**PRACTICAL 1**

**VISUALIZE A DATASET WITH ANOMALIES**

**USING RAPIDMINER AND NEO4J**

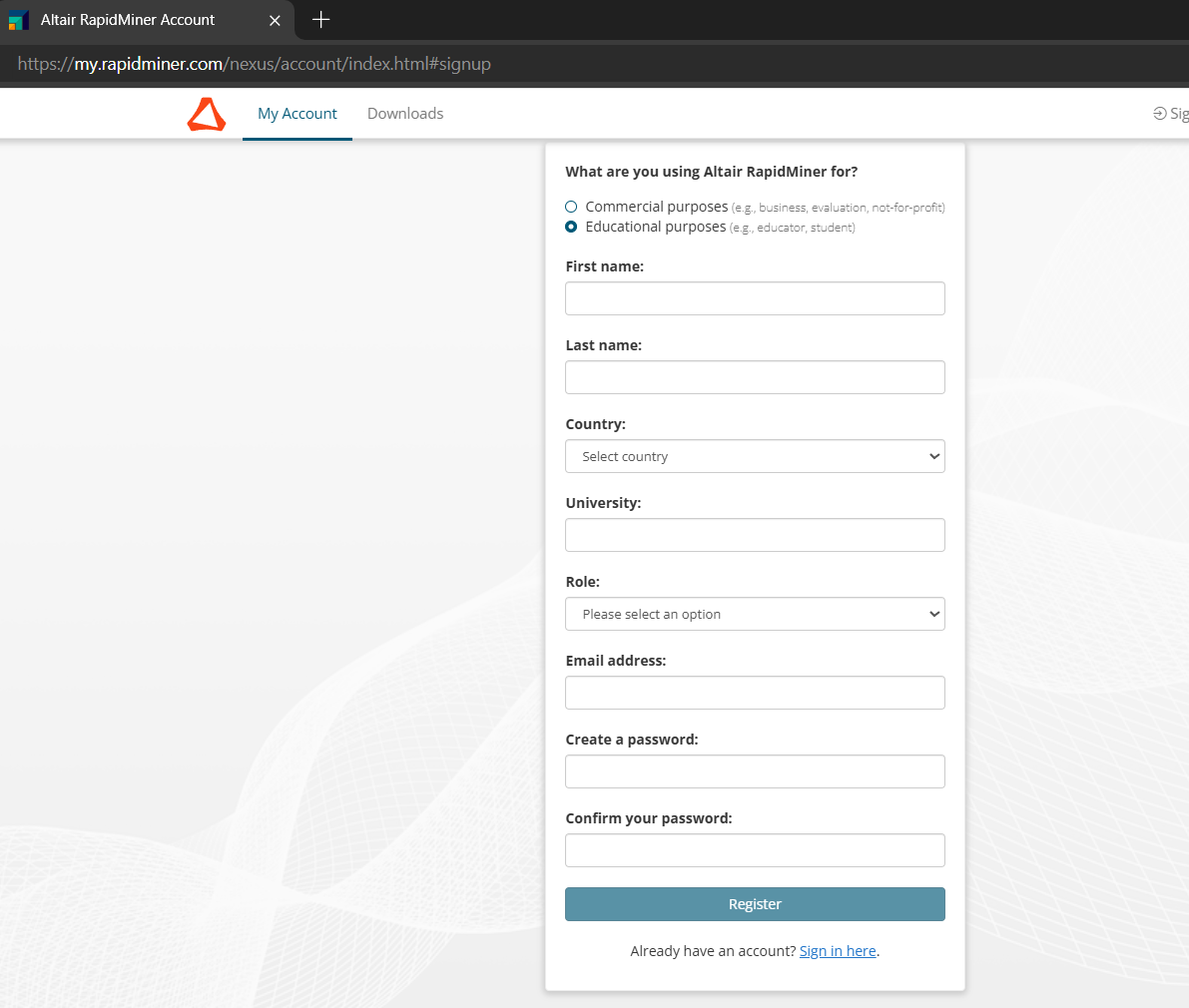
**[A] INSTALLING RAPIDMINER**

1. Go to Rapidminer by Altair downloads page <https://my.rapidminer.com/nexus/account/index.html>
2. If already have credentials click on Downloads tab and download it according to your OS

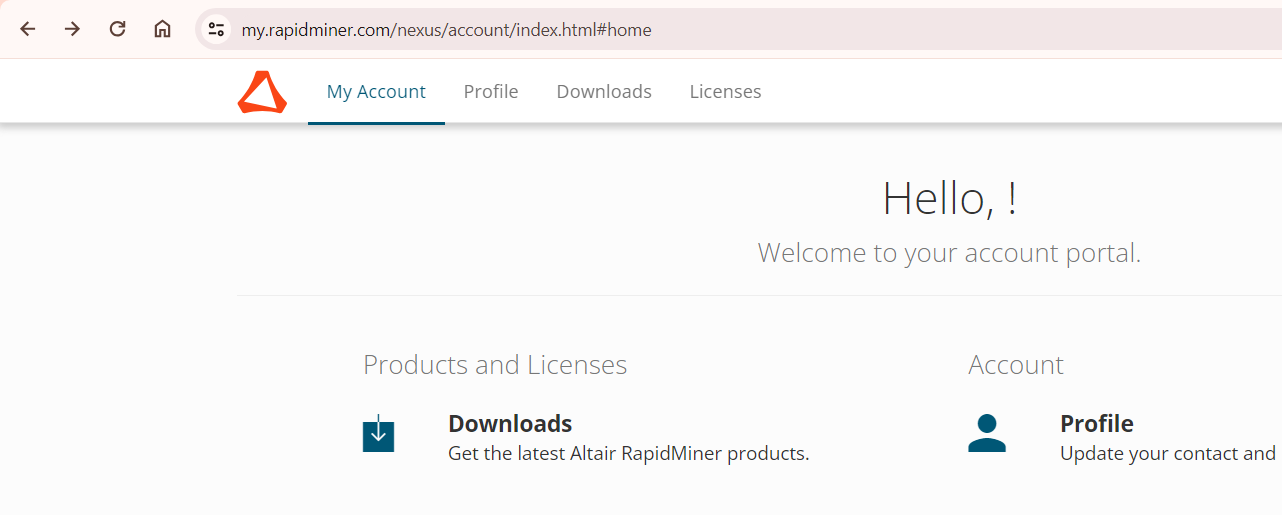
OR

If new then click on My Account Tab and create a new account (Required in further steps)

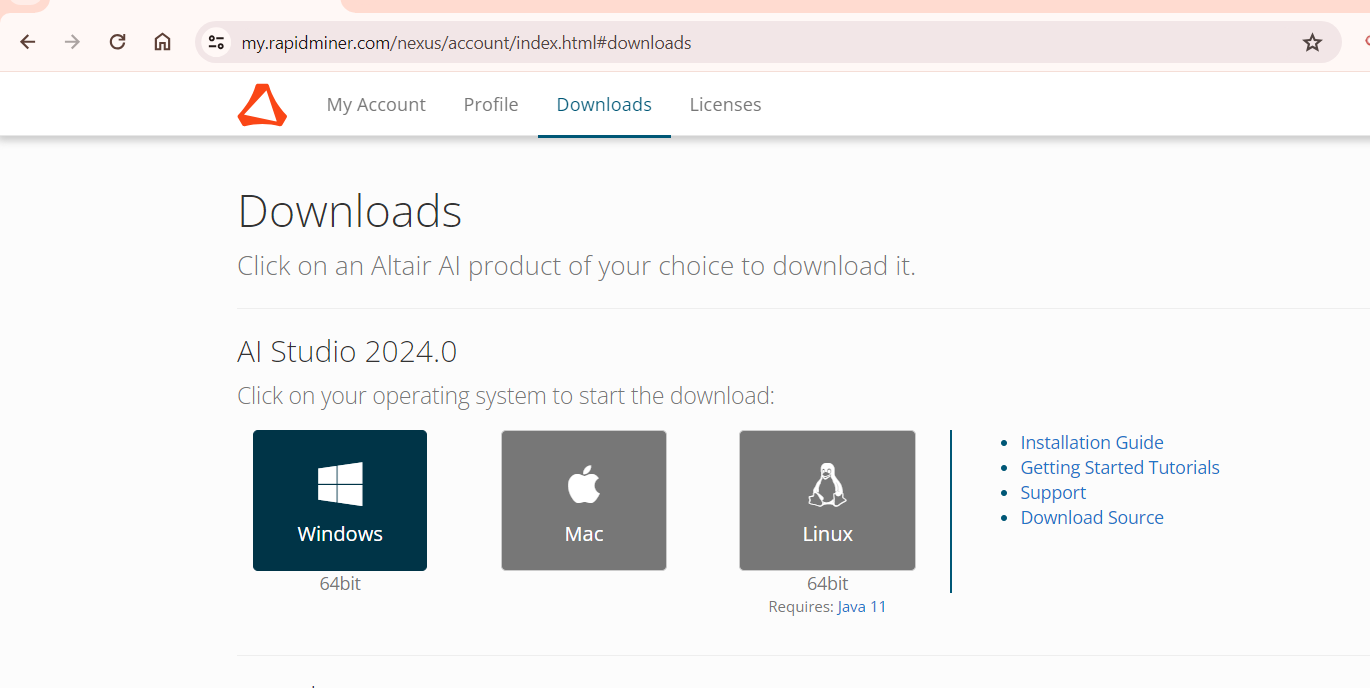
1. Verify your mail



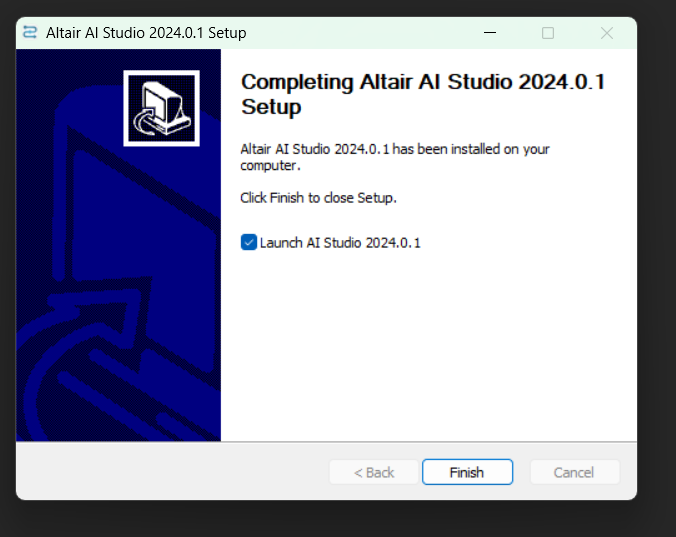
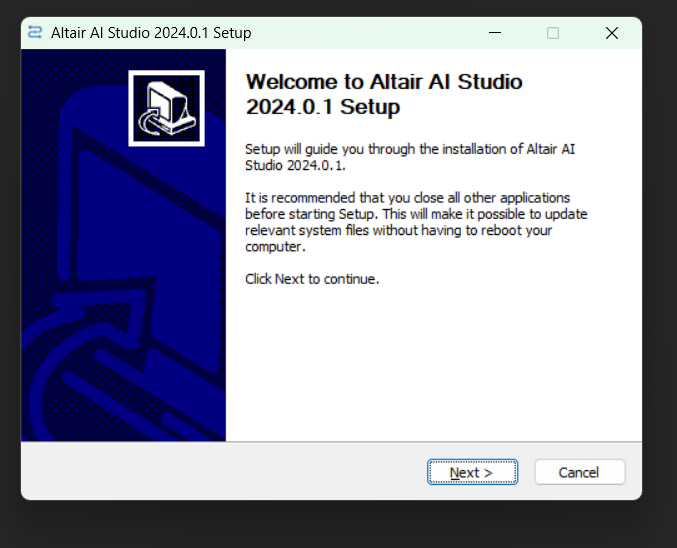
1. After verfiying mail you’ll be redirected to accounts portal click on downloads



