

Data Warehouse and ETL Implementation

ID/X Partners Data Scientist VIX

Presented by
Fawwaz Nurmansyah

Fawwaz Nurmansyah

About Me:

A graduate Bachelor of Biology from Airlangga University who is interest in pursuing a career related to data after completing some bootcamp. Able to work well independently and as part of a team, even under tight deadlines and pressure.

Have less than a year experience in data science with understanding of data processing, data analysis, and machine learning. Proficient in extracting insights from complex datasets and transforming them into profitable recommendations for business.

Experience

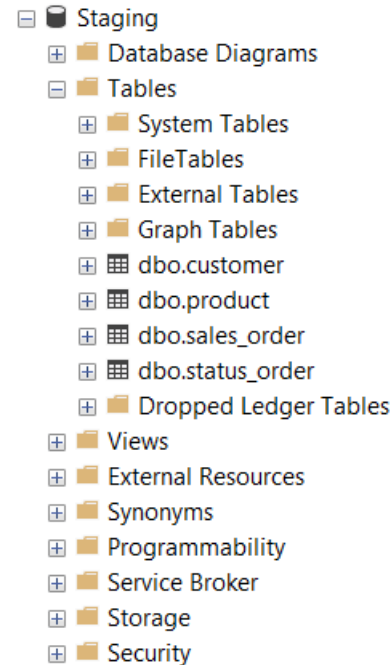
- Final Project Data Science Bootcamp Rakamin Academy
- Doing Exploratory Data Analysis on an E-commerce Shipping Data from [Kaggle](#), preprocessing data, and fitting a Machine Learning model to the dataset using Decision Tree, K-Nearest Neighbour (KNN), Adaboost, Extreme Gradient Boost (XGBoost), Random Forest Classifier, and CATBoost. The best fit models to predict the dataset using XGBoost with highest recall score of 98% and 97% recall cross-validation training and testing data. [Portofolio Link](#)

Case Study

Dalam melakukan restore database menggunakan file Staging.bak, hal pertama yang dilakukan adalah

1. Klik kanan pada folder Database dan memilih Restore Database
2. Kemudian, memilih sourcenya sebagai Device dan klik titik tiga.
3. Klik Add untuk mencari lokasi file Staging.bak
4. Untuk selanjutnya, klik OK sampai terdapat pesan bahwa database tersebut selesai dilakukan restore.

Pada gambar di samping merupakan hasil restore database tersebut yang memiliki 4 tabel, yaitu 'customer', 'product', 'sales_order', dan 'status_order'.

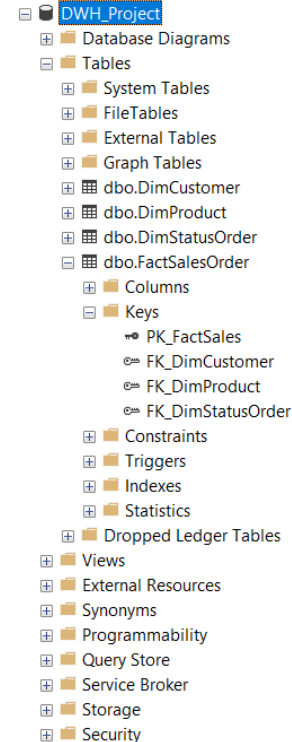


Case Study

Untuk membuat data warehouse, hal pertama yang dilakukan adalah membuat database yang akan dijadikan data warehouse yang akan diolah dengan menggunakan Talend sebagai ETL Studio.

Berikut adalah database dengan 4 tabel yang akan dijadikan data warehouse, yaitu 'DimCustomer', 'DimProduct', 'DimStatusOrder', dan 'FactSalesOrder'. Pada tabel dengan awalan 'Dim' merupakan tabel dimensi dan 'Fact' yang merupakan tabel Fakta.

Berdasarkan Foreign Key yang terdapat pada tabel 'FactSalesOrder', dapat dibilang bahwa skema pada database ini adalah Star Schema.

