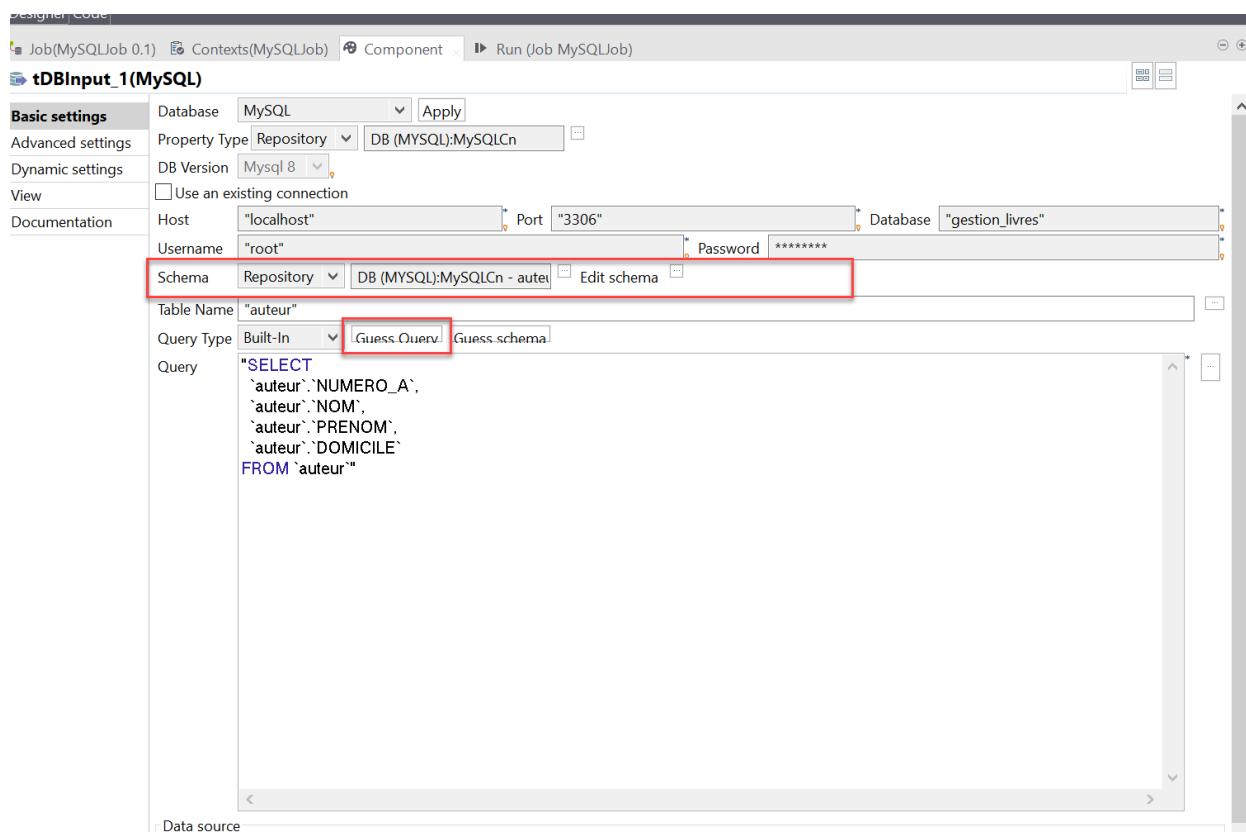
Sélectionner le type « Property Type » la valeur « Repository » et cliquer sur



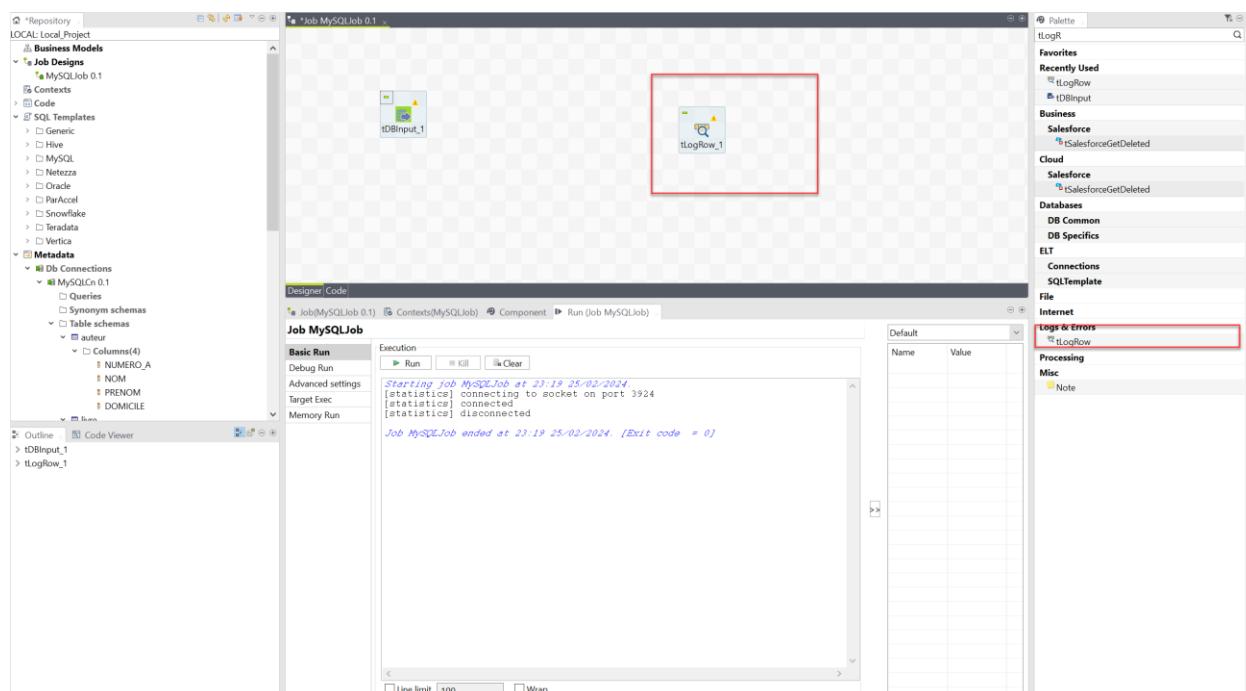
Sélectionner le type « MYSQLCN 0.1et cliquer sur OK



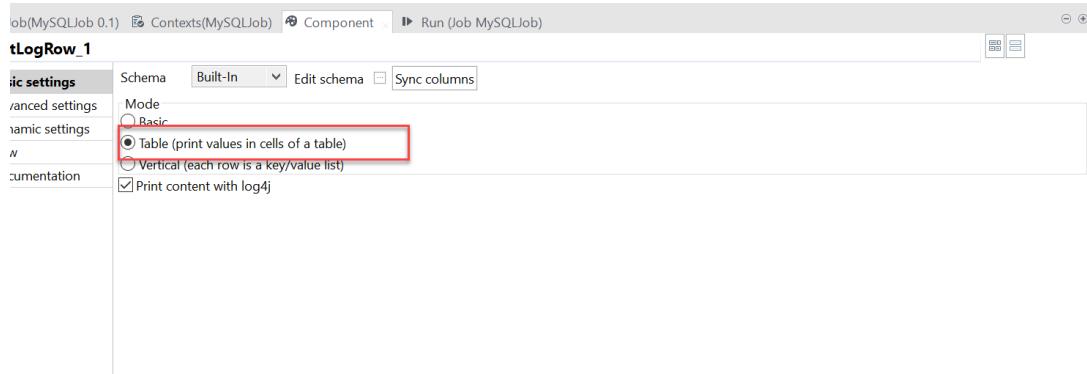
configurer le schéma et la requête à exécuter en cliquant sur Guess Query pour générer une requête de sélection avec les champs nécessaires



18. Création d'un premier job : ajout d'un composant de log

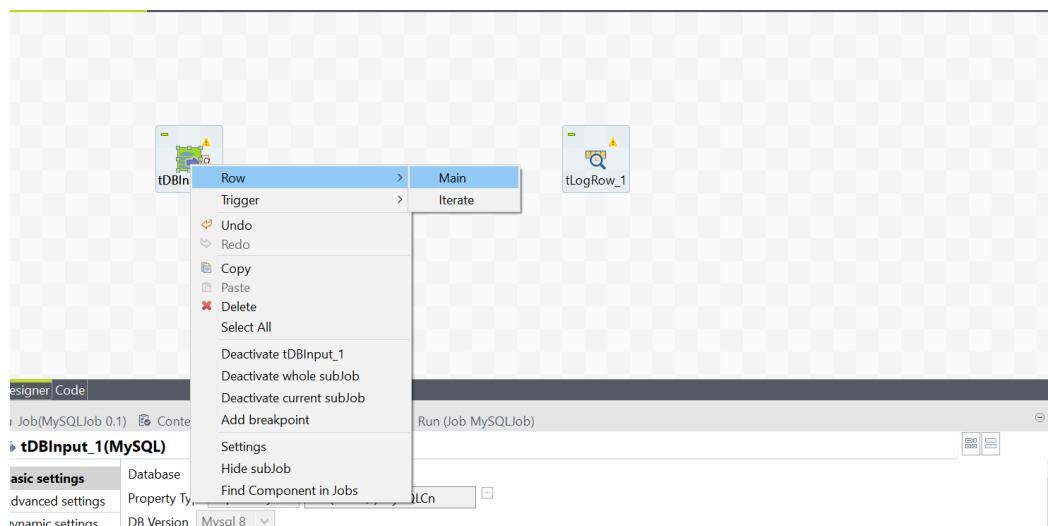


19. Configurer le composant de log



20. Liaison des composants

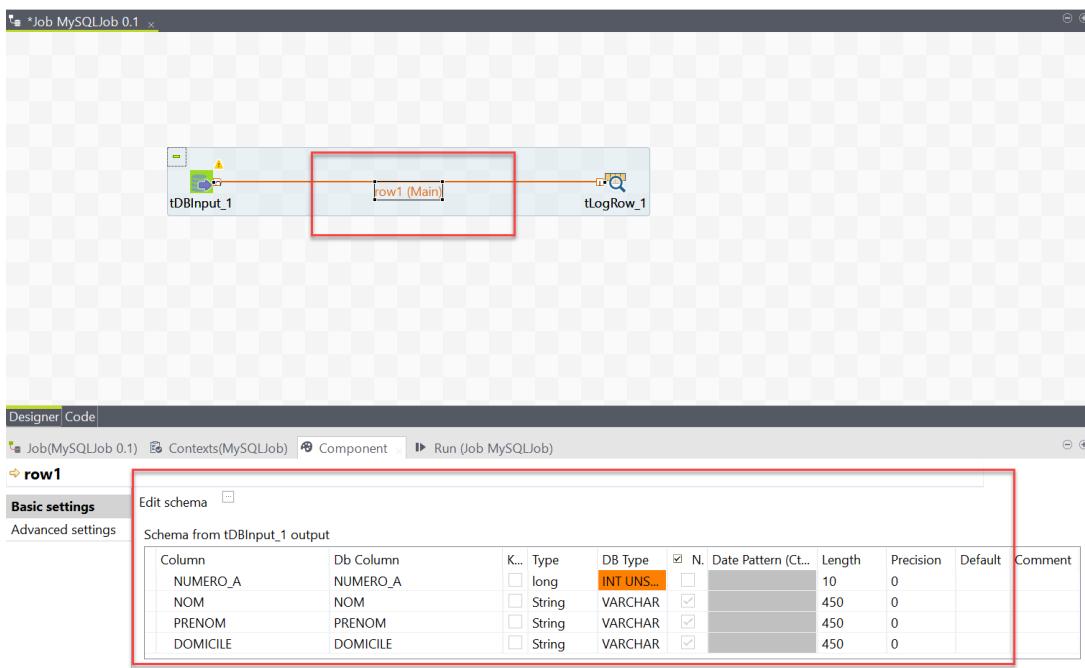
- Cliquer sur le composant tDBInput(MySQL) avec le bouton droit et afficher le menu contextuel



- Choisir Row -> Main et cliquer sur le composant tLogRow_1 pour établir la liaison