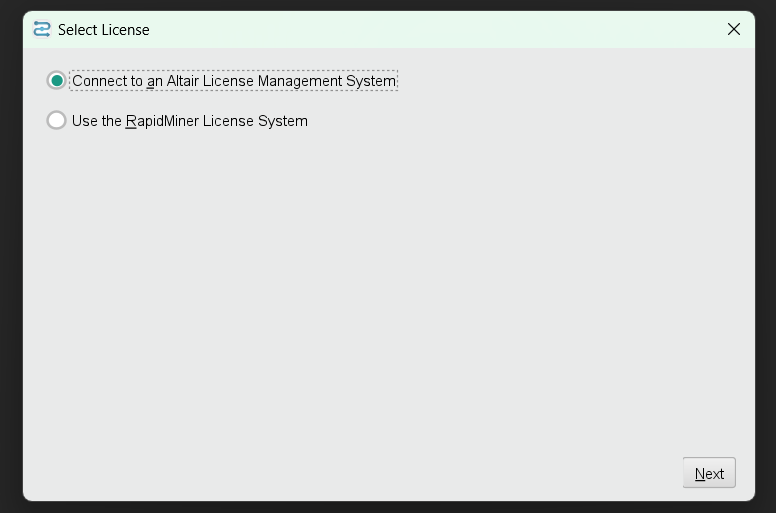
1. Download the installer



1. After download is complete open your exe file and and install it
2. By default click on next until finish

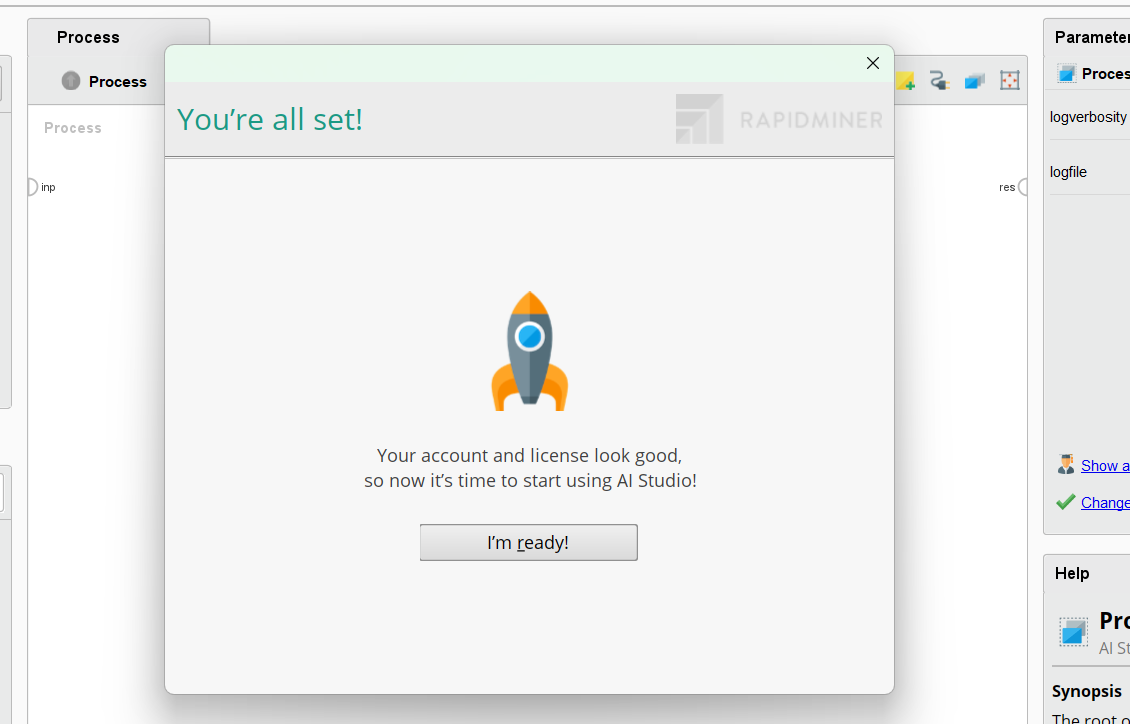
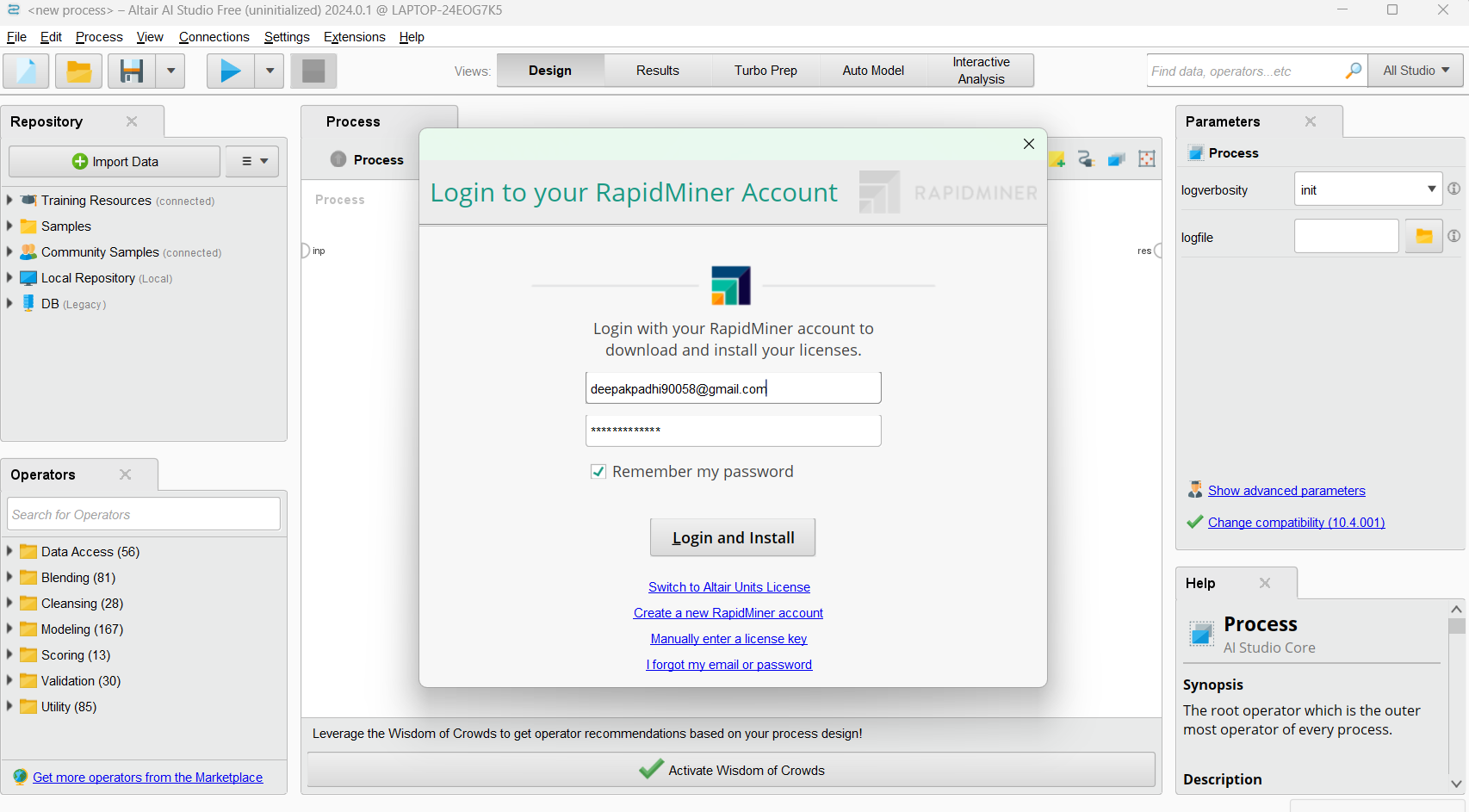


1. A prompt will come to select licence click on “use the RapidMiner Liscence System” and next



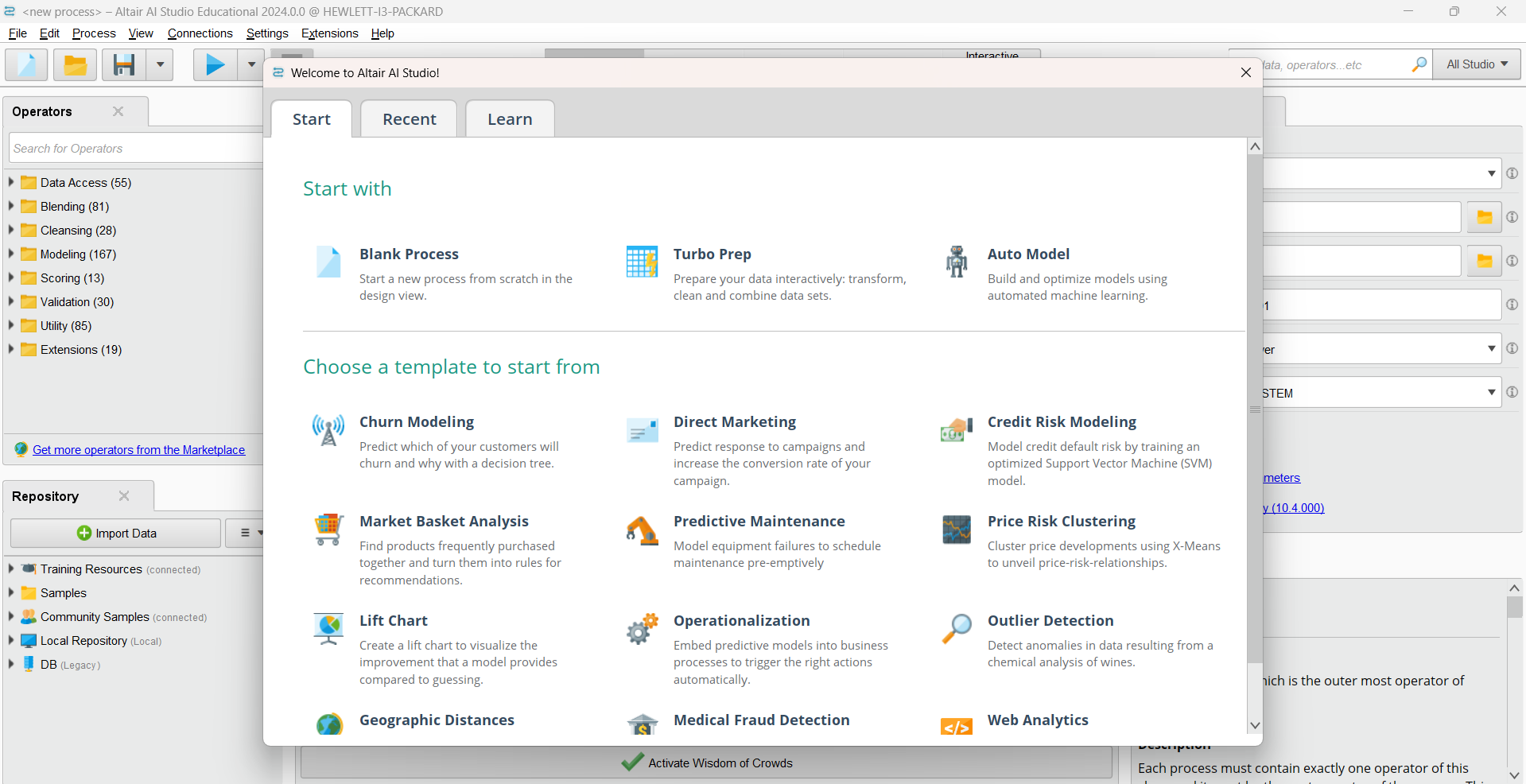


1. After the app gets starts it will ask you to enter the credentials that was generated while creating your account
2. After enetering your credentials proceed to login and install & I’m ready

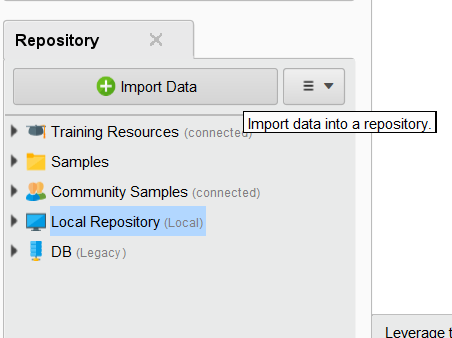


**[B] ANOMALY DETECTION WITH RAPIDMINER**

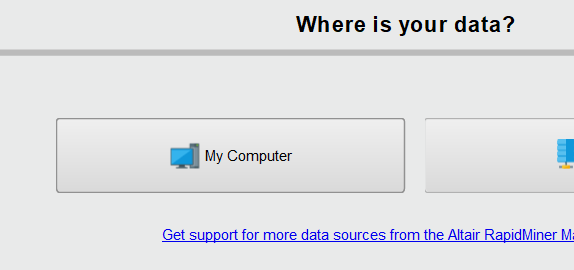
1. Click on Blank Process



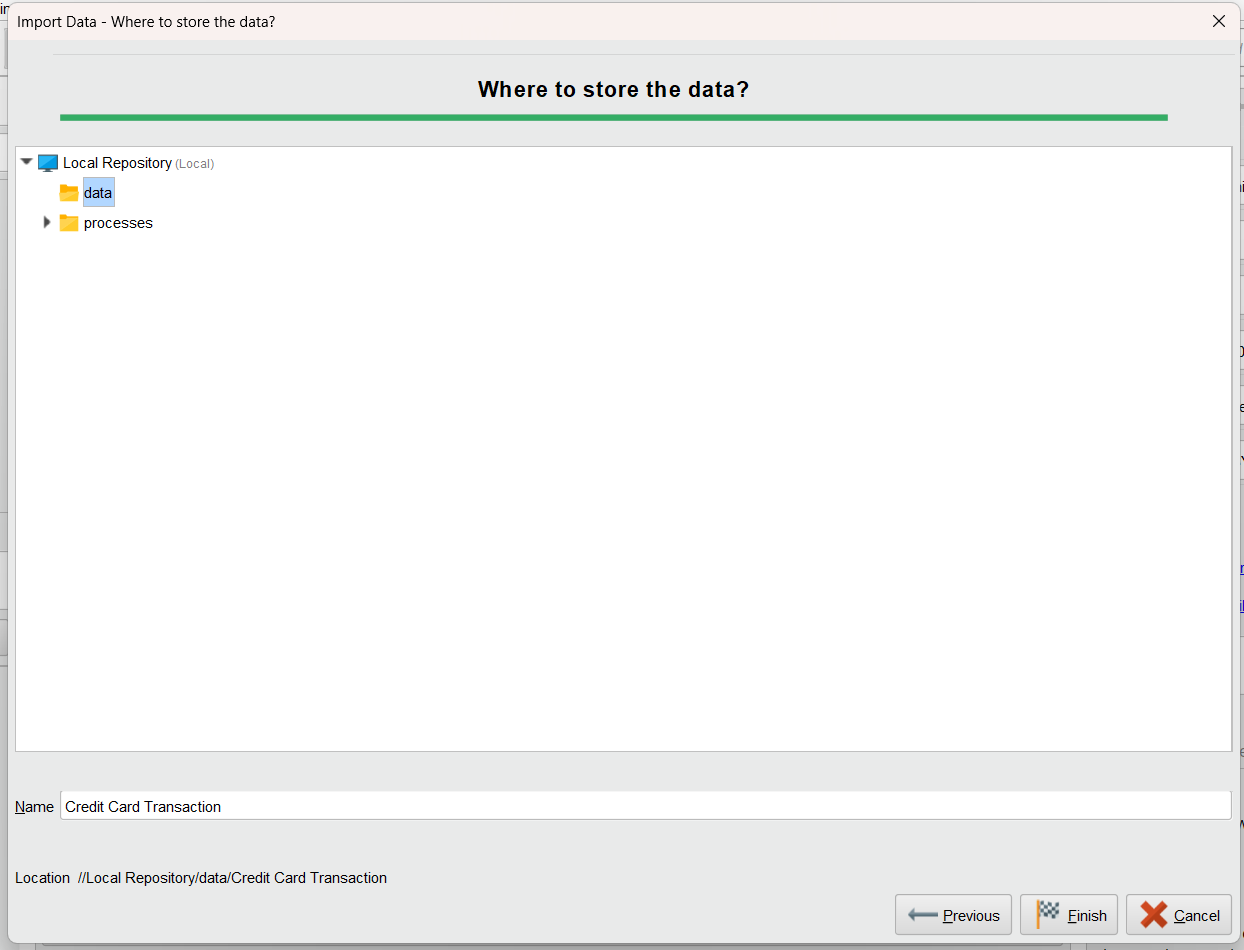
1. Import your dataset into RapidMiner by clicking on “import data”



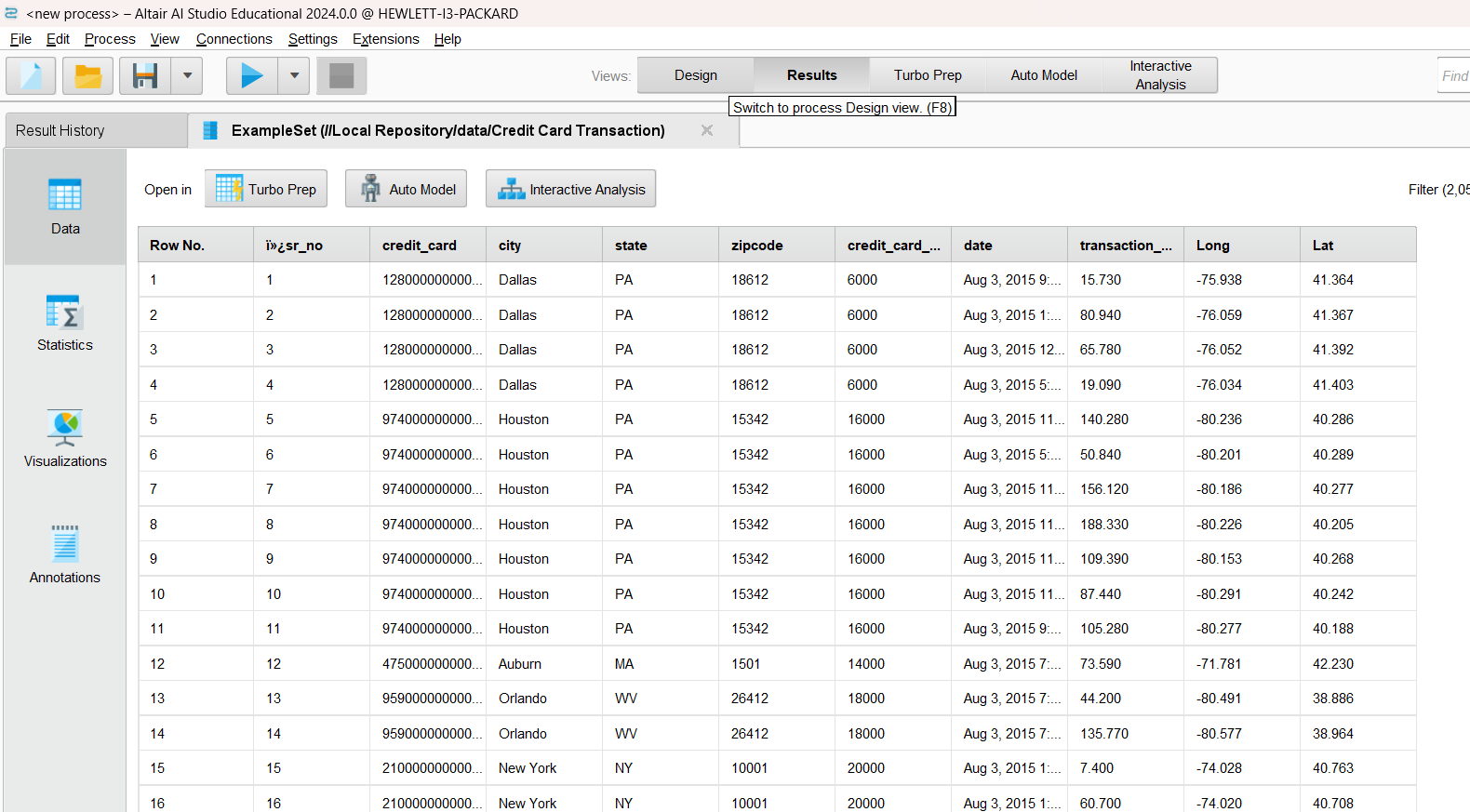
1. Select “My Computer” for choosing the dataset