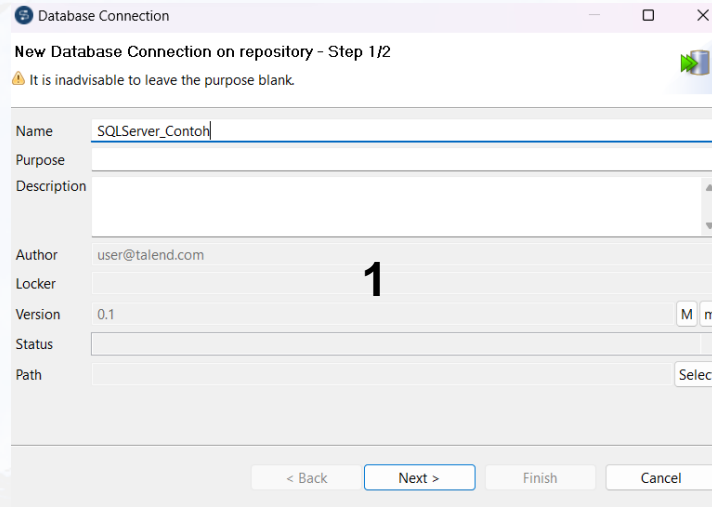


Query bisa dilihat [disini](#)

Case Study

Setelah dibuatnya database untuk data warehousing, langkah selanjutnya adalah membuat job di salah satu ETL Studio, yaitu Talend.

Sebelum dibuat Job Designs, langkah pertama adalah membuat koneksi dari SQL Server Management Studio ke Talend. Dibutuhkan dua koneksi database, yaitu database sumber dan database untuk data warehousing.



Database Connection

New Database Connection on repository - Step 1/2

It is inadvisable to leave the purpose blank.

Name: SQLServer_Contoh

Purpose:

Description:

Author: user@talend.com

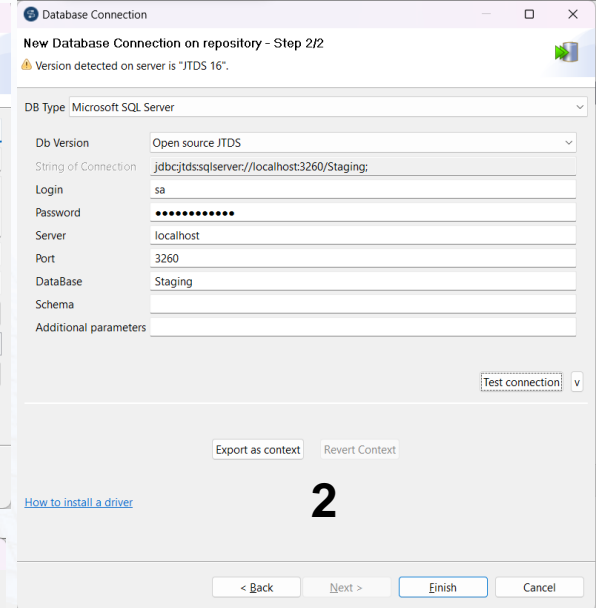
Locker:

Version: 0.1

Status:

Path:

Next >



Database Connection

New Database Connection on repository - Step 2/2

Version detected on server is "JTDS 16".

DB Type: Microsoft SQL Server

Db Version: Open source JTDS

String of Connection: jdbc:tds:sqlserver://localhost:3260/Staging;

Login: sa

Password:

Server: localhost

Port: 3260

DataBase: Staging

Schema:

Additional parameters:

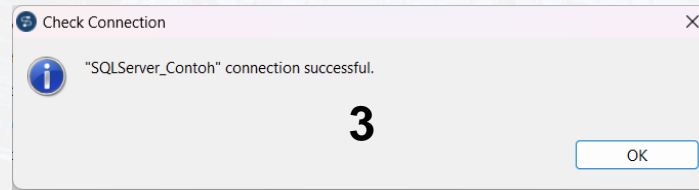
Test connection

Export as context

Revert Context

How to install a driver

Next >



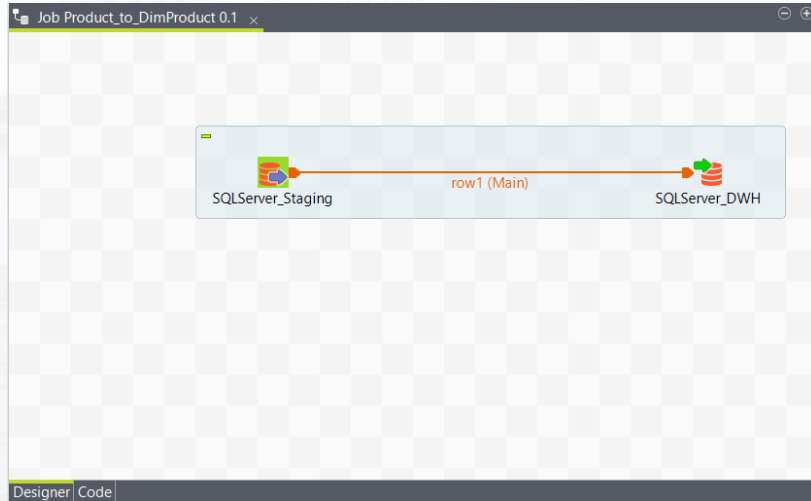
Check Connection

"SQLServer_Contoh" connection successful.