- La liaison est établie en fournissant l'output attendu

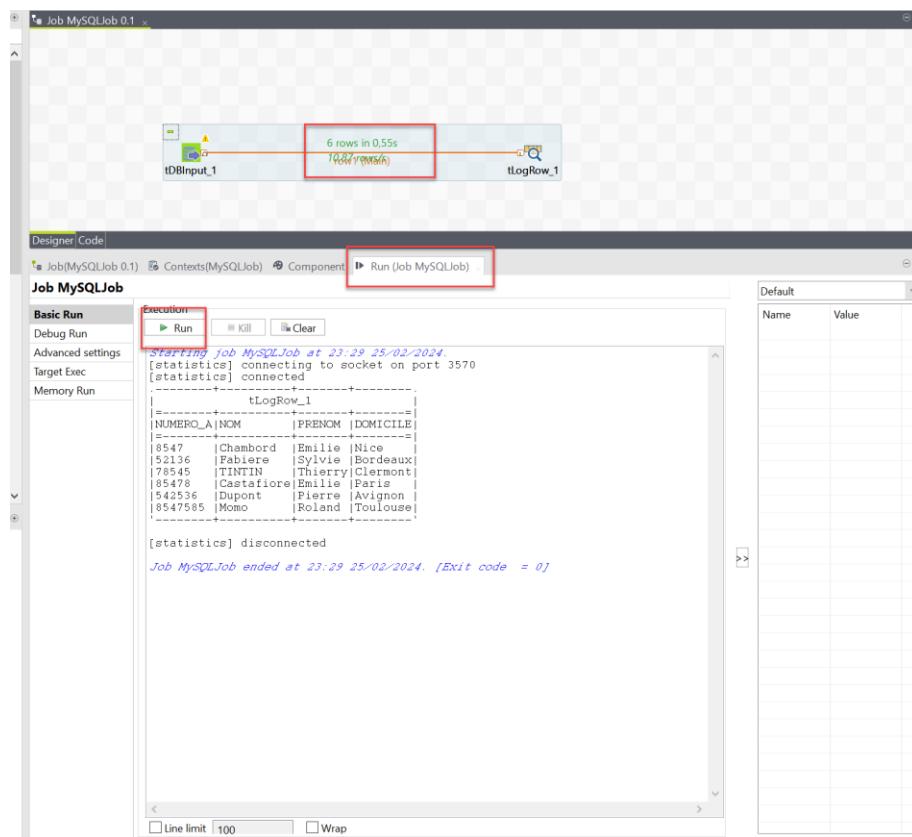


The screenshot shows the Talend Designer interface with the following details:

- Job Overview:** *Job MySQLJob 0.1
- Job Flow:** tDBInput_1 → row1 (Main) → tLogRow_1
- Schema Editor (row1 (Main)):**
 - Basic settings:** Advanced settings
 - Edit schema:** Schema from tDBInput_1 output
 - Table:**

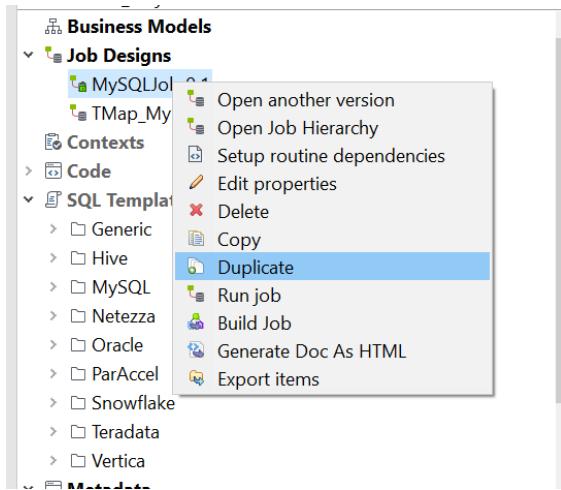
Column	Db Column	K...	Type	DB Type	N.	Date Pattern (Ct...)	Length	Precision	Default	Comment
NUMERO_A	NUMERO_A	<input type="checkbox"/>	long	INT UNSIGNED	<input checked="" type="checkbox"/>		10	0		
NOM	NOM	<input type="checkbox"/>	String	VARCHAR	<input checked="" type="checkbox"/>		450	0		
PRENOM	PRENOM	<input type="checkbox"/>	String	VARCHAR	<input checked="" type="checkbox"/>		450	0		
DOMICILE	DOMICILE	<input type="checkbox"/>	String	VARCHAR	<input checked="" type="checkbox"/>		450	0		

21. Exécuter le Job

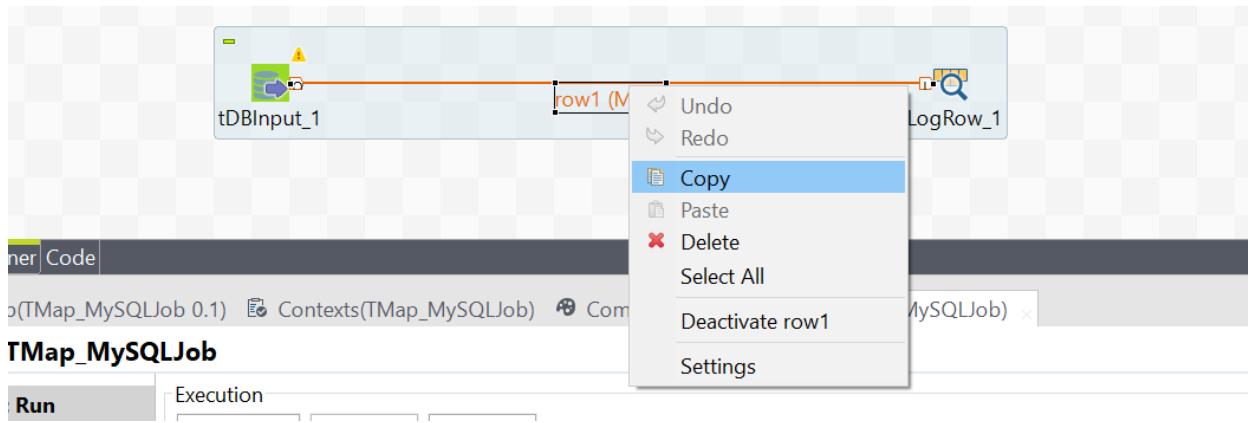


22. Modification de ce job : ajout d'un composant d'aiguillage tMap

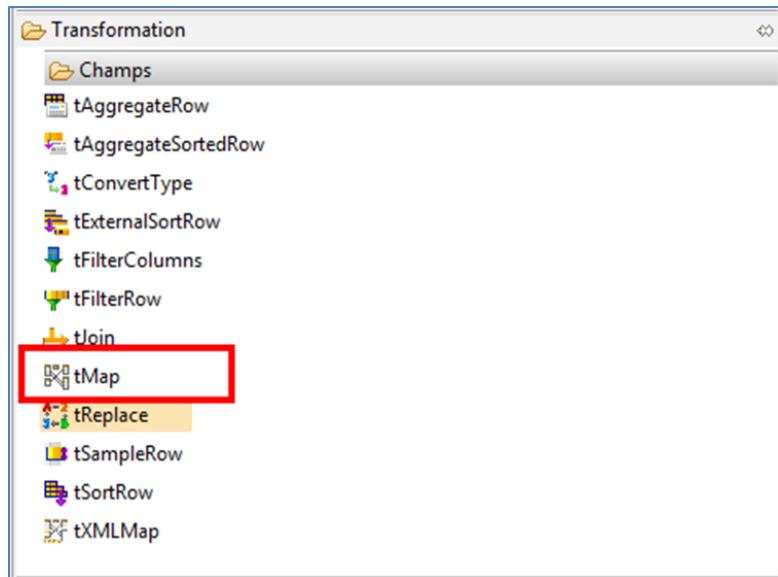
- Dupliquer le projet



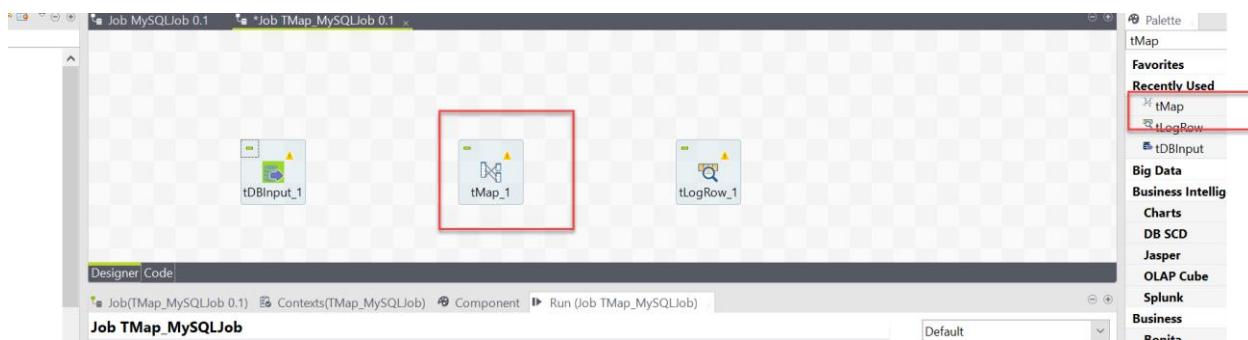
- Supprimer la liaison dans le nouveau job



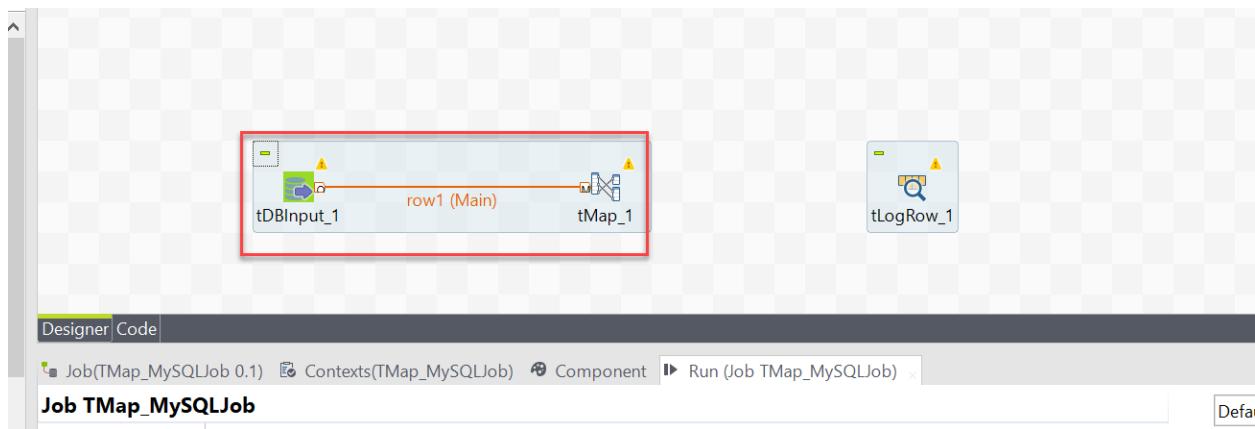
- Ajouter un composant tMap



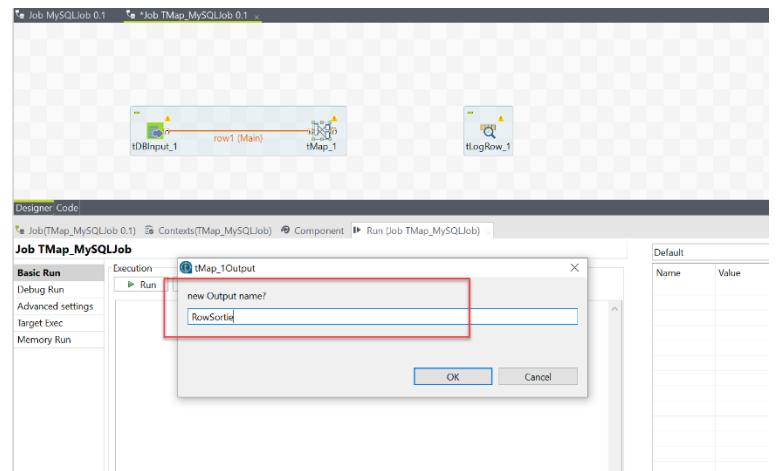
- Modification de ce job : liaison avec les autres composants