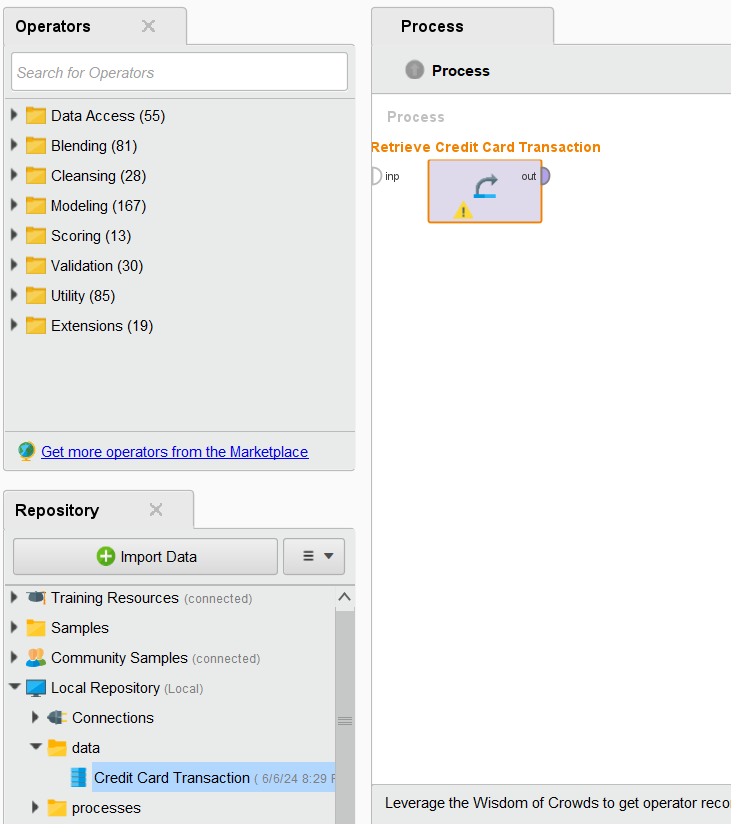
1. Download the Credit Card Transaction csv file <https://drive.google.com/file/d/1xeejp6XcAwpwVwsqNmOhDBuO6E4UE9y9/view?usp=sharing>
2. Browse through your location and select the dataset called Credit Card Transaction.csv and click on next if don’t want to change anything
3. In where to store data under the local repository click on data and can rename your file then click on finish



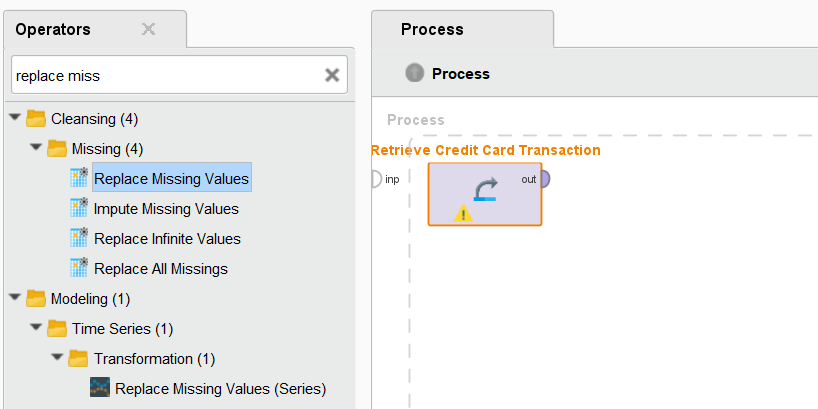
1. If result tab opens and your data shows correctly means import was successful
2. Switch to Design tab



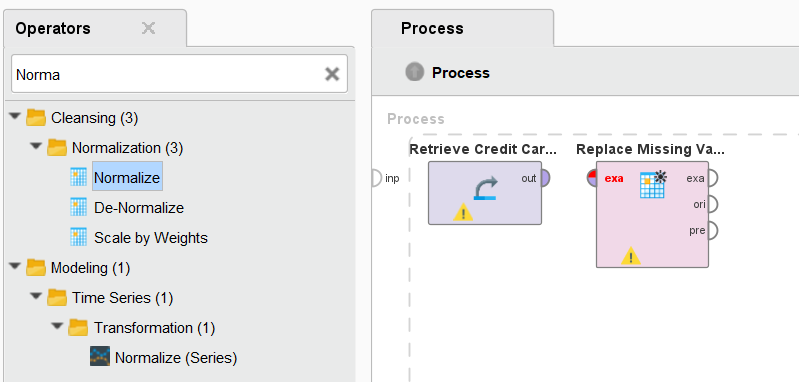
1. Drag and drop your dataset from local repository into the process



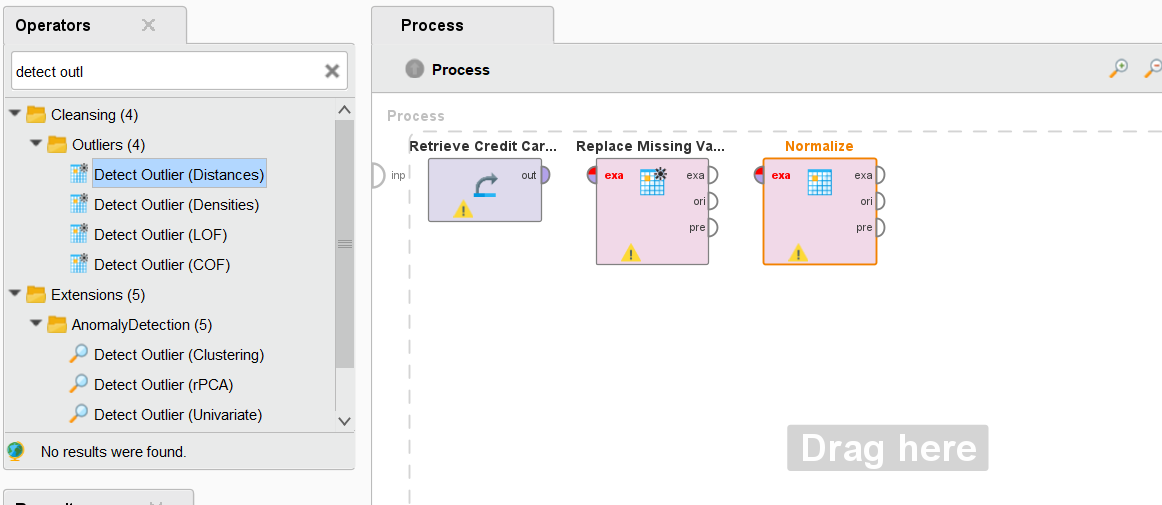
1. Now go to operators panel and search for replace missing values then drag and drop it into the process (It is used to check and replaces any missing values by default it is set average value will be given you can change it by clicking on that operator)



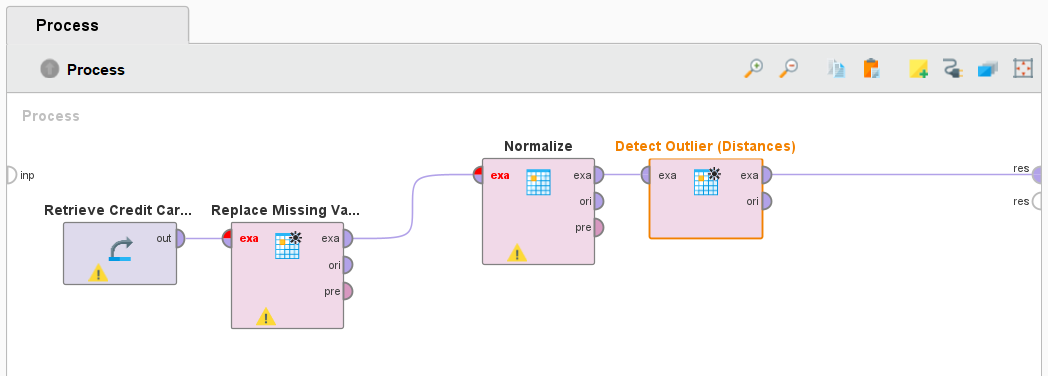
1. Now go to operators panel and search for Normalize then drag and drop it into the process (It is used to scale values so they fit in a specific range.)



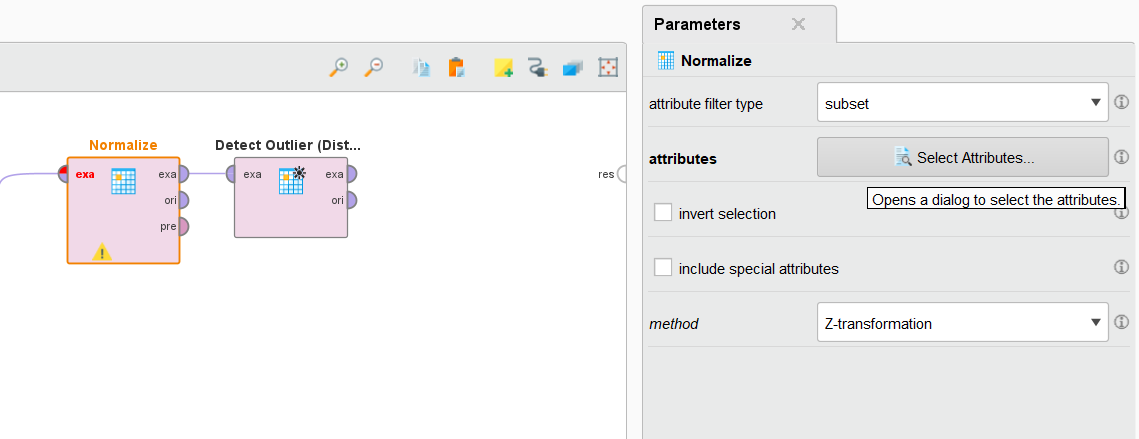
1. Now go to operators panel and search for detect outlier distances then drag and drop it into the process (It will detect anomalies in the dataset and create a new column for it stating TRUE OR FALSE)



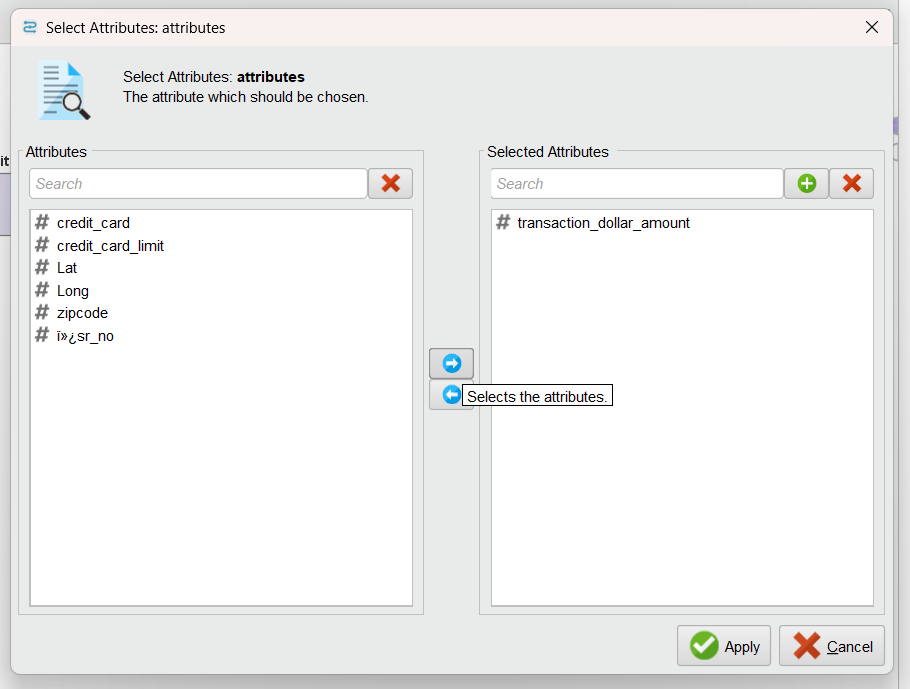
1. Connect all the port from dataset to the result port by clicking from on port to another



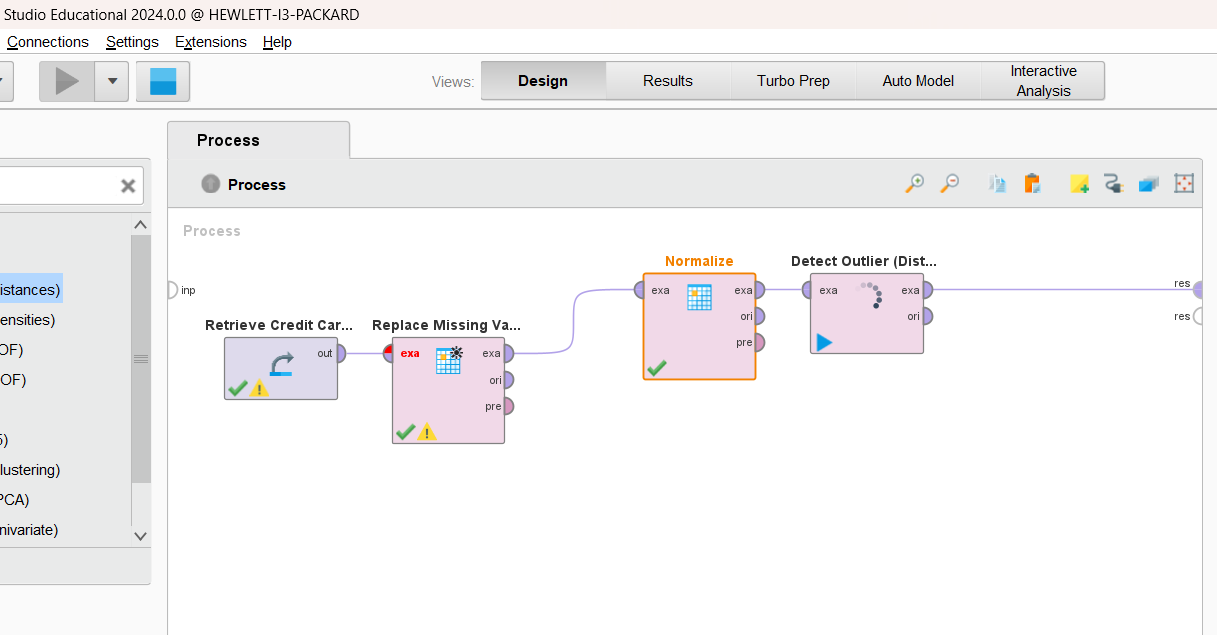
1. Before running click on the normalize operator and in the parameters panel select the attribute filter type to subset and in below click on select attributes