OK

Case Study

Setelah dibuatnya koneksi, kemudian melakukan pembuatan Job Designs.



Schema of SQLServer_DWH

SQLServer_Staging (Input - Main)

Column	Db Column	K...	Type	DB Ty...	✓	N...	Date Pa...	Len...	Prec...	D...	Co...
product_id	ProductID	✓	Int	INT				10	0		
product_na...	ProductName		Stri...	VARC...				255	0		
product_ca...	ProductCategory		Stri...	VARC...				255	0		
product_uni...	ProductUnitPrice		Int...	INT		✓		10	0		

SQLServer_DWH (Output)

Column	Db Column	K...	Type	DB Ty...	✓	N...	Date Pa...	Len...	Prec...	D...	Co...
product_id	ProductID	✓	Int	INT				10	0		
product_name	ProductName		Stri...	VARC...				255	0		
product_cate...	ProductCategory		Stri...	VARC...				255	0		
product_uni...	ProductUnitPrice		Int...	INT		✓		10	0		

Job(Product_to_DimProduct 0.1) Contexts(Product_to_DimProduct) Component Run (Job Product_to_DimProduct)

SQLServer_Staging(tDBInput_1)(Microsoft SQL Server)

Basic settings

Username: "sa" Password: *****

Schema: Repository DB (MSSQL):SQLServer_Staging - pi Edit schema

Table Name: "product" The variable attached to this parameter is: _TABLE

Query Type: Built-In Guess Query Guess schema

Query: "SELECT * FROM product"

Data source: This option only applies when deploying and running in the Talend Runtime

☐ Specify a data source alias

SQLServer_DWH(tDBOutput_1)(Microsoft SQL Server)

Basic settings

Host: "localhost" Port: "3260" Schema: ""

Database: "DWH_Project"

Username: "sa" Password: *****

Table: "DimProduct"

Action on table: Default Turn on identity insert Action on data: Insert

Schema: Built-In Edit schema Sync columns

Data source: This option only applies when deploying and running in the Talend Runtime