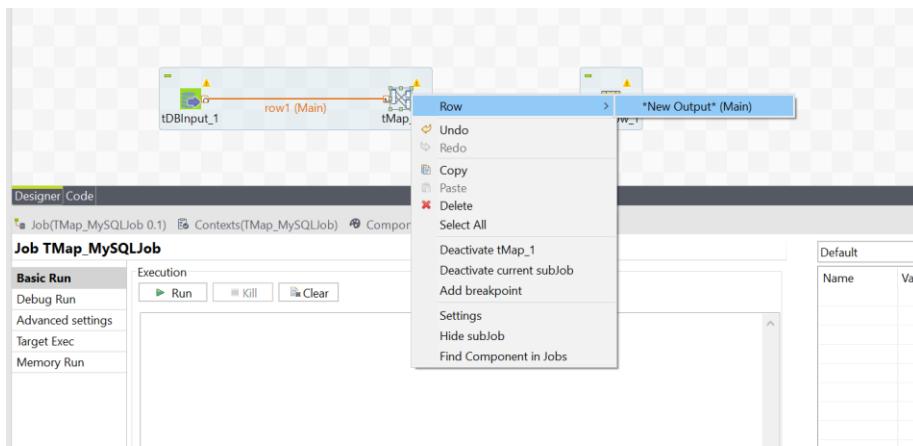


- Modification de ce job : Ajout liaison entre liaison Input(MySQL) entre tMap

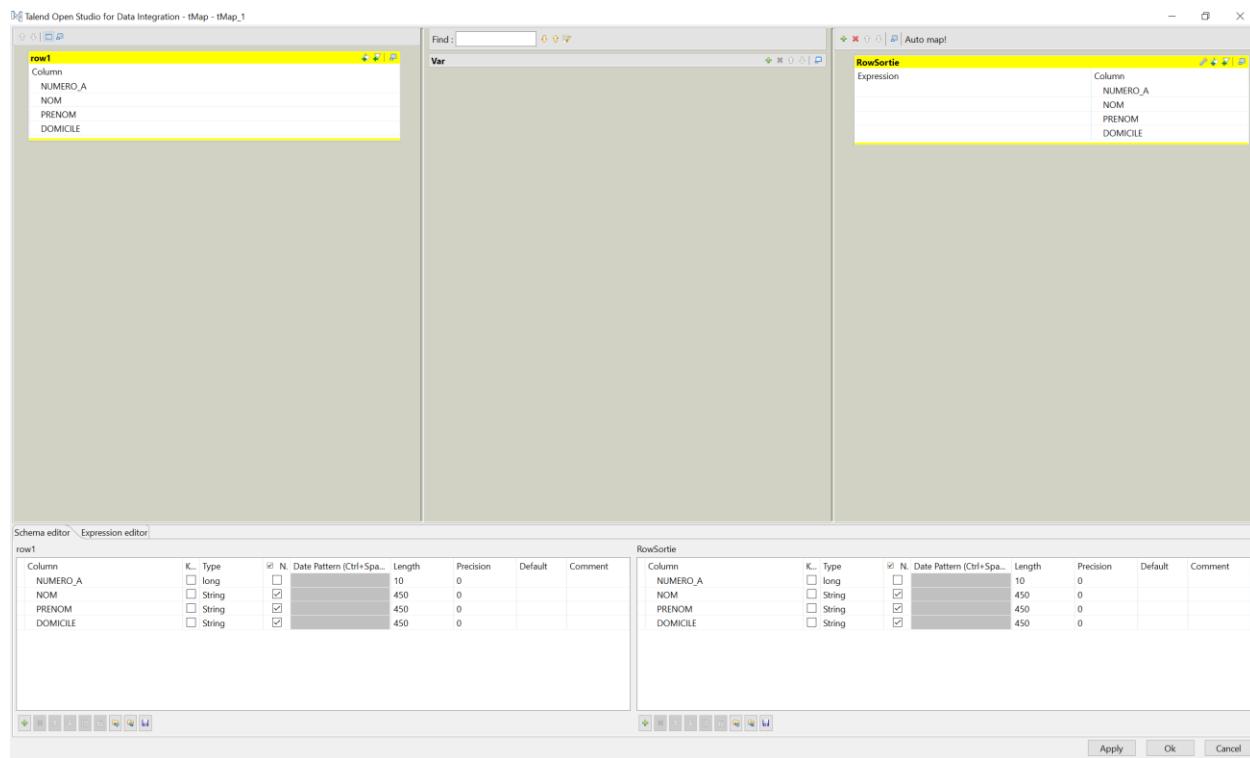


- Etablir liaison entre tMapr et LogRow en spécifiant le nom de l'output

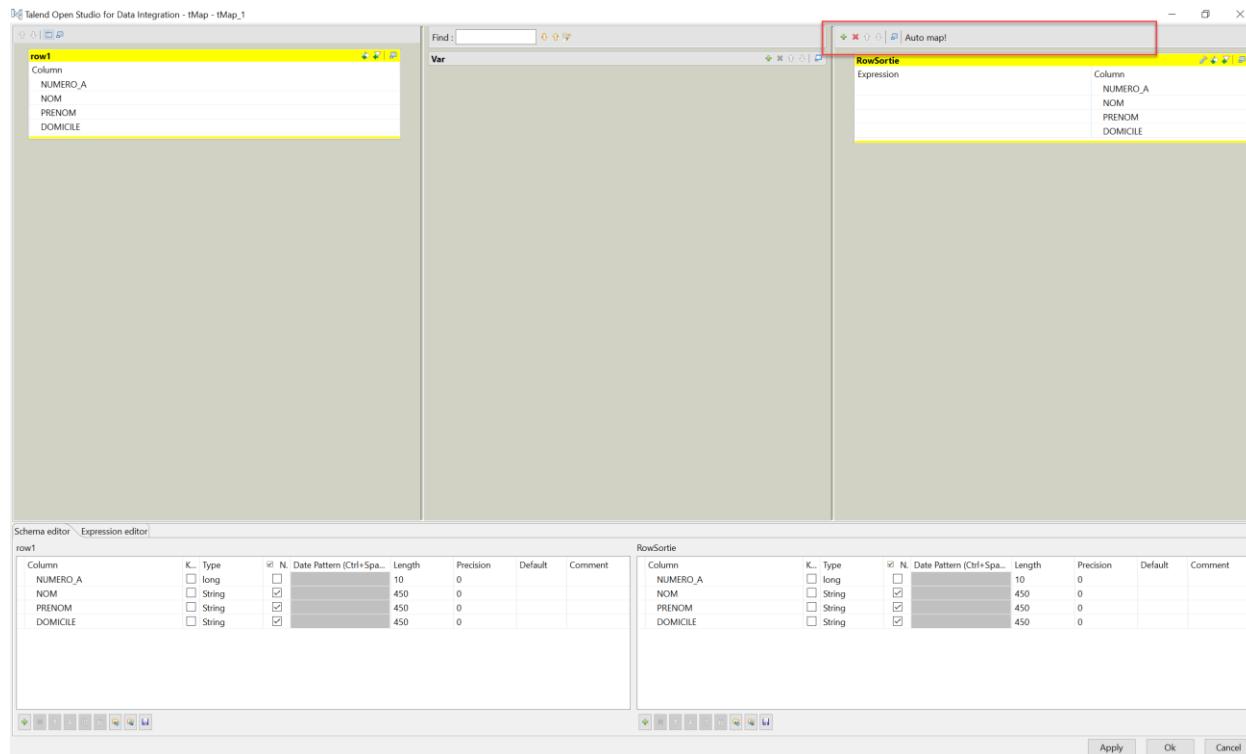




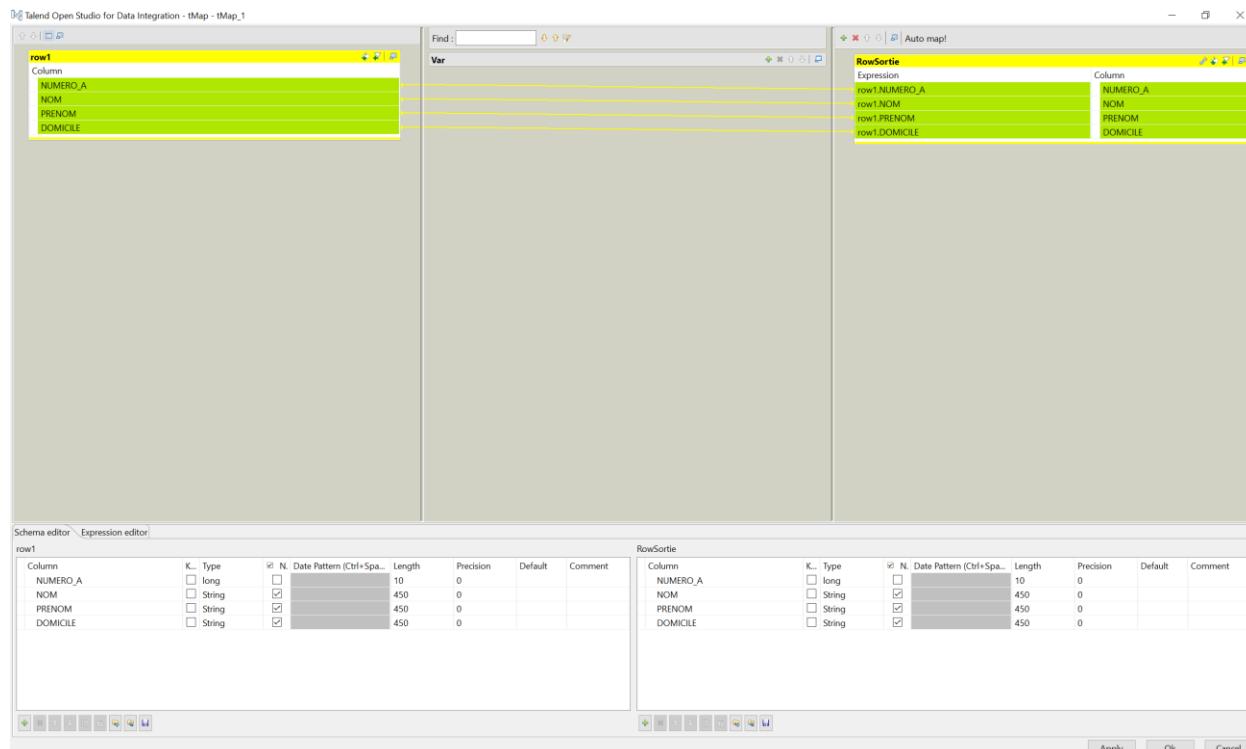
- Ouvrir le tMap



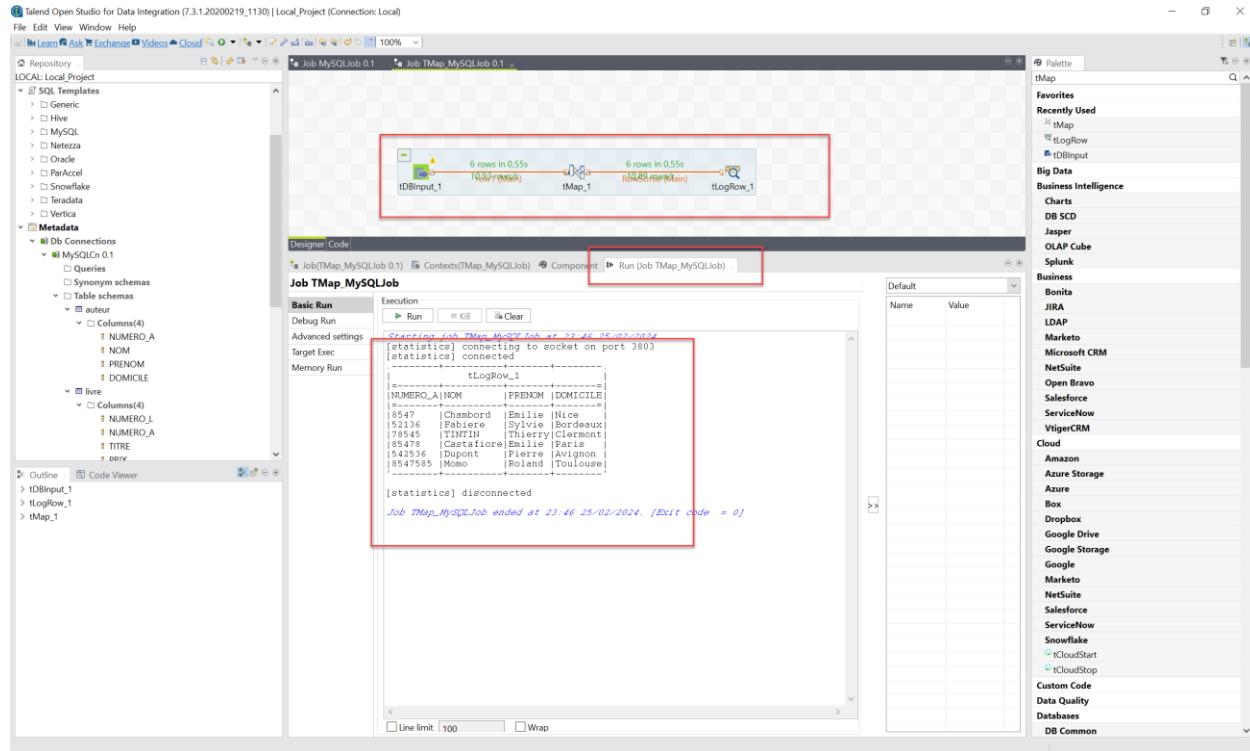
- Puis appuyer sur mapping auto en haut à droite