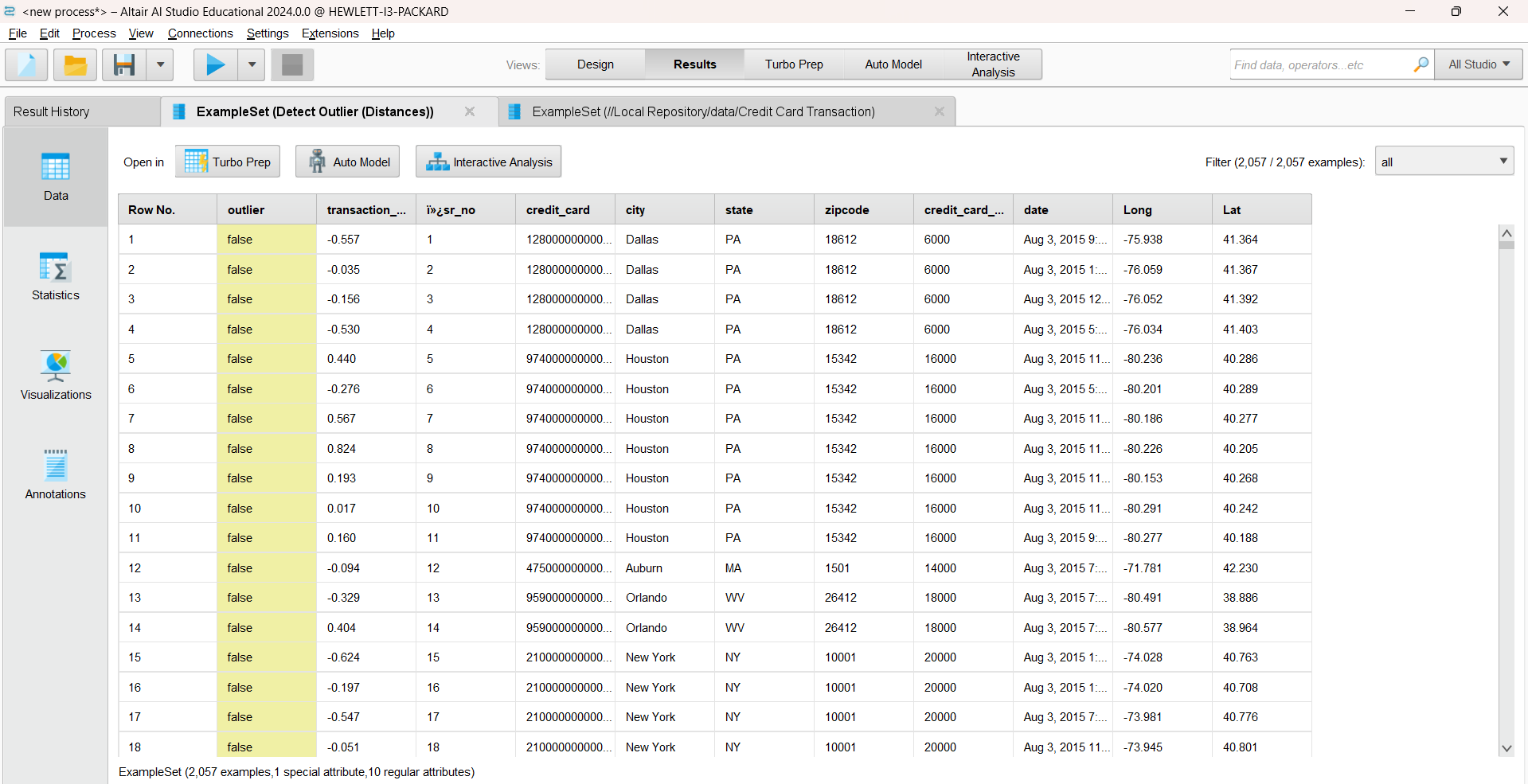
1. In this move the transaction\_dollar to selected attributes



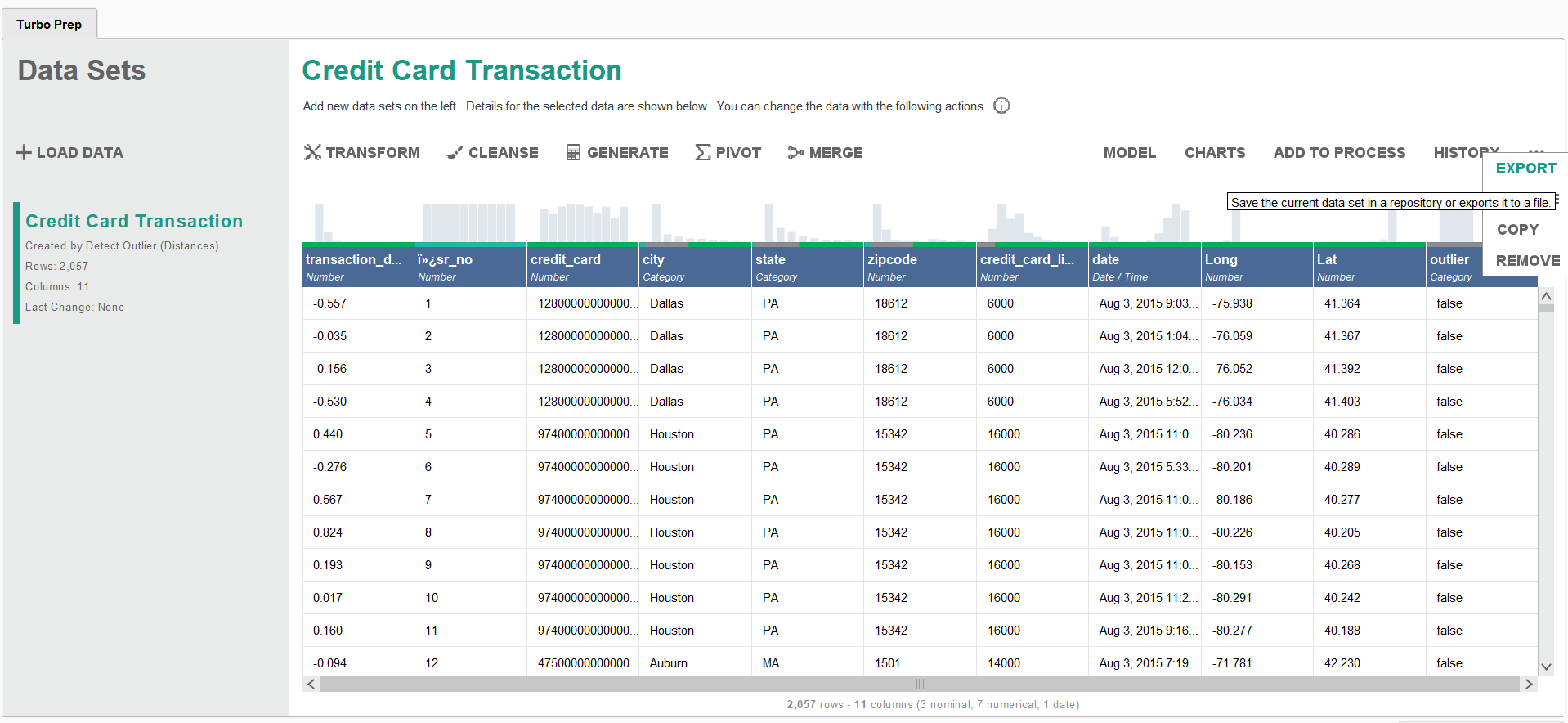
1. Now click on run (play button) to run the process



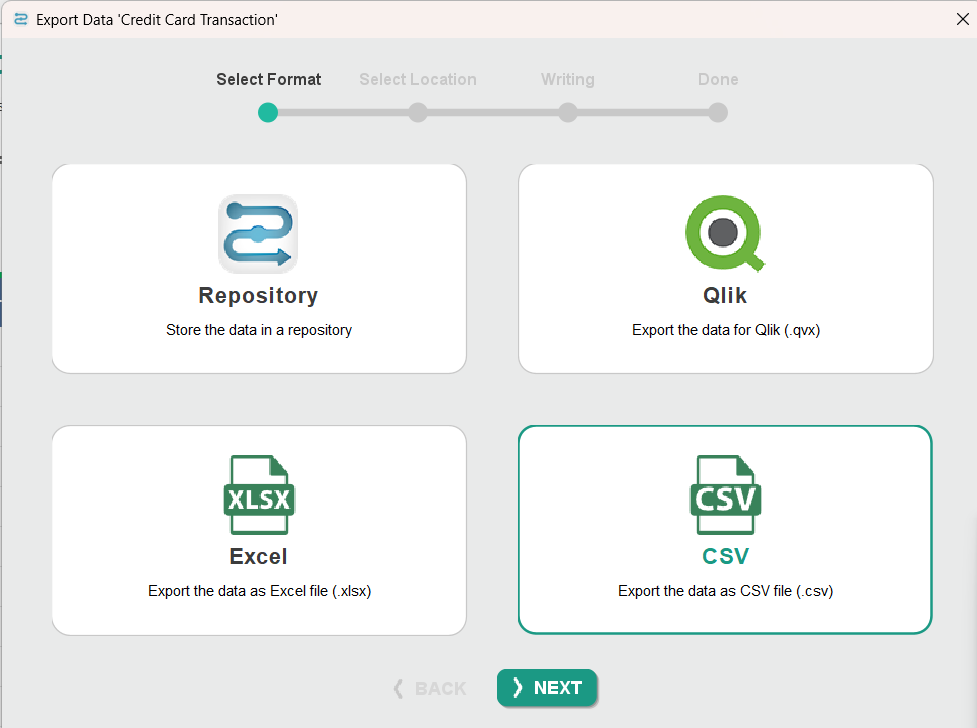
1. After it runs successfully it will show the results you can check out the statistics and visualization panels to see insights into it, also a new column shows whether the row data has any outlier or not
2. Click on the Turbo prep button above the data set



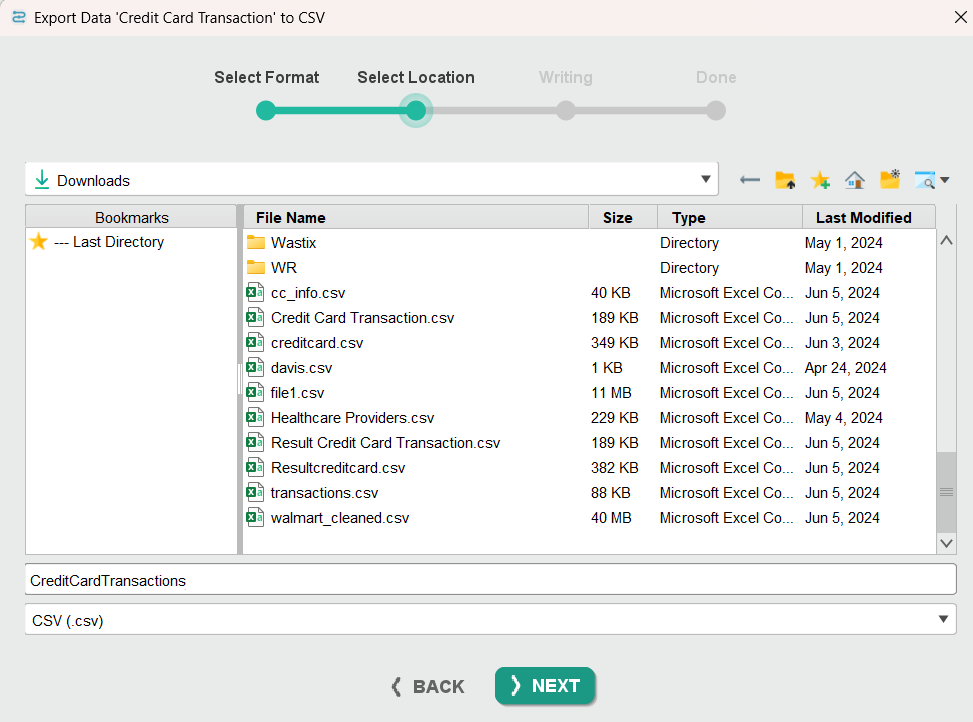
1. In the right side click on **:** menu and click on EXPORT