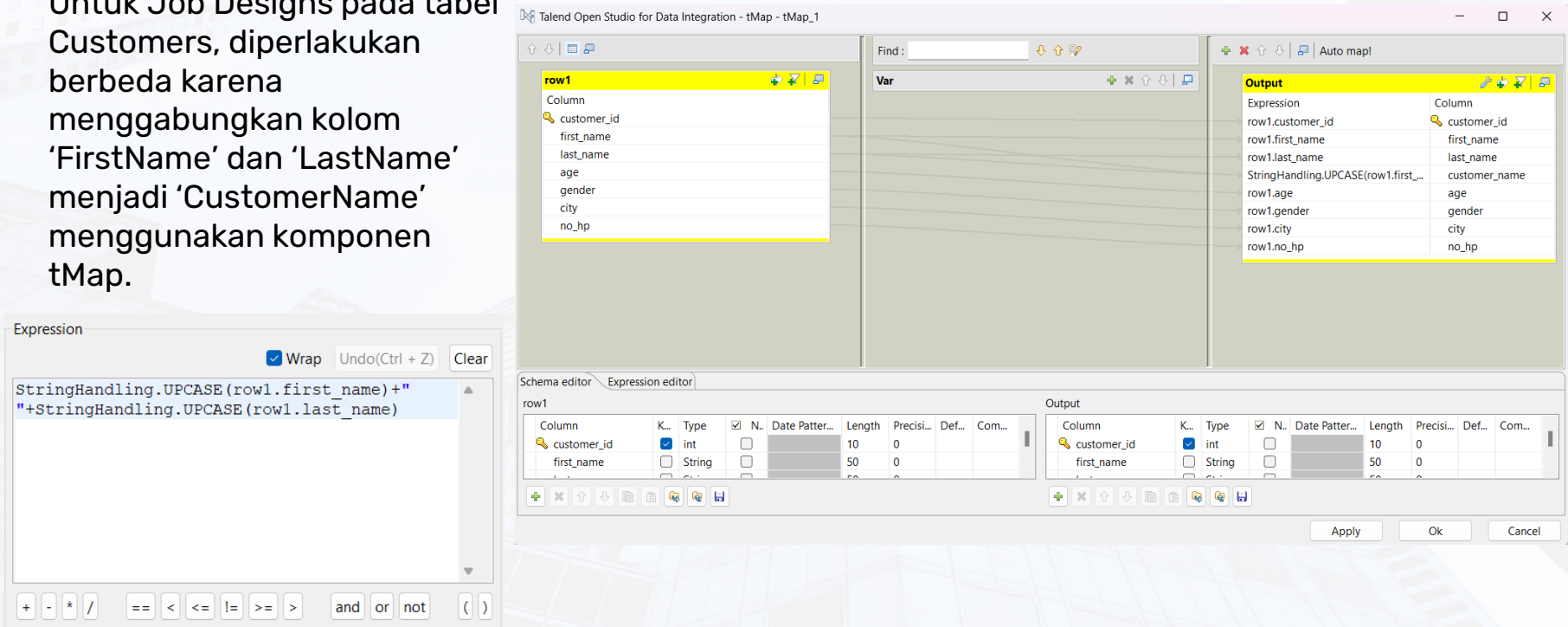
☐ Specify a data source alias

☐ Die on error

OK Cancel

Case Study

Untuk Job Designs pada tabel Customers, diperlakukan berbeda karena menggabungkan kolom 'FirstName' dan 'LastName' menjadi 'CustomerName' menggunakan komponen tMap.



The screenshot displays the Talend Open Studio for Data Integration interface, specifically the tMap component configuration. The main window is titled "Talend Open Studio for Data Integration - tMap - tMap_1".

Left Panel (Schema editor): Shows the input schema "row1" with the following columns:

Column
customer_id
first_name
last_name
age
gender
city
no_hp

Bottom Left Panel (Expression editor): Shows the expression for the "CustomerName" output column:

```
StringHandling.UPCASE(row1.first_name)+"  
"+StringHandling.UPCASE(row1.last_name)
```

Right Panel (Auto map!): Shows the output schema "Output" with the following columns:

Expression	Column
row1.customer_id	customer_id
row1.first_name	first_name
row1.last_name	last_name
StringHandling.UPCASE(row1.first_...	customer_name
row1.age	age
row1.gender	gender
row1.city	city
row1.no_hp	no_hp

Bottom Right Panel (Schema editor): Shows the output schema "Output" with the following columns:

Column	K...	Type	✓	N.	Date Patter...	Length	Precisi...	Def...	Com...
customer_id	int	int	✓			10	0		
first_name	String	String				50	0		

Case Study

Challenge selanjutnya membuat Stored Procedure yang akan menampilkan tabel yang berisi OrderID, CustomerName, ProductName, Quantity, dan StatusOrder berdasarkan StatusID yang akan dikeluarkan.

Pada tabel dibawah merupakan hasil panggil Stored Procedure dengan StatusID = 2.

Query bisa dilihat [disini](#)

Results		Messages			
	OrderID	CustomerName	ProductName	Quantity	StatusOrder
1	1301	LIA RAHMAWATI	Converse Cap Original	2	Awaiting Shipment
2	1304	AJENG SRIASIH	T-Shirt Polo Nevada	2	Awaiting Shipment
3	1307	RAHMA AMELIA	Pull & Bear T-Shirt	1	Awaiting Shipment

```
SQLQuery1.sql - L...IUHHQ\VICTUS (63))* X
CREATE OR ALTER PROCEDURE summary_order_status
(@StatusID int) AS
BEGIN
    SELECT
        a.OrderID,
        b.CustomerName,
        c.ProductName,
        a.Quantity,
        e.StatusOrder
    FROM FactSalesOrder a
    JOIN DimCustomer b ON a.CustomerID = b.CustomerID
    JOIN DimProduct c ON a.ProductID = c.ProductID
    JOIN DimStatusOrder e ON a.StatusID = e.StatusID
    WHERE a.StatusID = @StatusID
END

EXEC summary_order_status @StatusID = 1
```


GitHub Link

<https://github.com/FawwazN/data-engineer-idx>

Video Presentation Here

<https://drive.google.com/file/d/1-IYAhCnHmV6nKhYrxNGNHLsrb43HdZJH/view?usp=sharing>

Thank You