- Une correspondance sera établie automatiquement

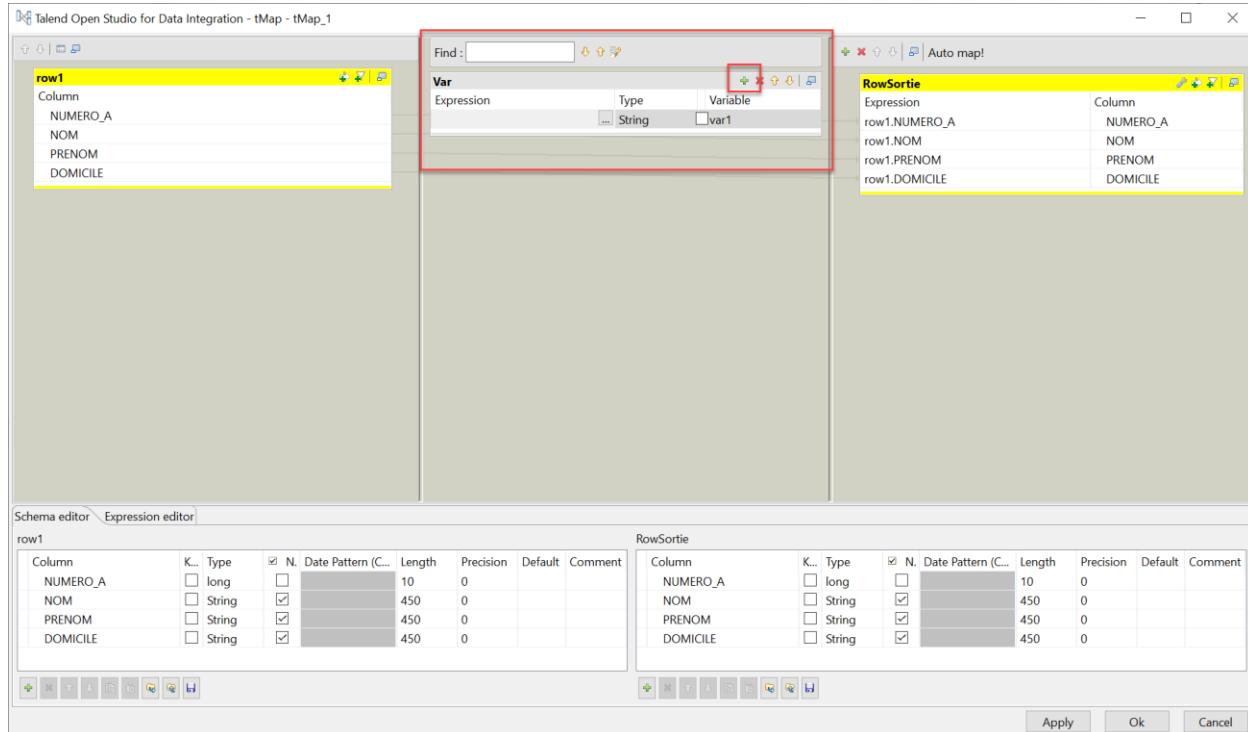


- Exécuter le job à nouveau (normalement il n'y a pas de changement)

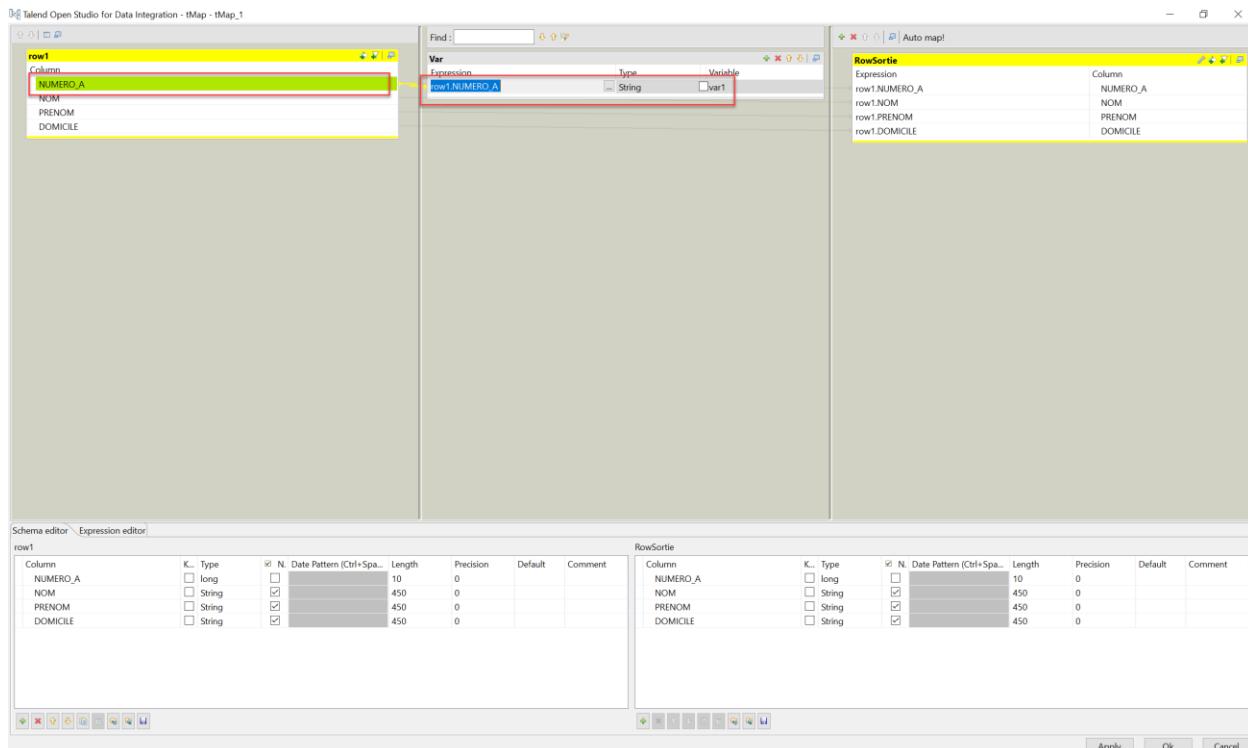


23. Modification de ce job :

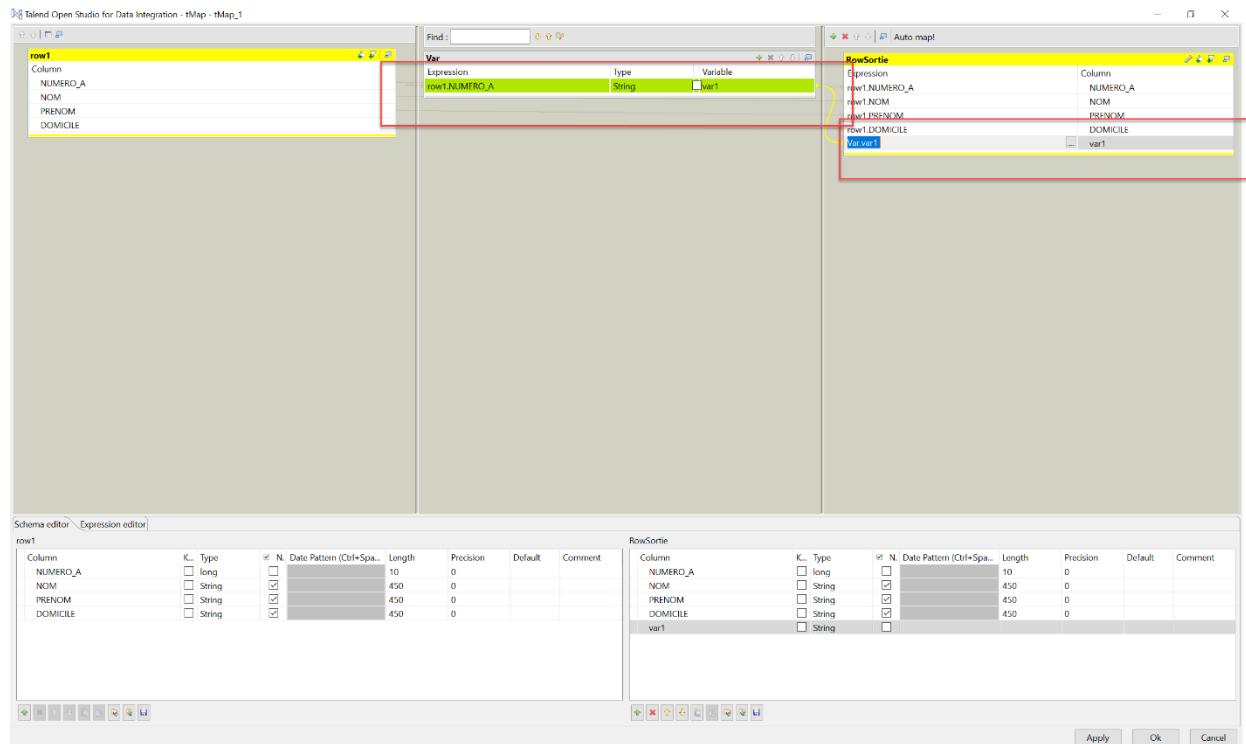
- Ouvrir à nouveau le tMap et créer le champ var1 dans la partie droite.



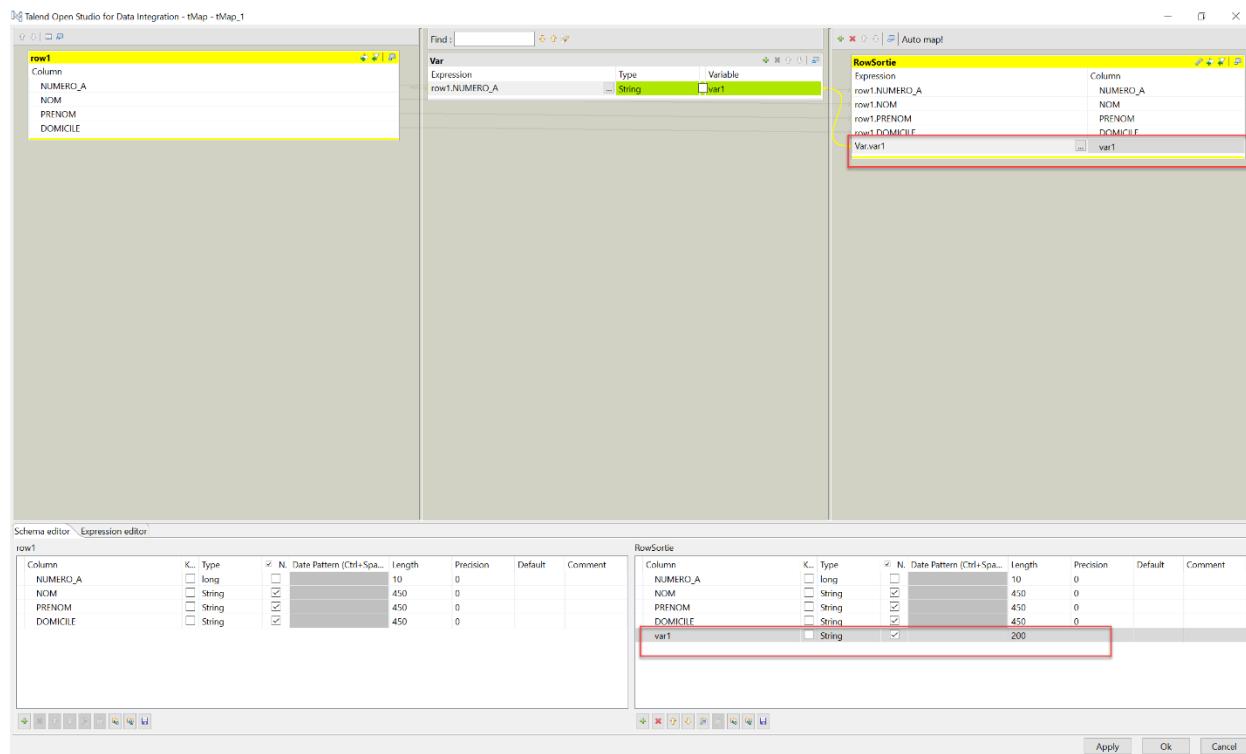
- Etablir une liaison entre la variable et Numéro en glissant le Numero_A vers la variable**