1. Select format as csv

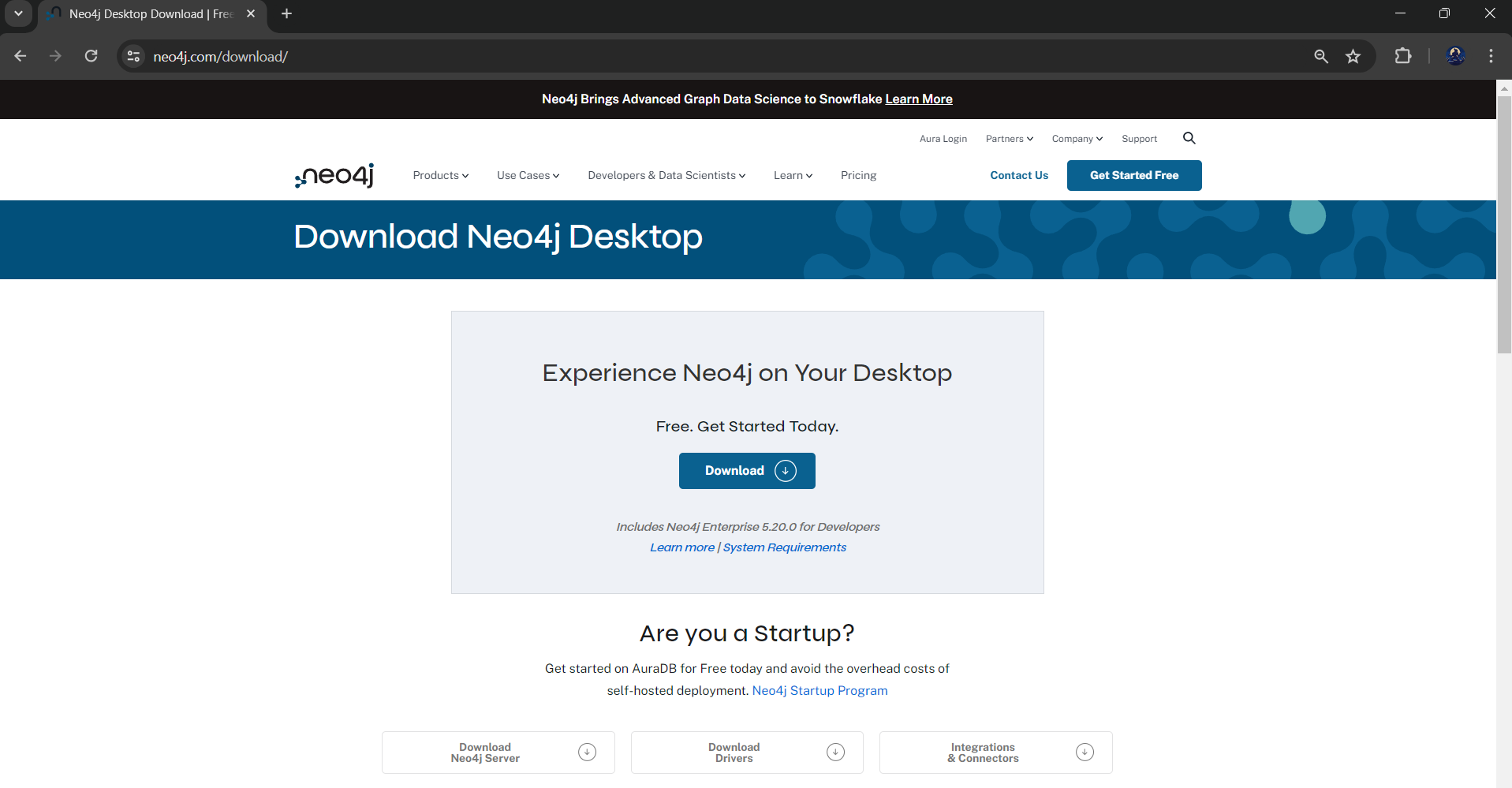


1. Select the location you want to save and save it as CreditCardTransactions

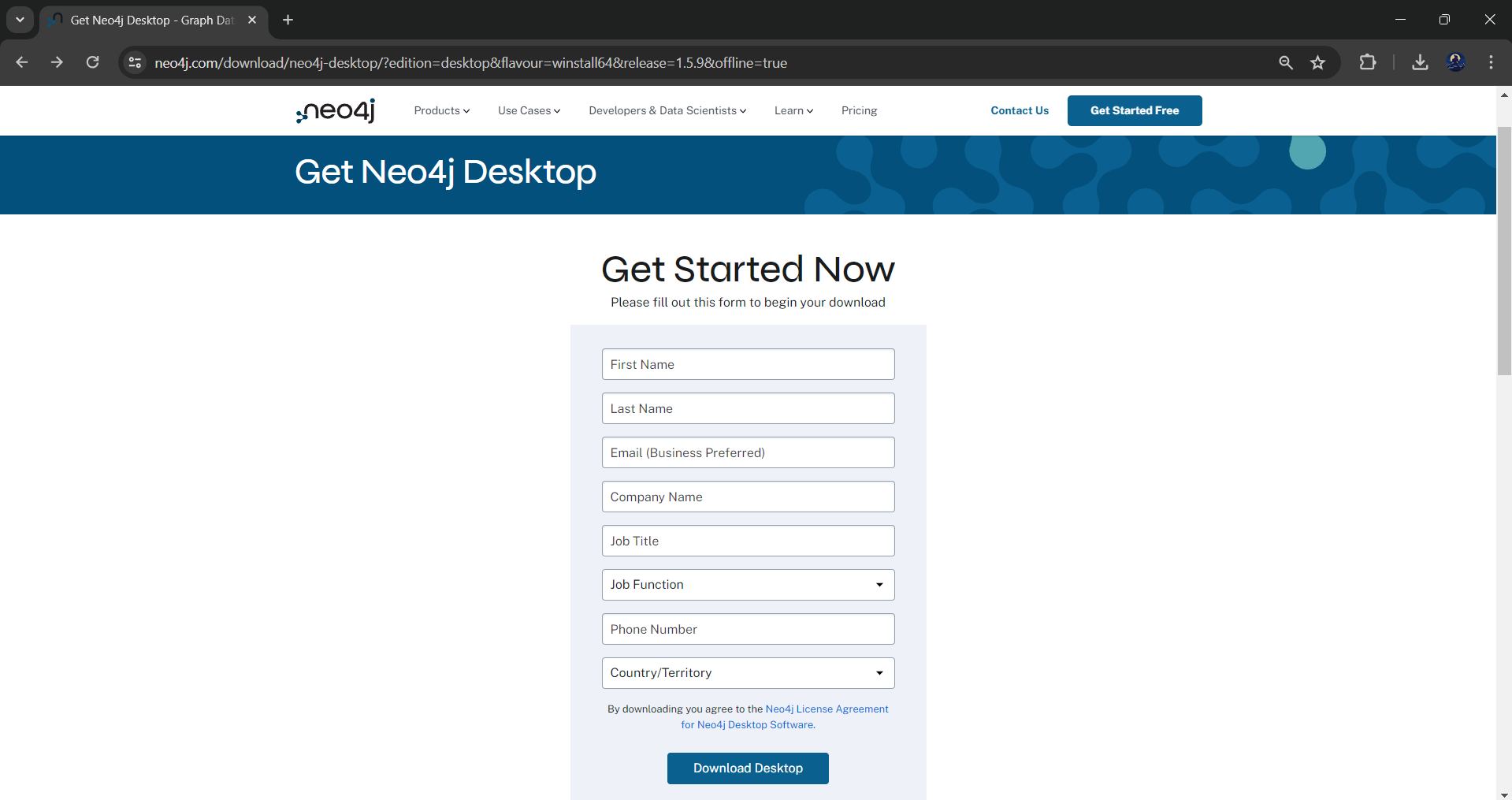


**[C] INSTALLING NEO4J**

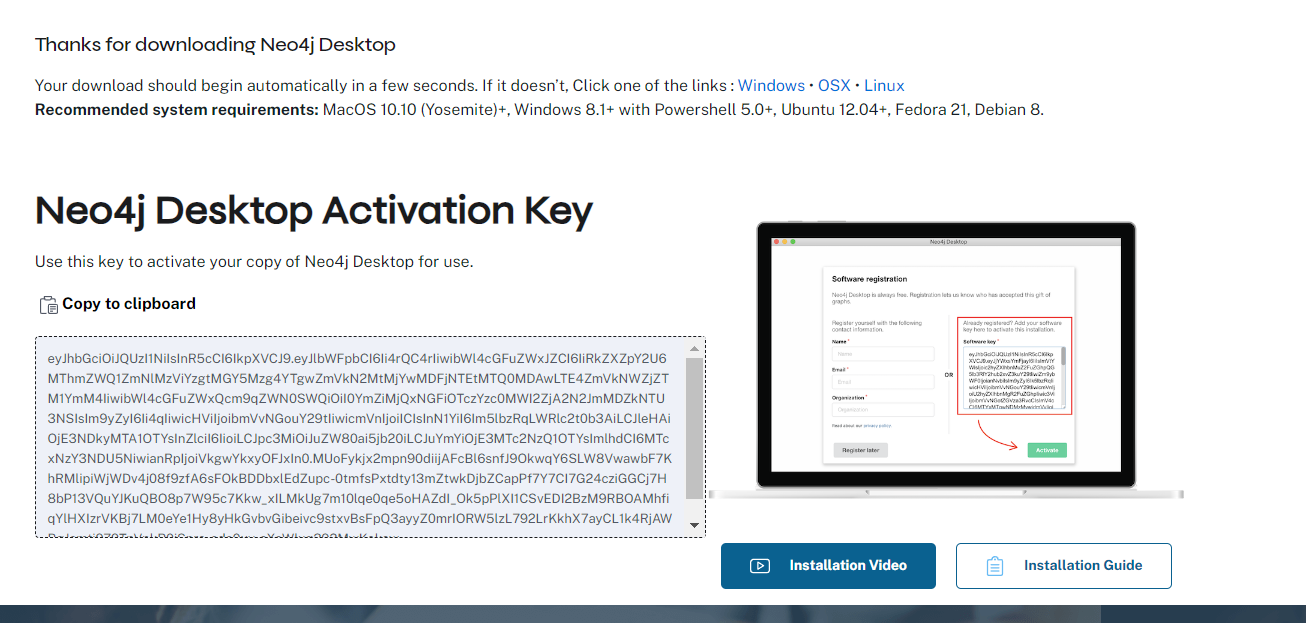
1. Go to neo4j downloads page and click on download <https://neo4j.com/download/>



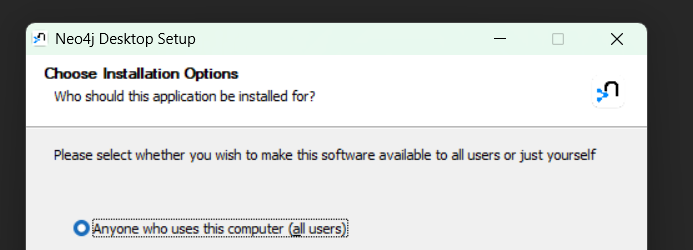
1. Fill in the details and click on download



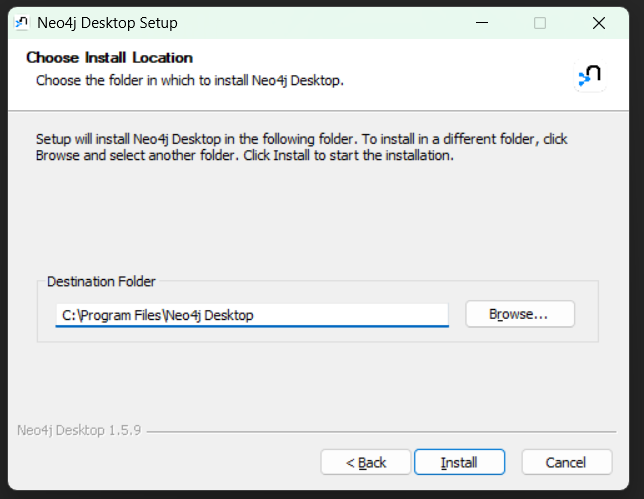
1. After clicking on download it will provide an activation key copy it and save it for further steps

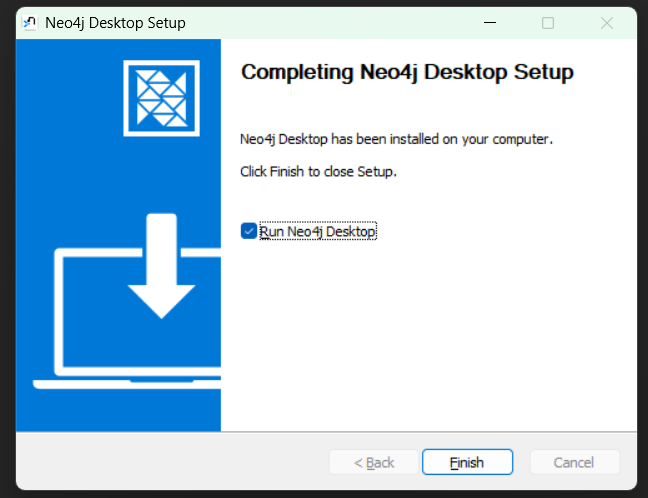


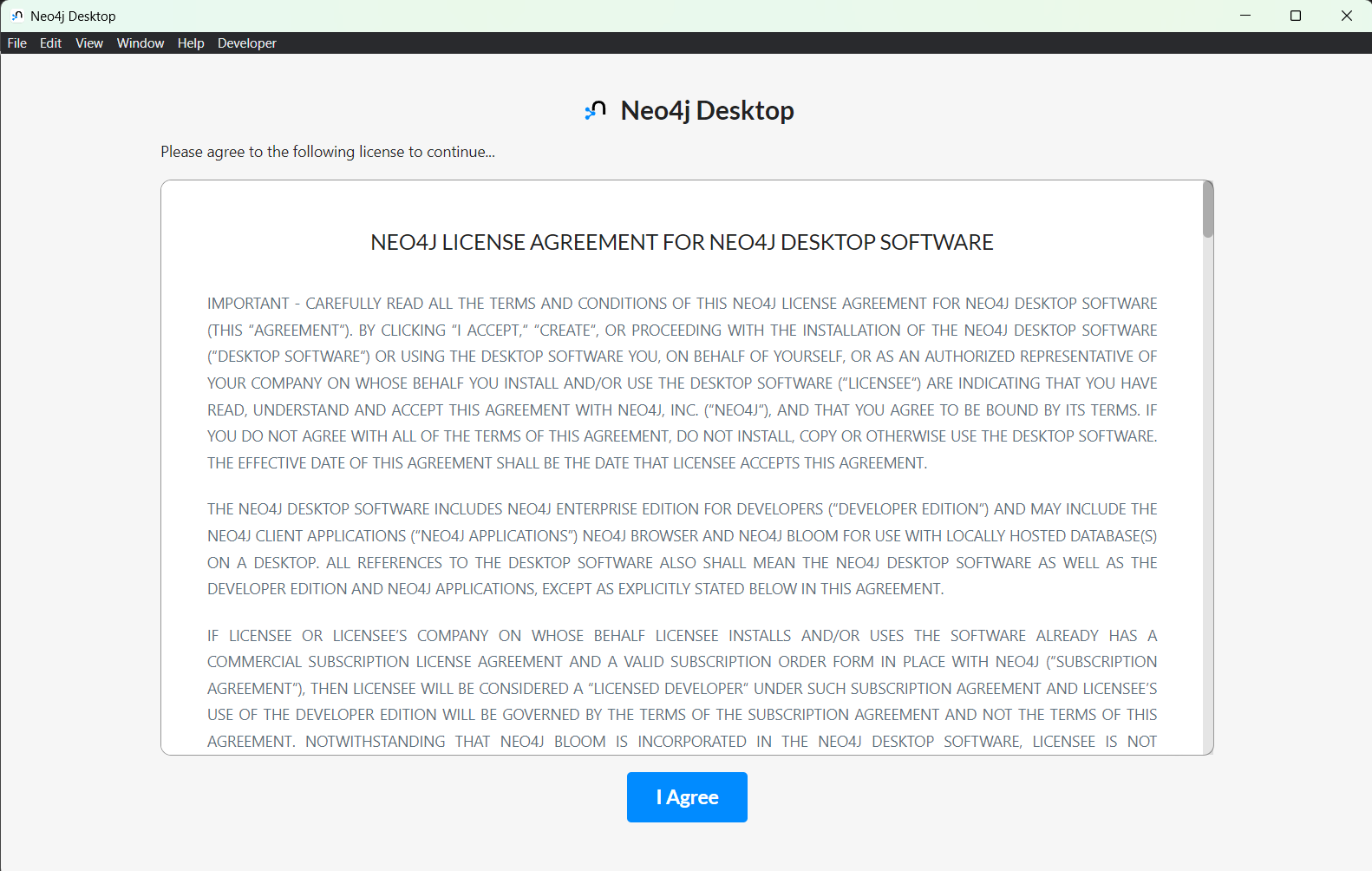
1. Open the exe file and start the installation, for installation option Choose for all users



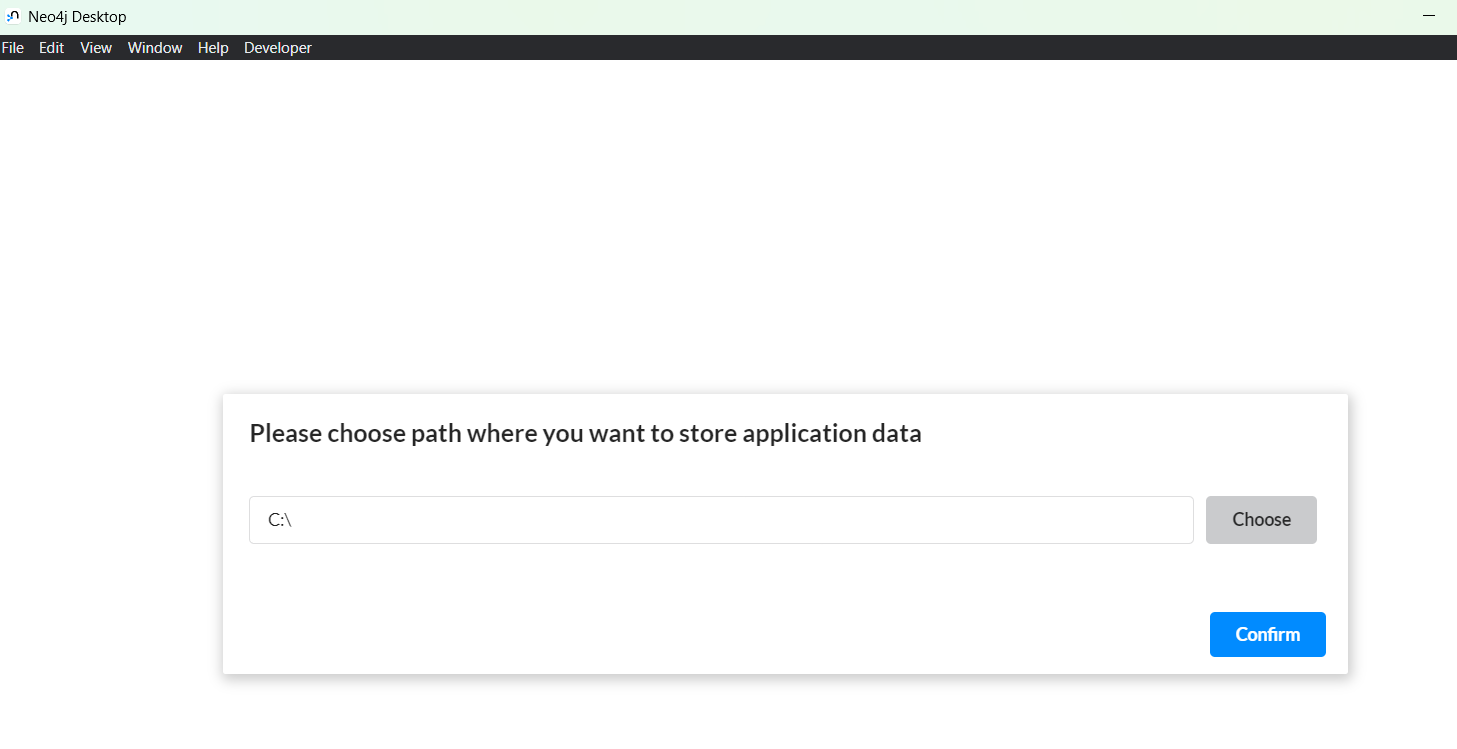
1. Select the download location and install it then click on finish to launch the application



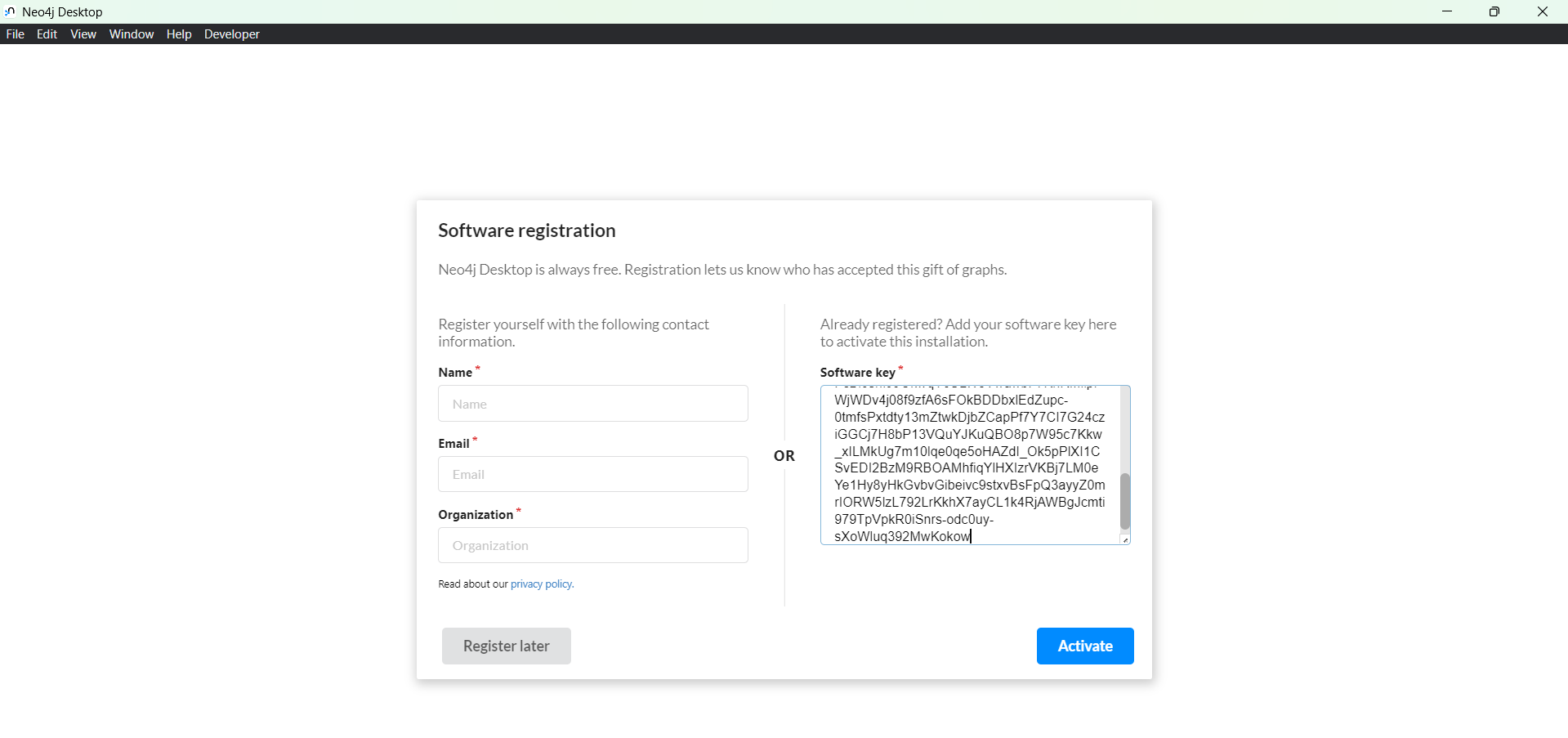




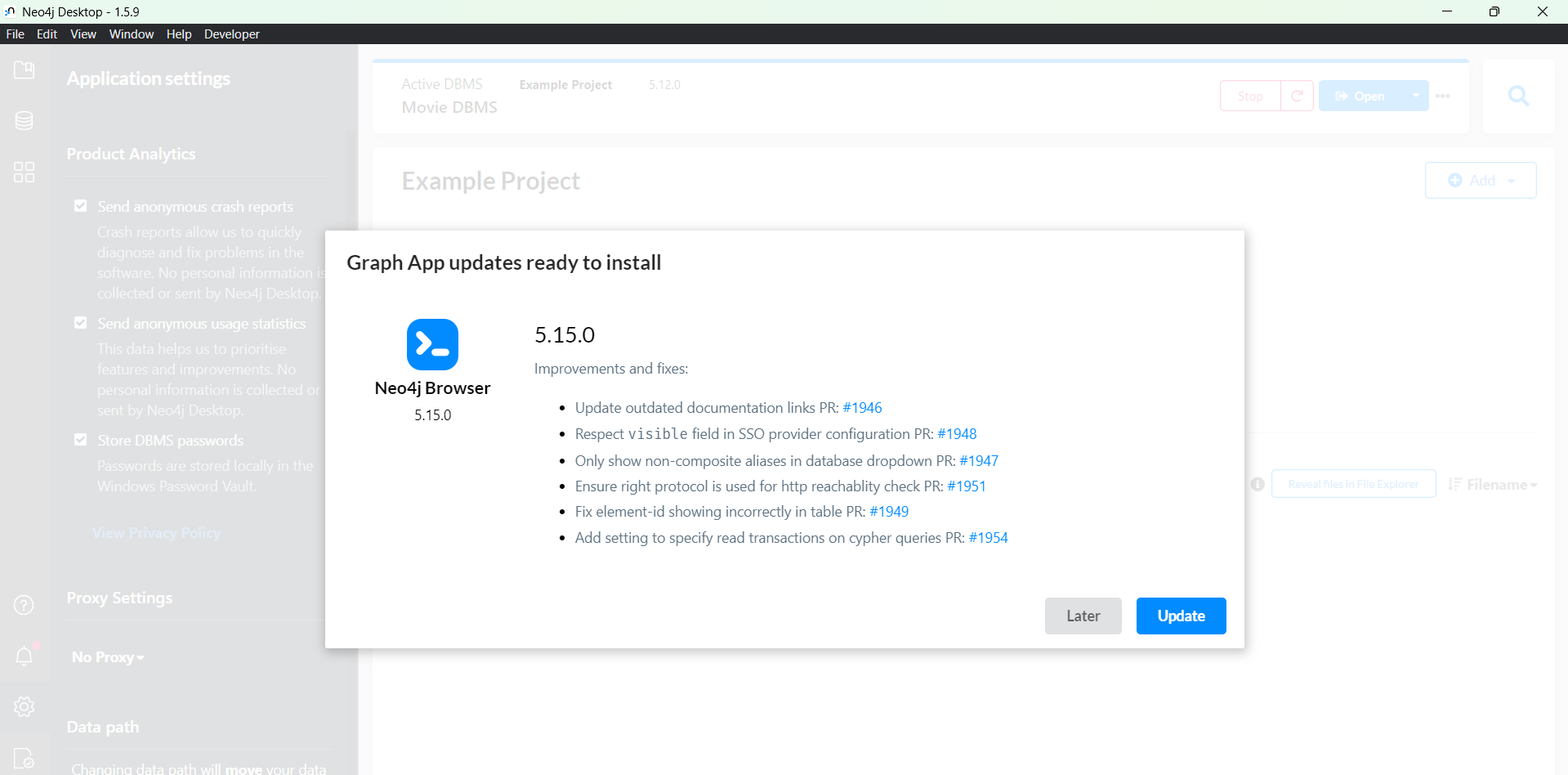
1. Not the path where you are saving the application data it will be required to import the dataset



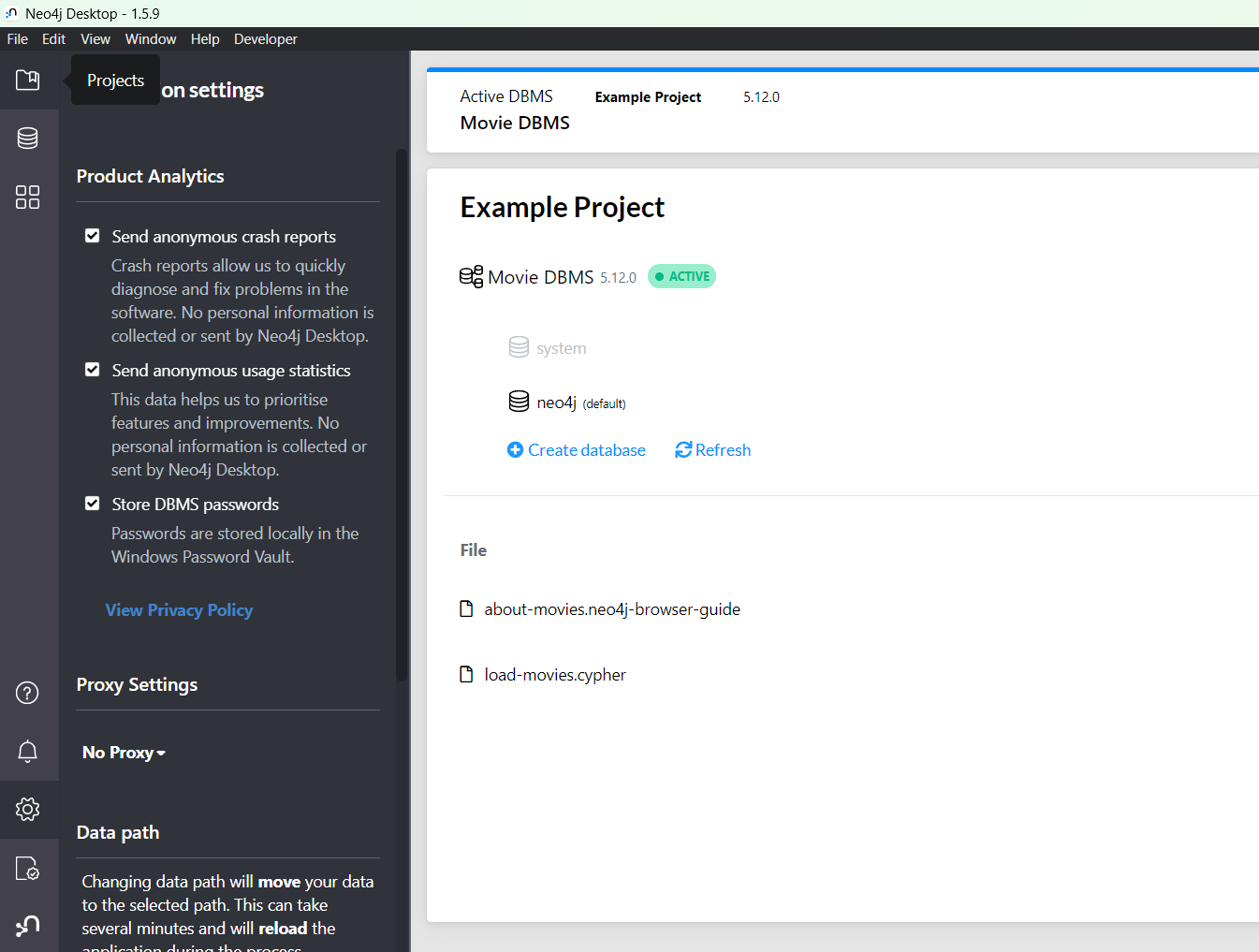
1. Paste the activation code got while downloading the application and click on activate to make it ready

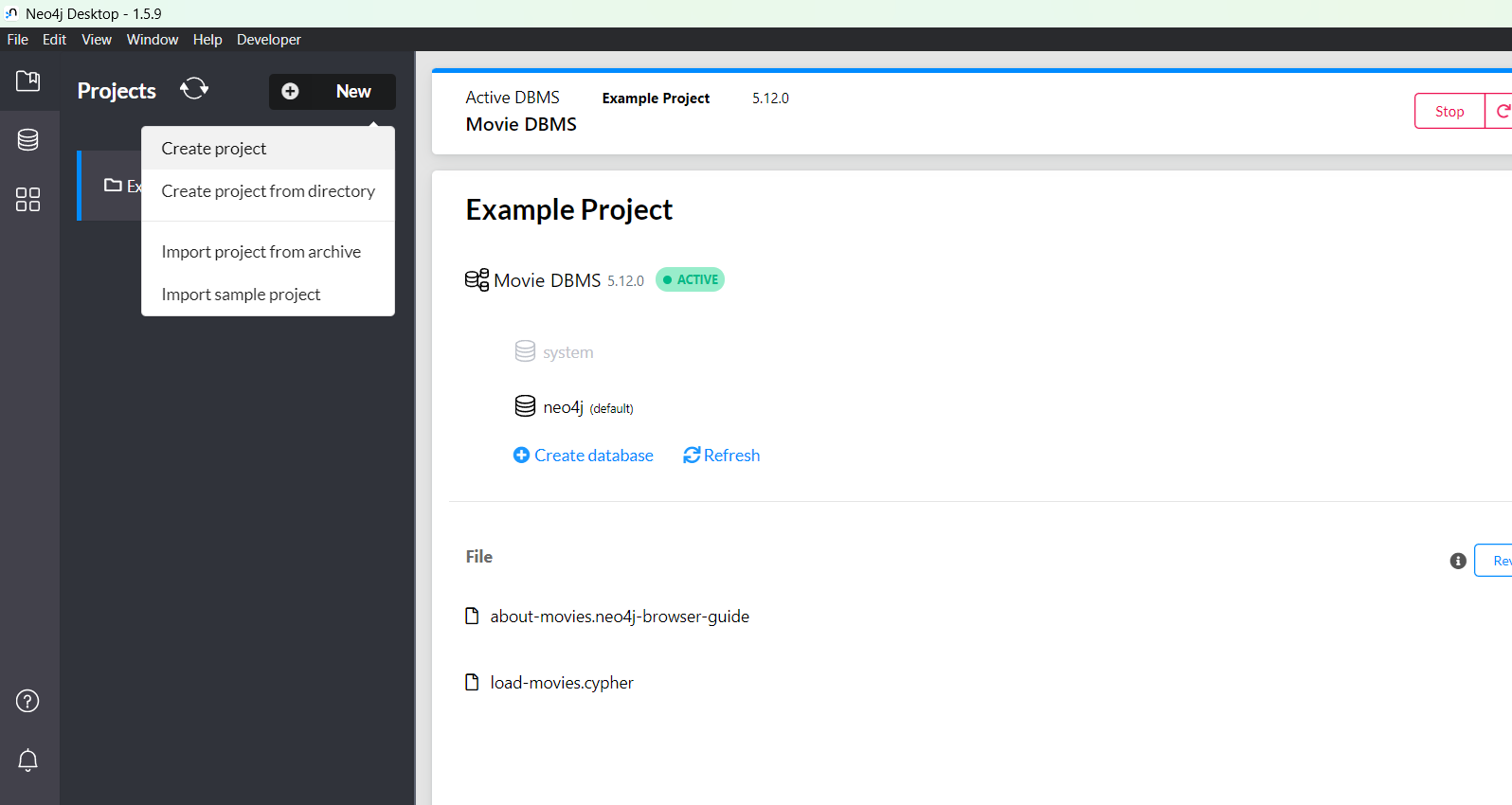
1. Click on ok and let all updates get installed allow for it



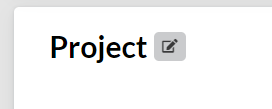
1. If want to diable tracking click on setting and disable send anonymous reports and stats



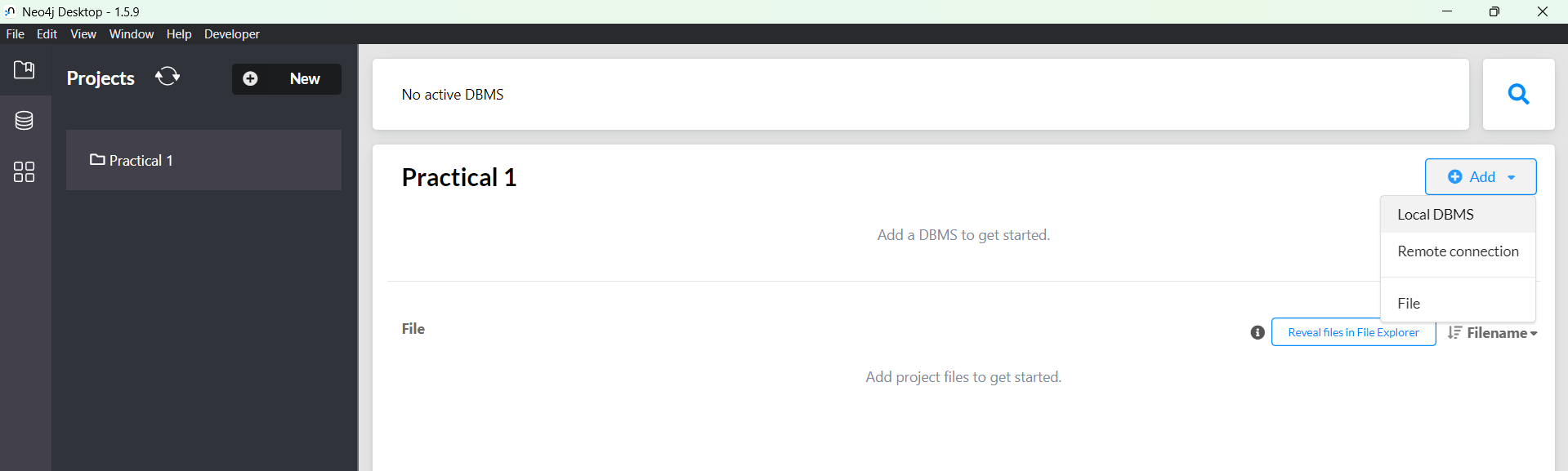
1. Click on the New panel and create a new project