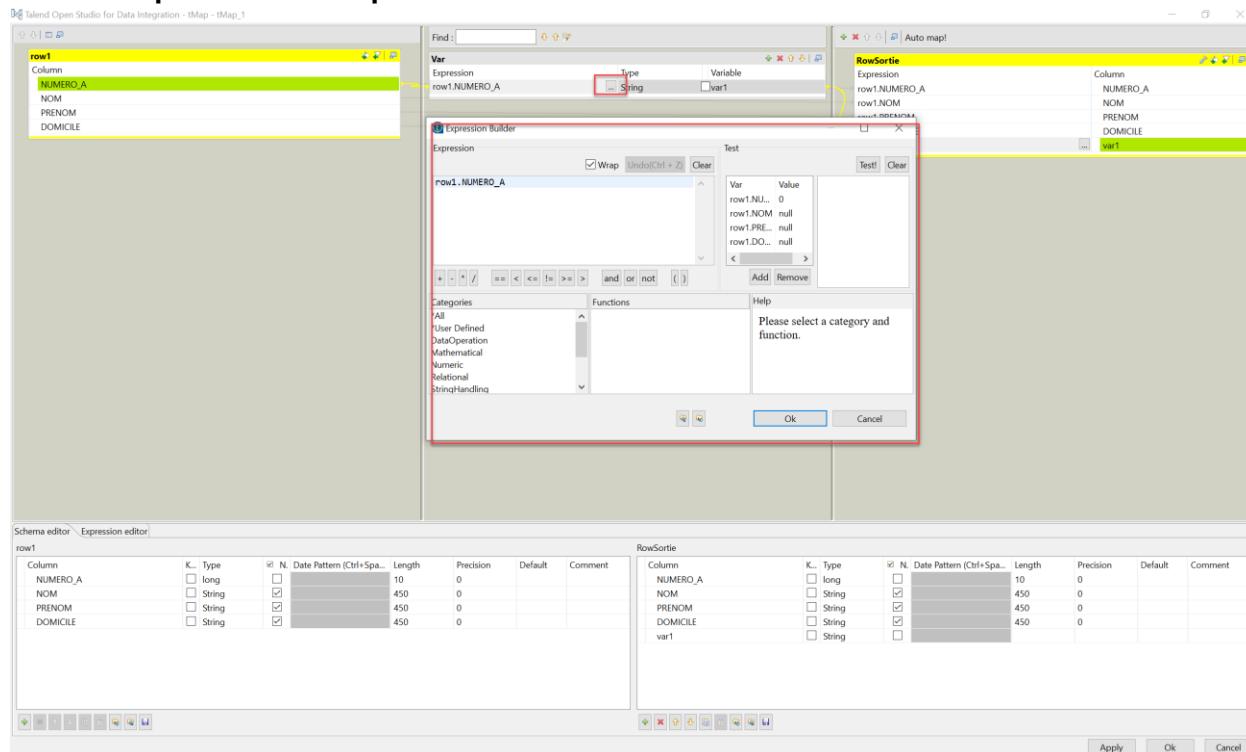
- Glisser cette variable vers la sortie**



- Cocher que celui-ci peut être null et qu'il peut avoir une taille de 200 (cf. partie en bas à droite).**

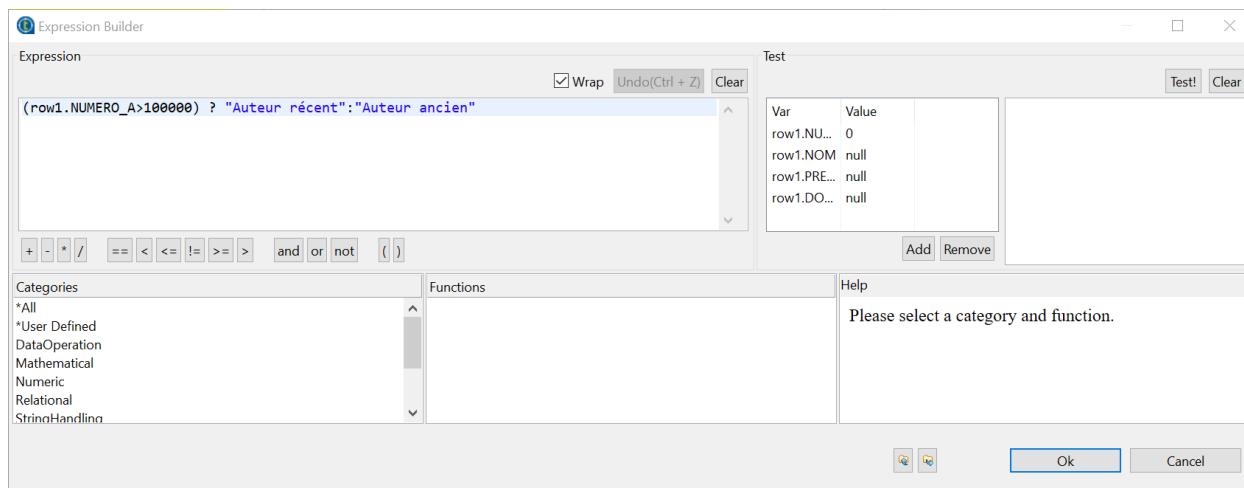


- Créer un nouveau champ dans la colonne du milieu puis cliquer sur les « ... » pour éditer l'expression :

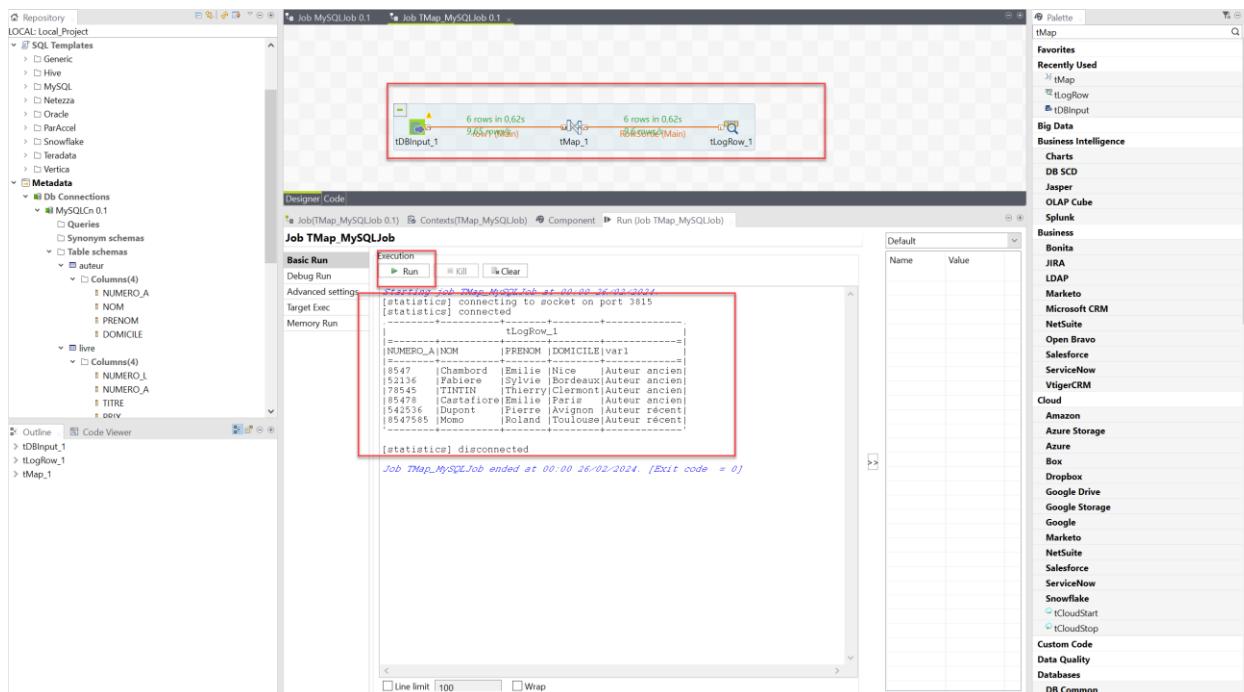


- Ajouter le test ternaire dans le nouveau champ :

- **(row1.NUMERO_A>100000)?"Auteur récent":"Auteur ancien"**

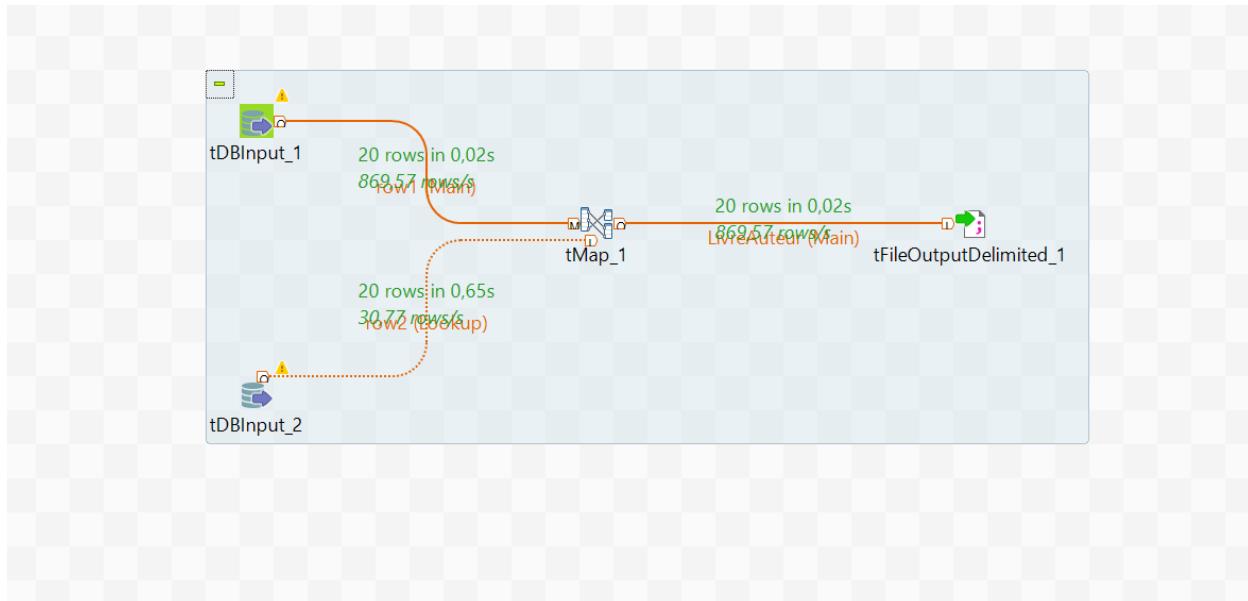


- Exécuter à nouveau le job, vous devriez obtenir ce type de résultat



Exercice à faire

L'objectif de cet exercice est de créer une ETL permettant de fusionner les tables "auteurs" et "livres" afin de stocker les données dans un fichier CSV nommé "BookAuteur.csv".



Votre job Talend devrait ressembler à quelque chose comme cela :

tMySQLInput ---(main)--> tMap ---(main)--> tFileOutputDelimited | ---(lookup)--> (Lookup sur la table auteur)

1. Creation de la base de données :

Pour créer une nouvelle base de données en utilisant le fichier SQL fourni, vous pouvez suivre les étapes suivantes :

- Modifier le fichier docker compose
- Exécution du script SQL : Copiez le contenu du fichier SQL dans un fichier nommé, par exemple, schema.sql.
- Utilisation de Docker pour MySQL : Utilisez Docker pour créer un conteneur MySQL dans lequel vous exécuterez le script SQL pour créer la nouvelle base de données et ses tables.

La nouvelle base de données à créer est décrite dans le fichier SQL fourni.

```
-- Create database
CREATE DATABASE IF NOT EXISTS `bookdb` DEFAULT CHARACTER SET utf8;
USE `bookdb`;

-- Table `auteur`
```



Talend for Data Integration

Trainer : Sellami Mokhtar
mokhtar.sellami@gmail.com



```
DROP TABLE IF EXISTS `auteur`;
CREATE TABLE `auteur` (
  `NUMERO_A` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `NOM` varchar(450) DEFAULT NULL,
  `PRENOM` varchar(450) DEFAULT NULL,
  `DOMICILE` varchar(450) DEFAULT NULL,
  PRIMARY KEY (`NUMERO_A`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;

-- Insert sample authors
INSERT INTO `auteur` (`NOM`, `PRENOM`, `DOMICILE`) VALUES
('Smith', 'John', 'New York'),
('Johnson', 'Emily', 'San Francisco'),
('Garcia', 'Miguel', 'Madrid'),
('Brown', 'Jessica', 'London'),
('Chen', 'Wei', 'Beijing'),
('Dubois', 'Marie', 'Paris'),
('Kim', 'Seong', 'Seoul'),
('Müller', 'Hans', 'Berlin'),
('Rossi', 'Marco', 'Rome'),
('Sato', 'Takashi', 'Tokyo');

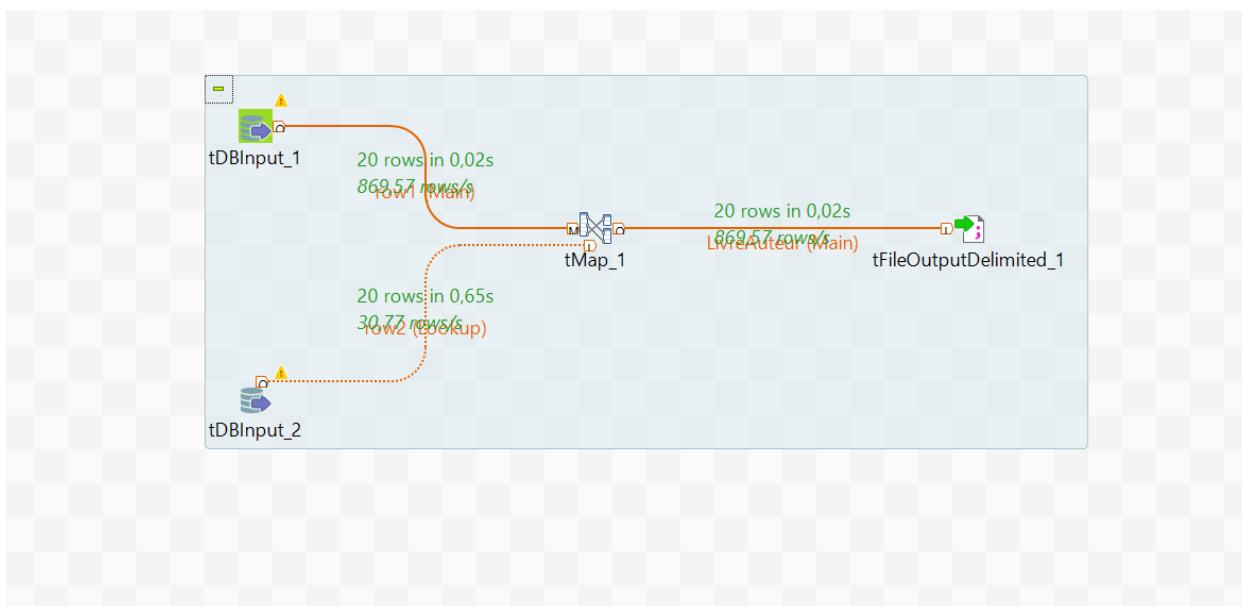
-- Table `livre`
DROP TABLE IF EXISTS `livre`;
CREATE TABLE `livre` (
  `NUMERO_L` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `NUMERO_A` int(10) unsigned NOT NULL,
  `TITRE` varchar(128) DEFAULT NULL,
  `PRIX` float DEFAULT NULL,
  PRIMARY KEY (`NUMERO_L`),
  CONSTRAINT `fk_auteur_numero` FOREIGN KEY (`NUMERO_A`) REFERENCES `auteur`(`NUMERO_A`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;

-- Insert sample books
INSERT INTO `livre` (`NUMERO_A`, `TITRE`, `PRIX`) VALUES
(1, 'Data Engineering for Dummies', 19.99),
(2, 'Practical Data Engineering', 29.99),
(3, 'Mastering Data Engineering', 34.99),
(4, 'Advanced Techniques in Data Engineering', 39.99),
(5, 'Big Data Infrastructure', 49.99),
(6, 'Data Pipelines and ETL Processing', 24.99),
(7, 'Data Science Essentials', 29.99),
```

```
(8, 'Machine Learning in Practice', 39.99),
(9, 'Cloud Data Solutions', 44.99),
(10, 'Data Governance and Compliance', 34.99),
(1, 'Data Warehousing Fundamentals', 29.99),
(2, 'Real-time Analytics', 49.99),
(3, 'Data Quality Management', 39.99),
(4, 'Data Mining Techniques', 34.99),
(5, 'Introduction to Hadoop', 29.99),
(6, 'Spark in Action', 49.99),
(7, 'Python for Data Analysis', 39.99),
(8, 'R Programming for Data Science', 34.99),
(9, 'SQL Mastery', 29.99),
(10, 'NoSQL Databases', 49.99);
```

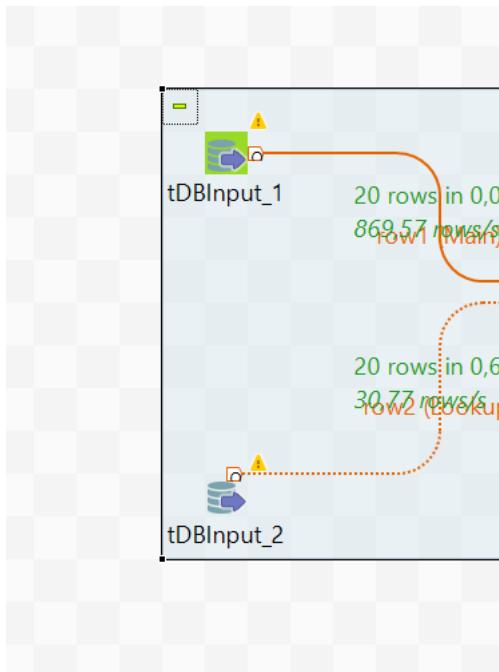
2. Creation de l'ETL :

Pour générer un fichier CSV avec les données fournies, vous pouvez utiliser Talend Data Integration. Voici comment vous pouvez le faire :



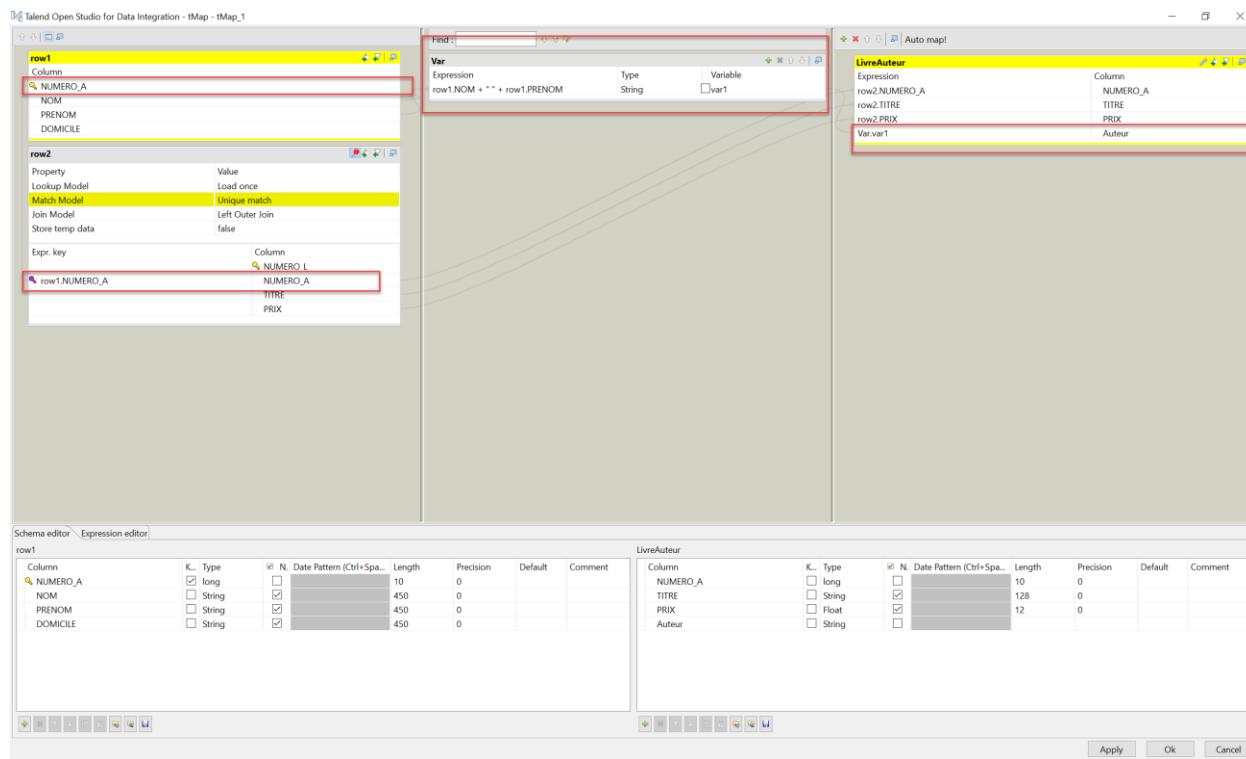
3. Extraction des données :

- Utilisez le composant **tMySQLInput** pour extraire les données des tables **livre** et **auteur** de la base de données **bookdb**.



4. Transformation des données :

- Dans le composant **tMap**, créez une colonne supplémentaire contenant le nom complet de l'auteur en concaténant les colonnes **NOM** et **PRENOM**.
- Réorganisez les colonnes selon le format requis pour le fichier CSV : **NUMERO_L, TITRE, PRIX, Auteur**.



Dans le composant **tMap**, vous pouvez utiliser une expression Java pour concaténer les colonnes **NOM** et **PRENOM**. Par exemple :

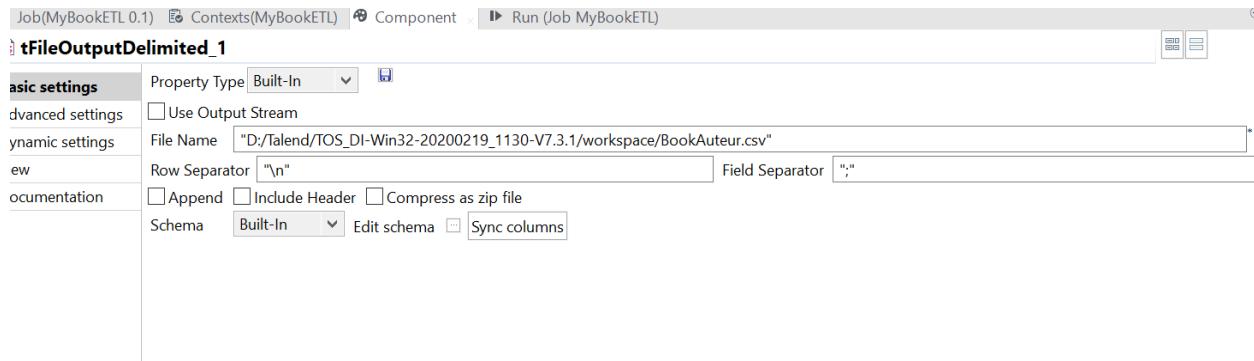
```
row1.NOM + " " + row1.PRENOM
```

Assurez-vous de mapper les colonnes appropriées de la table **livre** et de la table **auteur** dans le composant **tMap**.

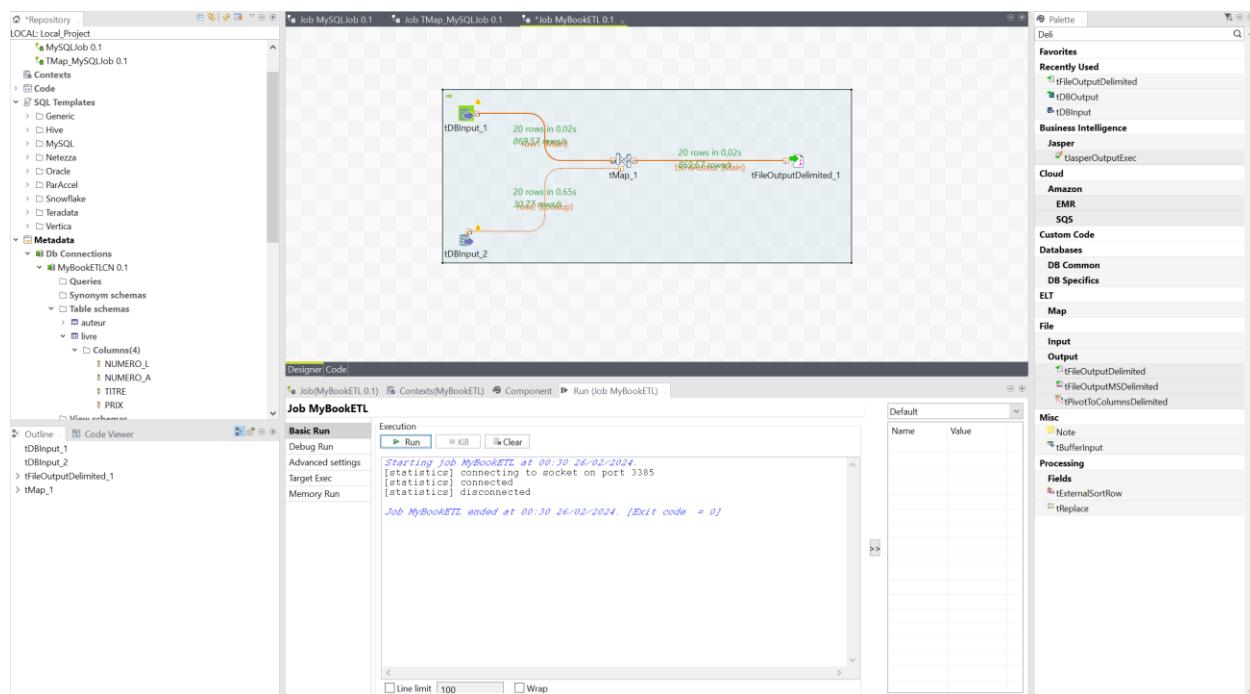
5. Chargement des données :

- Utilisez un composant **tFileOutputDelimited** pour écrire les données transformées dans un fichier CSV.
- Configurez le composant pour spécifier le chemin d'accès et le nom de fichier pour le fichier CSV de sortie.