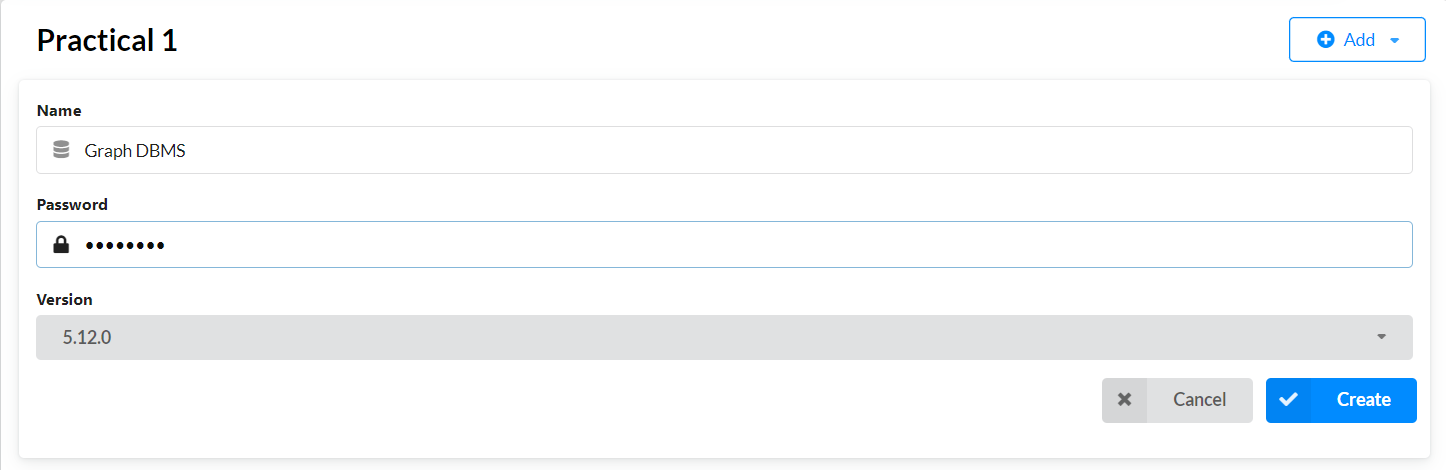
1. Rename the project to Practical 1



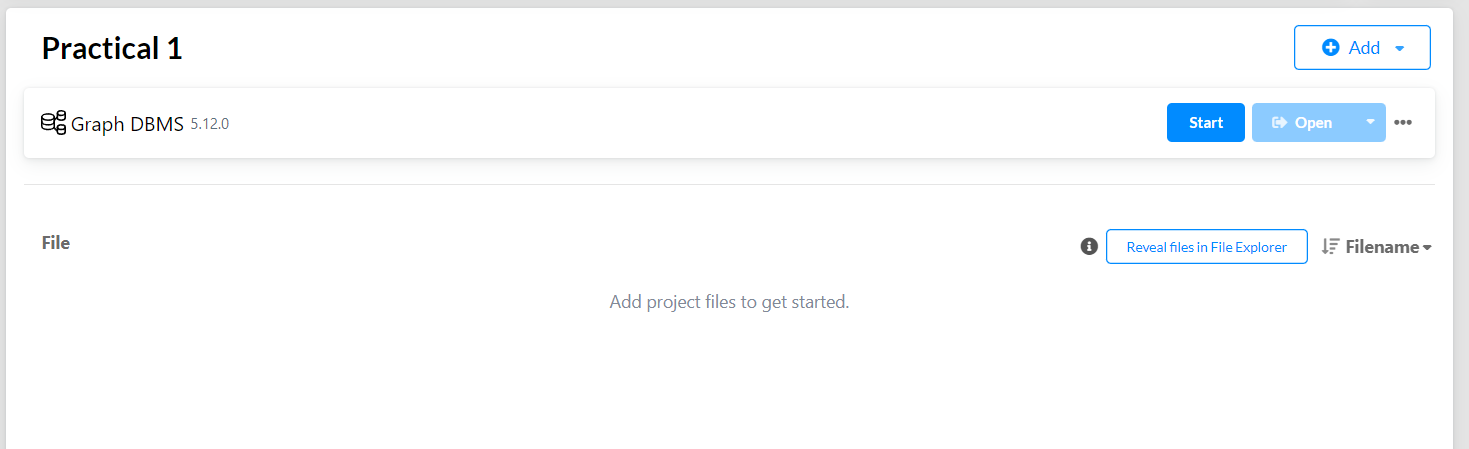
1. In the right side click on add panel and select local dbms



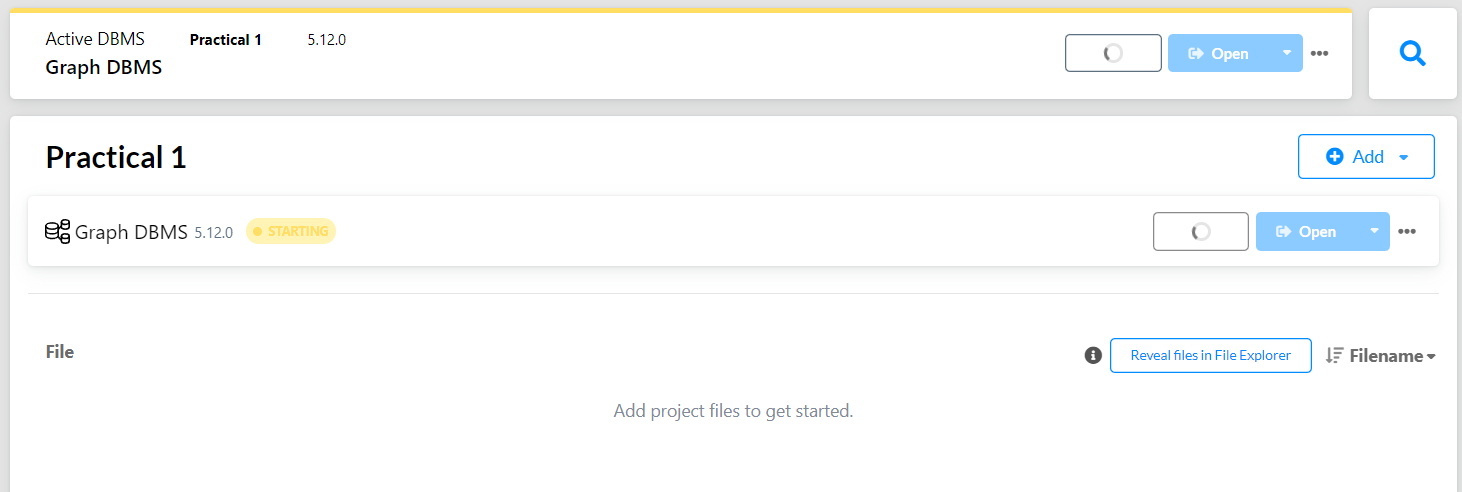
1. Set a password with eight characters long and click on create and wait till the dbms gets created



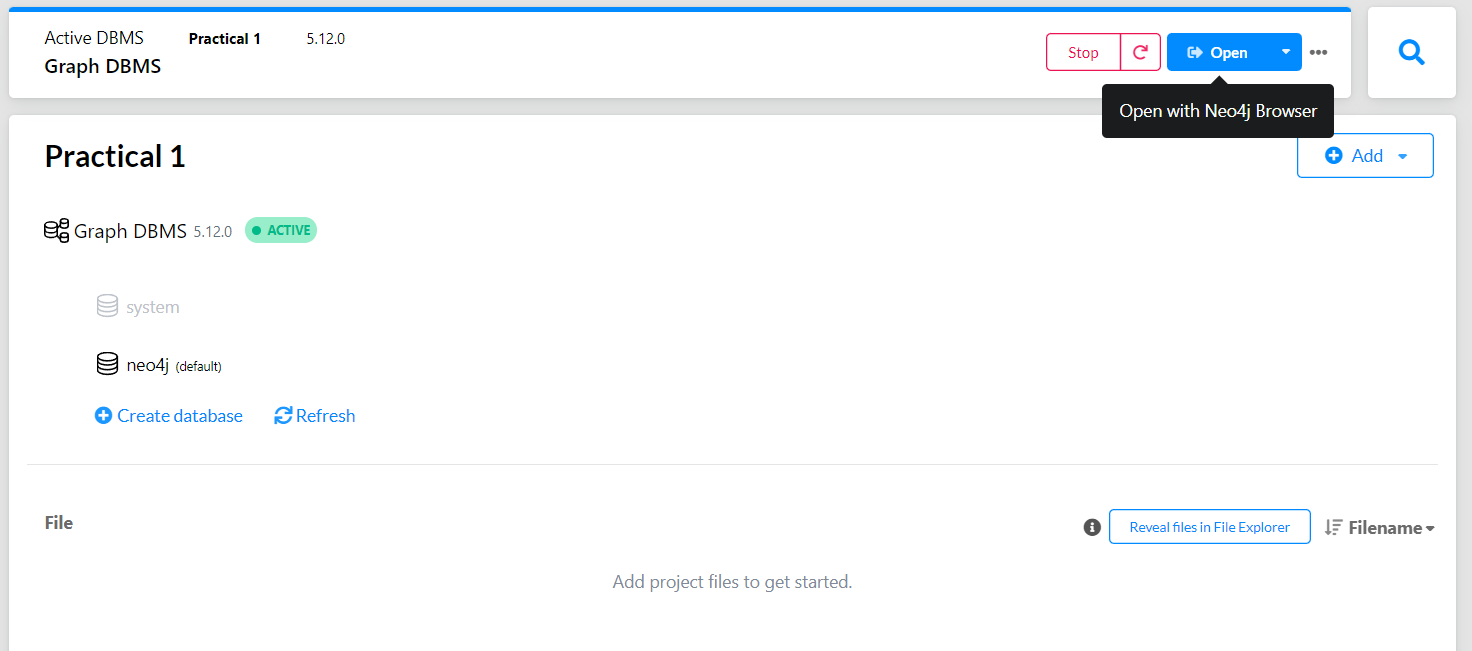
1. Now click on start button to strat the dbms



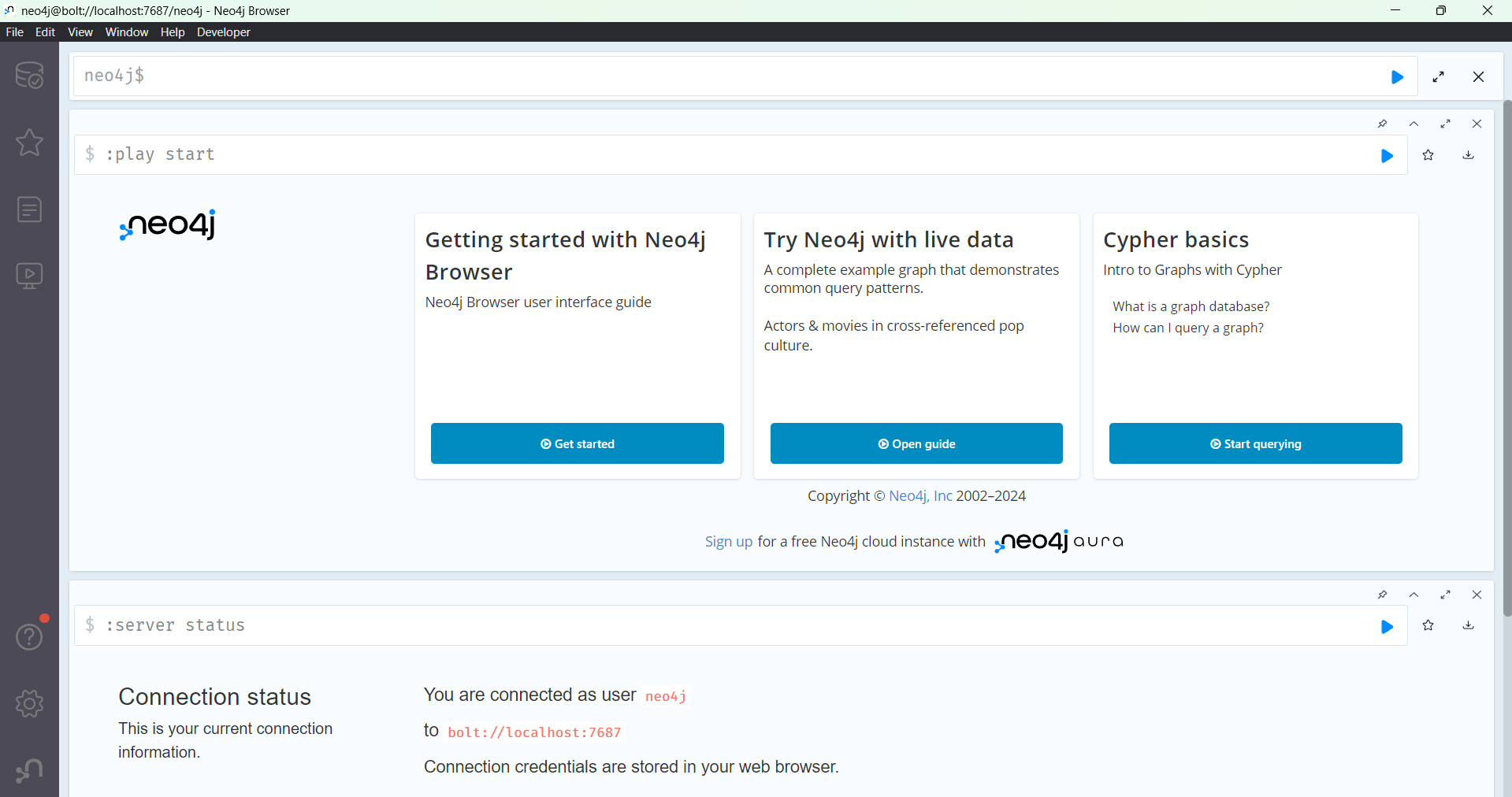
1. A terminal will open do not close it otherwise dbms will get stopped