Enfin, dans le composant **tFileOutputDelimited**, configurez le chemin d'accès et le nom de fichier pour votre fichier CSV de sortie, ainsi que les délimiteurs appropriés pour séparer les champs.



Exécuter le job et vérifier par la suite la bonne exécution ;



En suivant ces étapes, vous pourrez générer un fichier CSV avec les données fournies,



Talend for Data Integration

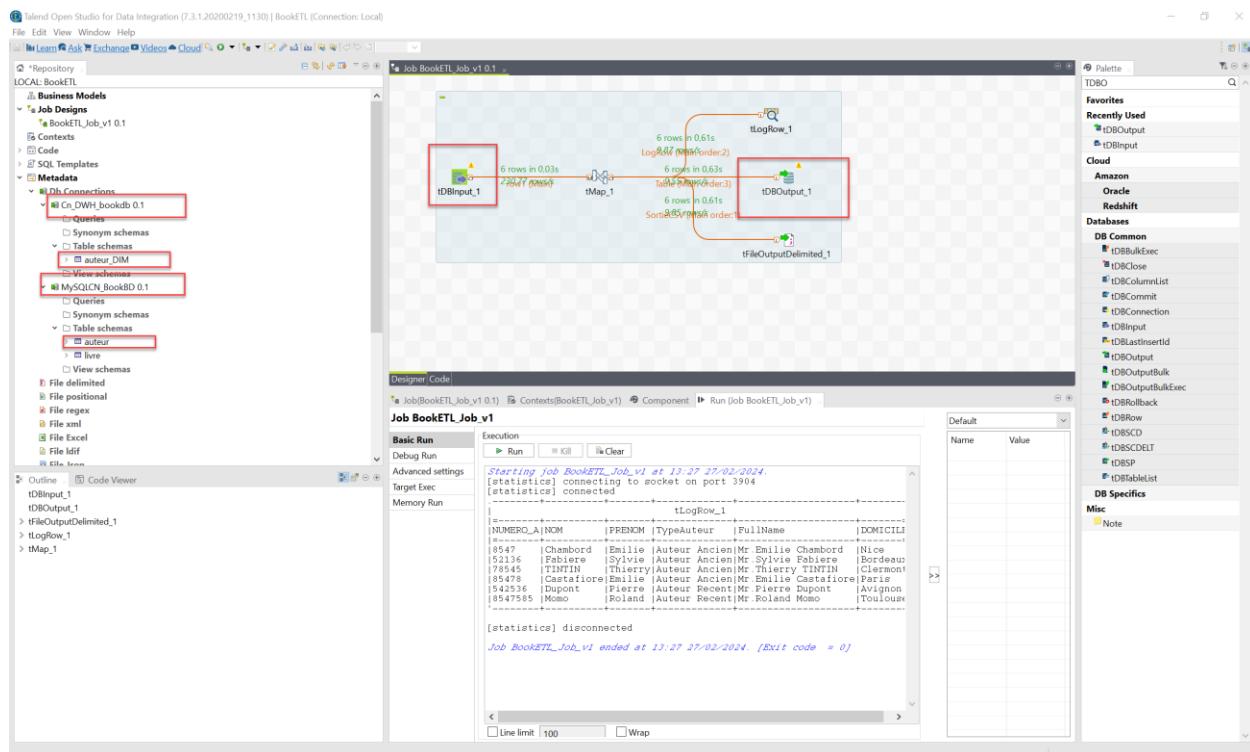
Trainer : Sellami Mokhtar
mokhtar.sellami@gmail.com



Exercice 2 :

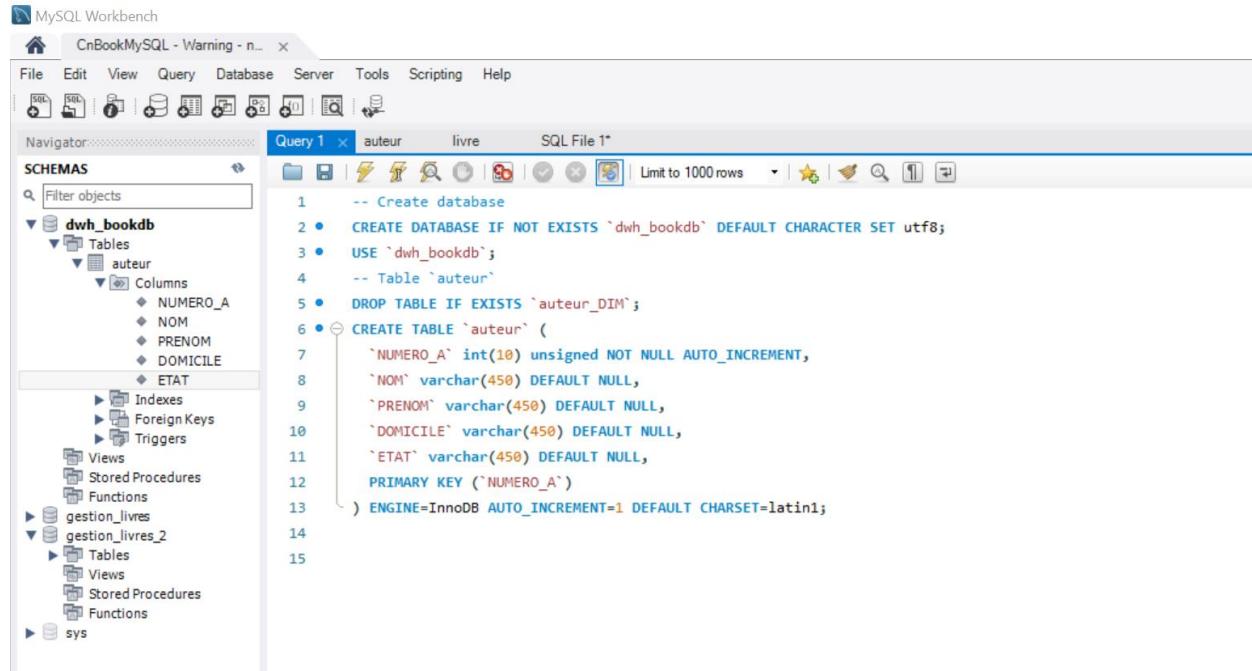
Chargement des données dans une bases de données MSYQL

Dans cet exemple on duplique le job et on ajoute un autre composant pour insérer des données dans la base de données cible



6. Préparation de la bases cibles :

Créer la base de données Target « dwh_books » avec le script suivant en utilisant workbench ou Docker Desktop



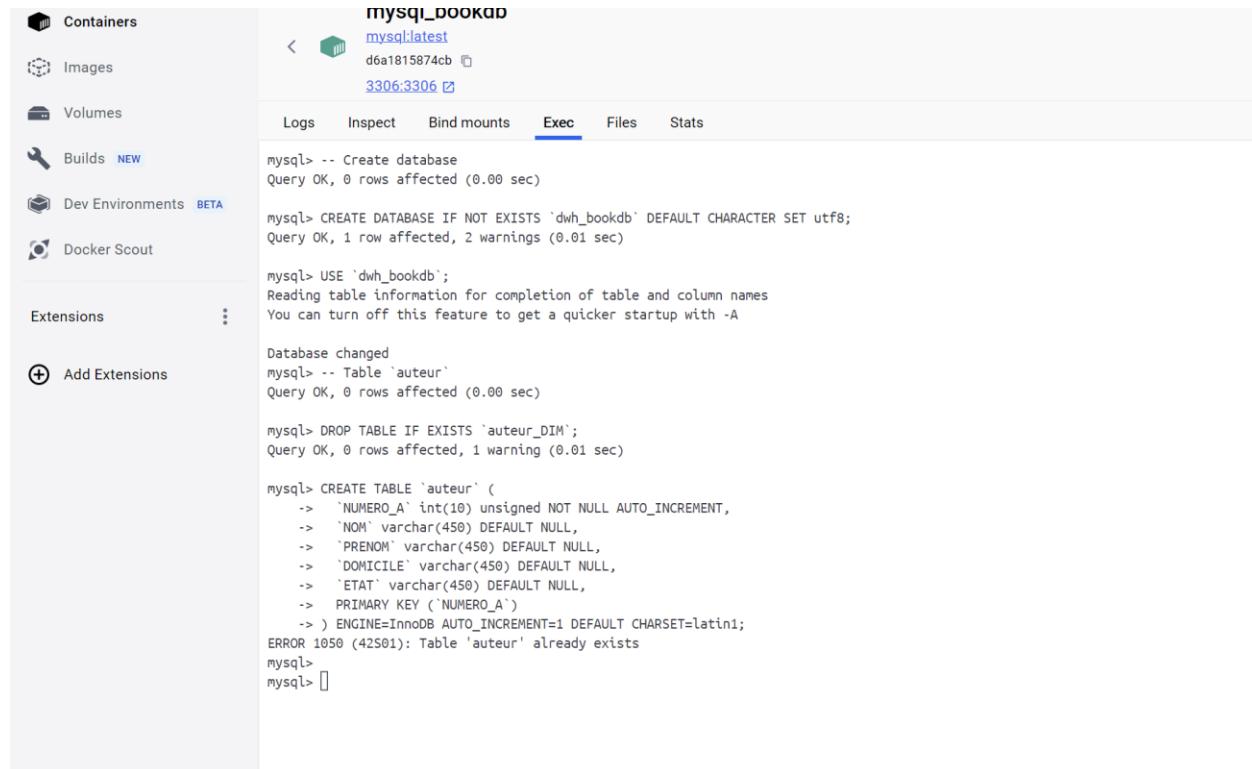
```

MySQL Workbench
CnBookMySQL - Warning - n...
File Edit View Query Database Server Tools Scripting Help
Navigator Schemas
Schemas
dwh_bookdb
  Tables
    auteur
      Columns
        NUMERO_A
        NOM
        PRENOM
        DOMICILE
        ETAT
      Indexes
      Foreign Keys
      Triggers
    Views
    Stored Procedures
    Functions
  gestion_livres
  gestion_livres_2
  Tables
    Views
    Stored Procedures
    Functions
sys

Query 1 × auteur livre SQL File 1*
1 -- Create database
2 • CREATE DATABASE IF NOT EXISTS `dwh_bookdb` DEFAULT CHARACTER SET utf8;
3 • USE `dwh_bookdb`;
4 -- Table `auteur`
5 • DROP TABLE IF EXISTS `auteur_DIM`;
6 • CREATE TABLE `auteur` (
7   `NUMERO_A` int(10) unsigned NOT NULL AUTO_INCREMENT,
8   `NOM` varchar(450) DEFAULT NULL,
9   `PRENOM` varchar(450) DEFAULT NULL,
10  `DOMICILE` varchar(450) DEFAULT NULL,
11  `ETAT` varchar(450) DEFAULT NULL,
12  PRIMARY KEY (`NUMERO_A`)
13 ) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;
14
15

```

Avec Docker



```

Containers
  mysql_db
    mysql:latest
      d6a1815874cb
      3306:3306
Logs Inspect Bind mounts Exec Files Stats

mysql> -- Create database
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE DATABASE IF NOT EXISTS `dwh_bookdb` DEFAULT CHARACTER SET utf8;
Query OK, 1 row affected, 2 warnings (0.01 sec)

mysql> USE `dwh_bookdb`;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> -- Table `auteur`
Query OK, 0 rows affected (0.00 sec)

mysql> DROP TABLE IF EXISTS `auteur_DIM`;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> CREATE TABLE `auteur` (
->   `NUMERO_A` int(10) unsigned NOT NULL AUTO_INCREMENT,
->   `NOM` varchar(450) DEFAULT NULL,
->   `PRENOM` varchar(450) DEFAULT NULL,
->   `DOMICILE` varchar(450) DEFAULT NULL,
->   `ETAT` varchar(450) DEFAULT NULL,
->   PRIMARY KEY (`NUMERO_A`)
-> ) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;
ERROR 1050 (42S01): Table 'auteur' already exists
mysql>
mysql> []