1. If the dbms gets started Click on open to start the neo4j browser on the localhost

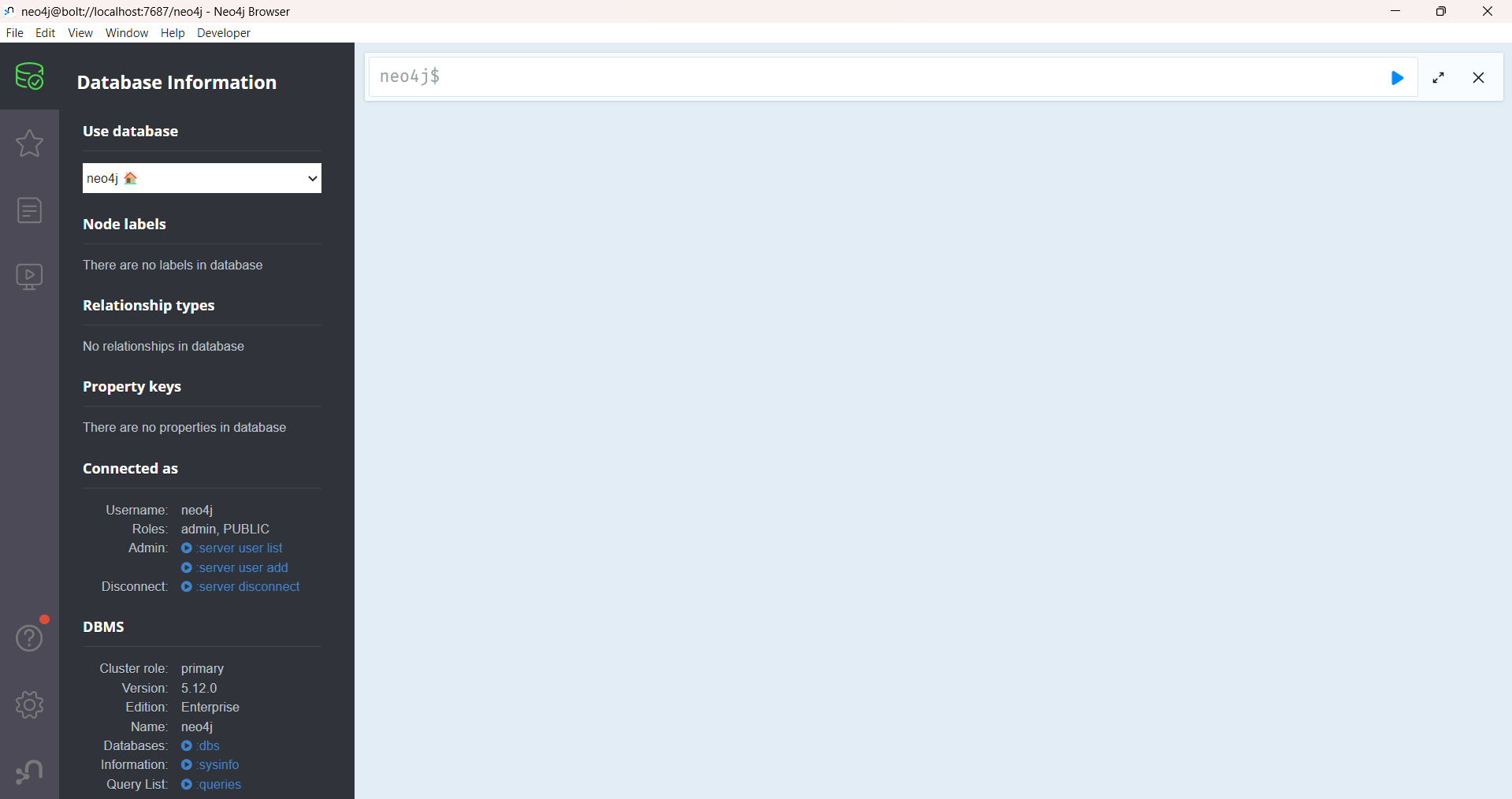


1. Neo4j Browser is opened if no issues detected

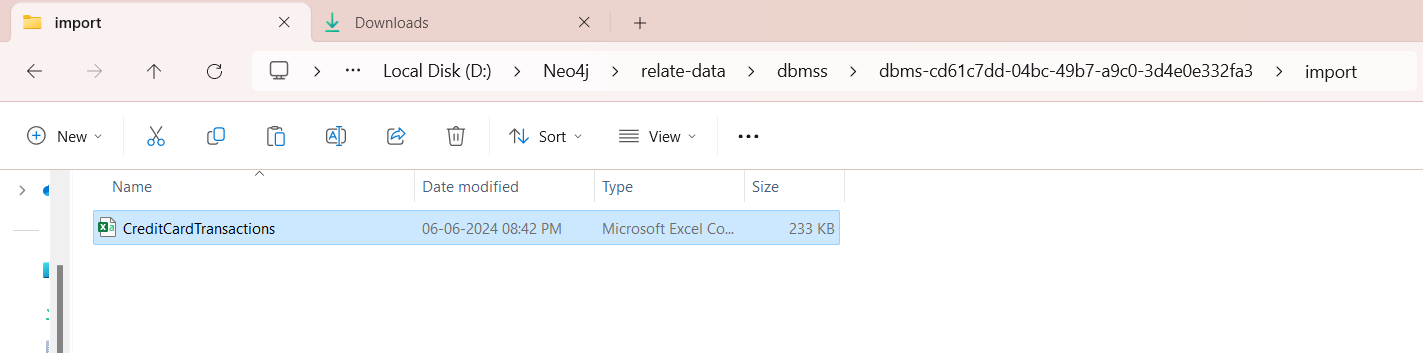


**[D] VISUALIZATION WITH NEO4J**

1. Clear all the cypher queries in the browser



1. Now copy and paste your processed CreditCardTransactions.csv file into your project import directory (/Neo4j/relate-data/dbmss/dbms-projectid /import/)



1. Get back to your neo4j browser and start your cypher query, first will import the dataset and assign the datatype of the columns
2. Type the cypher query and run it

// Load CSV data and create Transaction nodes

LOAD CSV WITH HEADERS FROM 'file:///CreditCardTransactions.csv' AS row

CREATE (t:Transaction {

    transaction\_dollar\_amount: toFloat(row.transaction\_dollar\_amount),

    credit\_card: row.credit\_card,

    city: row.city,

    state: row.state,

    zipcode: row.zipcode,

    credit\_card\_limit: toInteger(row.credit\_card\_limit),

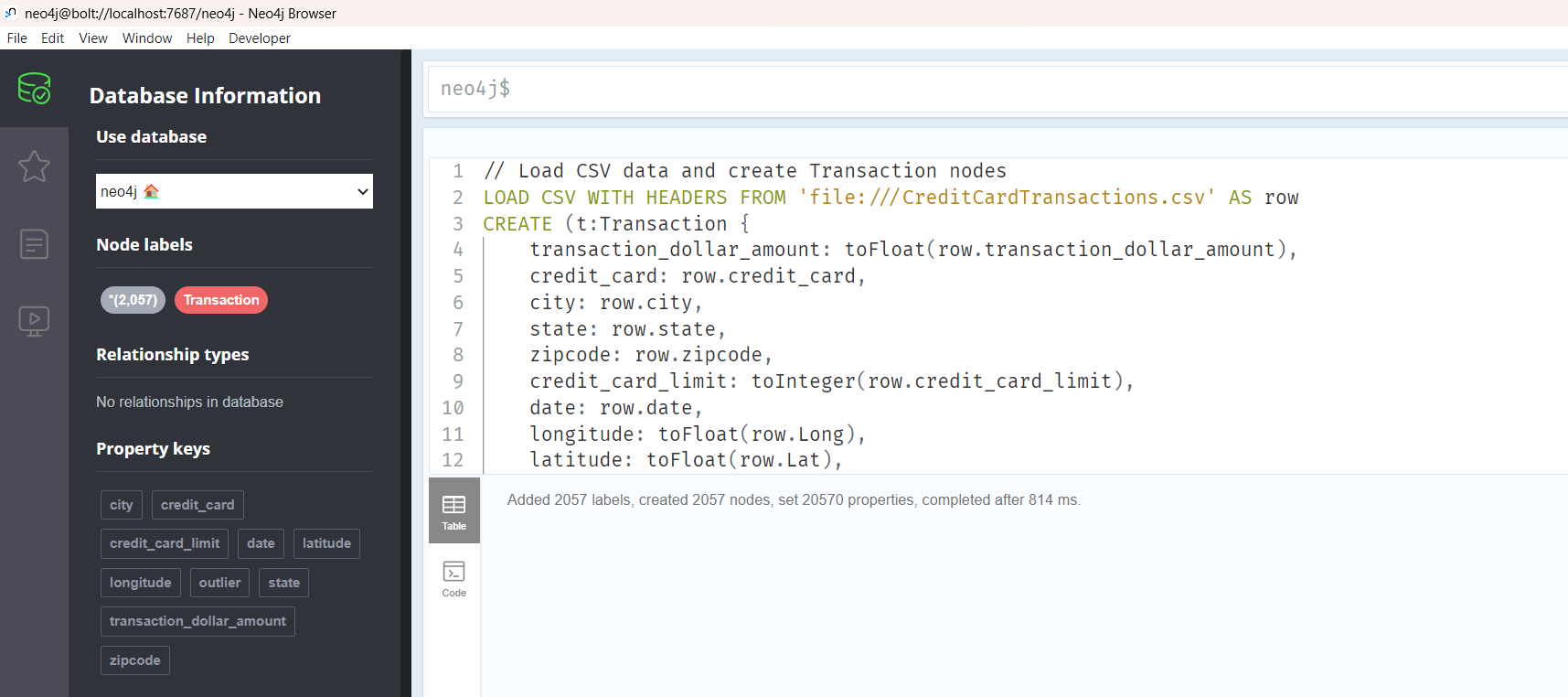
    date: row.date,

    longitude: toFloat(row.Long),

    latitude: toFloat(row.Lat),

    outlier: row.outlier

});



1. Next define the relationships between anomalous transactions and anomaly nodes

If it doesn’t shows any result check the csv file credit\_card number format and correct it

// relationships between anomalous transactions and Anomaly nodes

MATCH (t:Transaction)

WHERE t.outlier = 'TRUE'

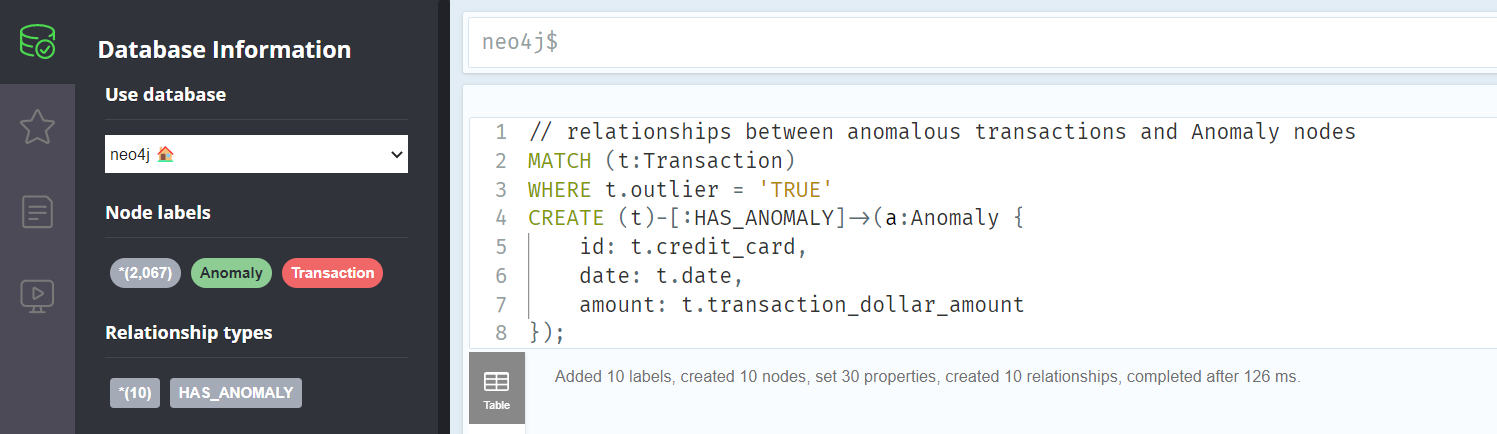
CREATE (t)-[:HAS\_ANOMALY]->(a:Anomaly {

    id: t.credit\_card,

    date: t.date,

    amount: t.transaction\_dollar\_amount

});

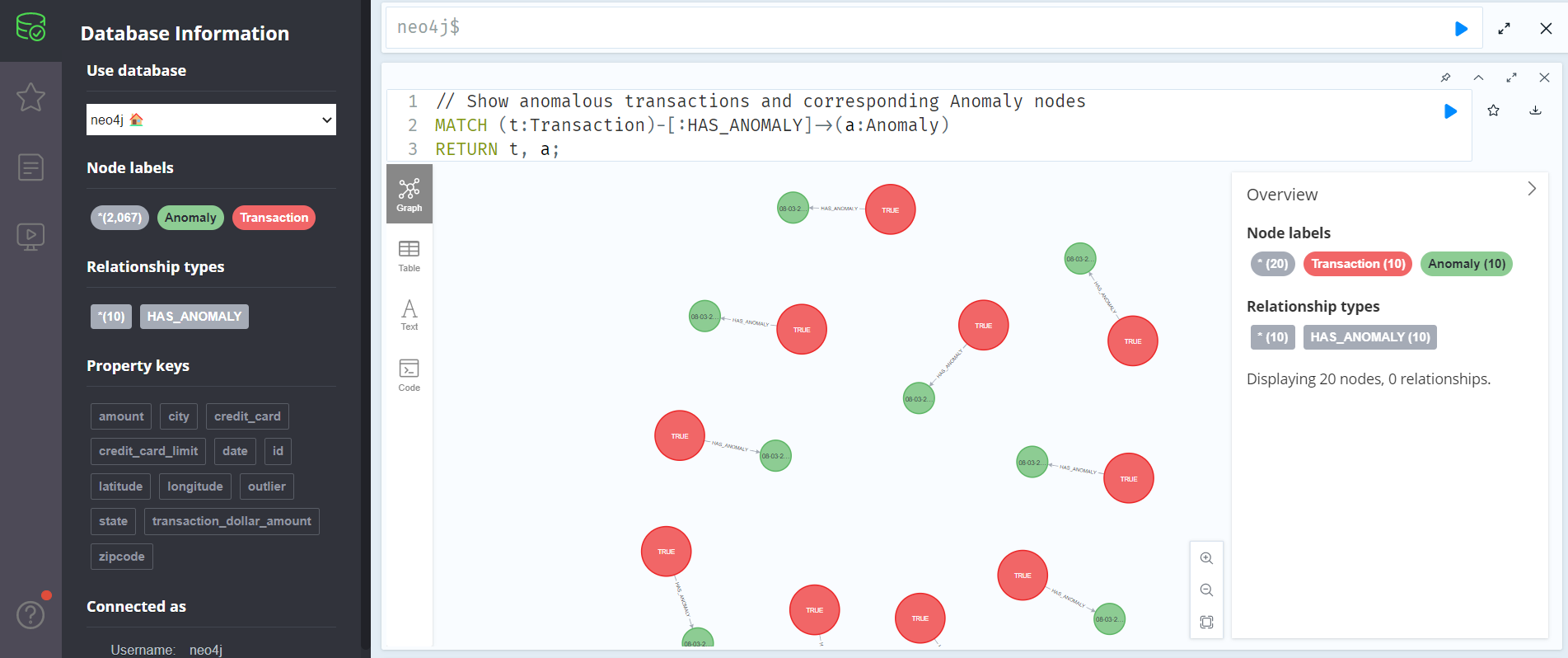


1. Visualize Transactions and its anomalies

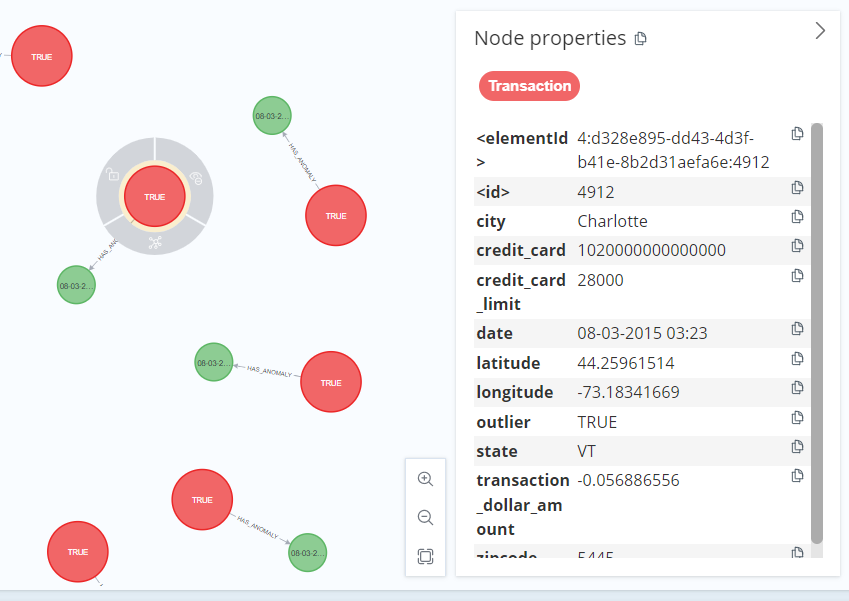
// Show anomalous transactions and corresponding Anomaly nodes

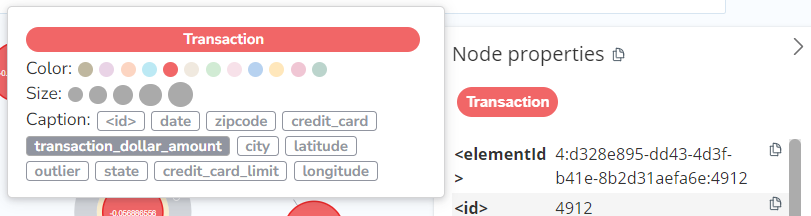
MATCH (t:Transaction)-[:HAS\_ANOMALY]->(a:Anomaly)

RETURN t, a;



1. You can see the details by clicking on nodes and also can modify according to you need by clicking on the label





1. Show anomalous transactions and corresponding Anomaly nodes for a specific city (Houston in this case)

We can Interactively explore anomalies by selecting its labels