```

-- Create database

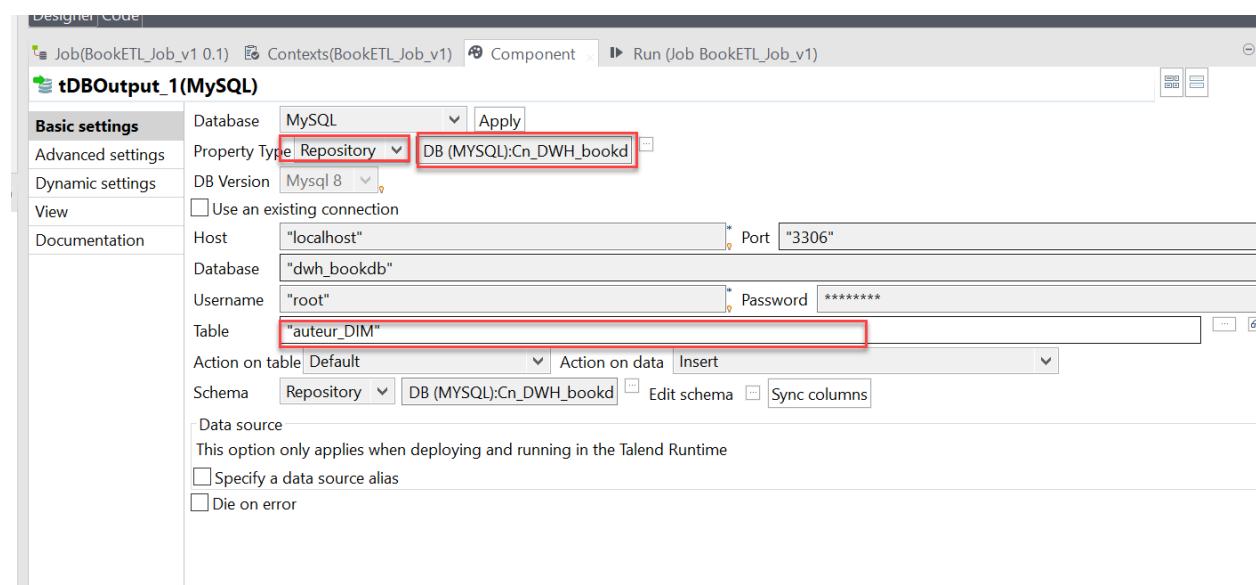
```

CREATE DATABASE IF NOT EXISTS `dwh_bookdb` DEFAULT CHARACTER SET utf8;
USE `dwh_bookdb`;
-- Table `auteur`
DROP TABLE IF EXISTS `auteur_DIM`;
CREATE TABLE `auteur` (
    `NUMERO_A` int(10) unsigned NOT NULL AUTO_INCREMENT,
    `NOM` varchar(450) DEFAULT NULL,
    `PRENOM` varchar(450) DEFAULT NULL,
    `DOMICILE` varchar(450) DEFAULT NULL,
    `ETAT` varchar(450) DEFAULT NULL,
    PRIMARY KEY (`NUMERO_A`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;
    
```

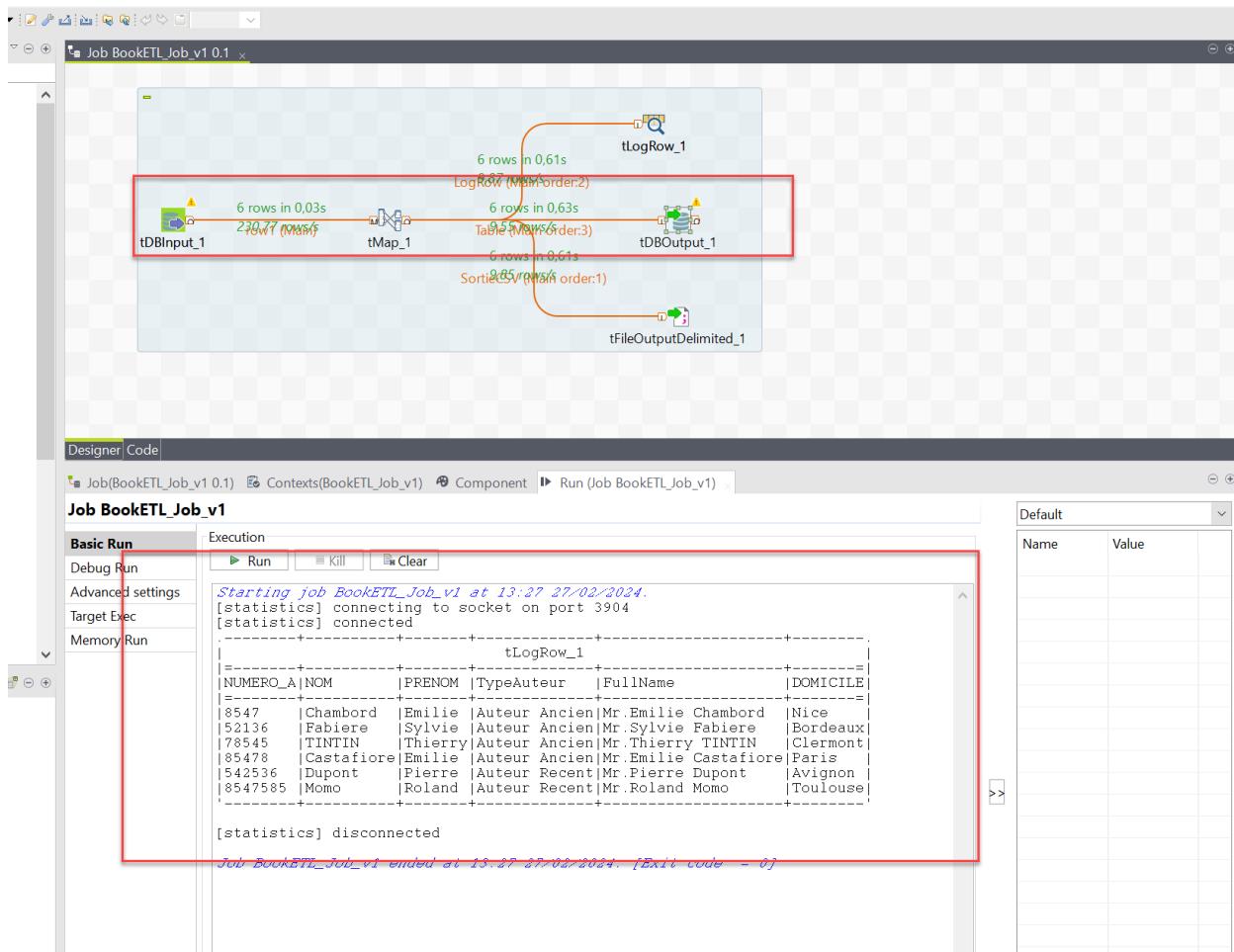
1. Chargement des données dans une table “Auteur_DIM”:

- Utilisez un composant **tDBOutput** pour écrire les données transformées dans une table mysql « auteur_dim » .

Enfin, dans le composant **tDBOutput**, configurez les paramètres de l'accès à la bases de données et de sortie, ainsi que le nom de la table « Auteur_DIM » appropriée.



Exécuter le job et vérifier par la suite la bonne exécution ;

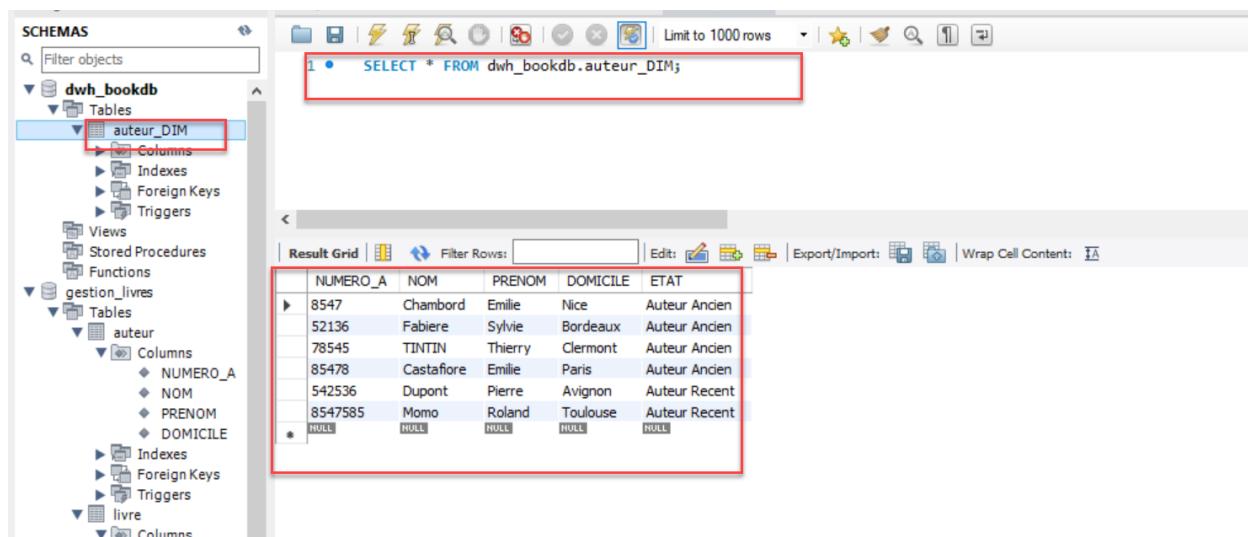


The screenshot shows the Talend Workbench interface. At the top, the job name "Job BookETL_Job_v1.0.1" is visible. The main area displays the job flow with components: tDBInput_1, tMap_1, tLogRow_1, tDBOutput_1, and tFileOutputDelimited_1. A red box highlights the central processing loop involving tMap_1, tLogRow_1, and tDBOutput_1. Below the flow, the "Designer" tab is active, showing the code for the job. The "Execution" tab is also visible, displaying the log output. The log shows the job starting at 13:27 27/02/2024, connecting to a socket on port 3904, and listing 6 rows processed in 0.61s. The output table is shown as:

NUMERO_A	NOM	PRENOM	TypeAuteur	FullName	DOMICILE
8547	Chambord	Emilie	Auteur Ancien	Mr. Emilie Chambord	Nice
52136	Fabiere	Sylvie	Auteur Ancien	Mr. Sylvie Fabiere	Bordeaux
78545	TINTIN	Thierry	Auteur Ancien	Mr. Thierry TINTIN	Clermont
85478	Castafiore	Emilie	Auteur Ancien	Mr. Emilie Castafiore	Paris
542536	Dupont	Pierre	Auteur Recent	Mr. Pierre Dupont	Avignon
8547585	Momo	Roland	Auteur Recent	Mr. Roland Momo	Toulouse

The execution log ends with the message "[statistics] disconnected".

En suivant ces étapes, vous pourrez charger dans la table « Auteur_DIM » avec les données fournies, avec Workbench ou Docker Desktop « Exec »



The screenshot shows the MySQL Workbench interface. On the left, the schema browser displays the "dwh_bookdb" database with tables like "auteur_DIM", "gestion_livres", and "livre". The "auteur_DIM" table is selected and highlighted with a red box. In the center, the SQL editor window contains the query: "SELECT * FROM dwh_bookdb.auteur_DIM;". The results grid below shows the data from the "auteur_DIM" table, which matches the data shown in the Talend execution log:

NUMERO_A	NOM	PRENOM	DOMICILE	ETAT
8547	Chambord	Emilie	Nice	Auteur Ancien
52136	Fabiere	Sylvie	Bordeaux	Auteur Ancien
78545	TINTIN	Thierry	Clermont	Auteur Ancien
85478	Castafiore	Emilie	Paris	Auteur Ancien
542536	Dupont	Pierre	Avignon	Auteur Recent
8547585	Momo	Roland	Toulouse	Auteur Recent
NULL	NULL	NULL	NULL	NULL



Talend for Data Integration

Trainer : Sellami Mokhtar
mokhtar.sellami@gmail.com



```
mysql> SELECT * FROM dwh_bookdb.auteur_DIM;
+-----+-----+-----+-----+
| NUMERO_A | NOM      | PRENOM   | DOMICILE | ETAT
+-----+-----+-----+-----+
| 8547    | Chambord | Emilie   | Nice     | Auteur Ancien |
| 52136   | Fabiere   | Sylvie   | Bordeaux  | Auteur Ancien |
| 78545   | TINTIN    | Thierry  | Clermont  | Auteur Ancien |
| 85478   | Castafiore | Emilie   | Paris    | Auteur Ancien |
| 542536  | Dupont    | Pierre   | Avignon  | Auteur Recent |
| 8547585 | Momo      | RoLand   | Toulouse | Auteur Recent |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```