

Name: Bhagyashri Pravin Niphade

Class: BE - B

Roll No: 07

Experiment No. 1

Title:

For an organization of your choice, choose a set of business processes. Design star / snow flake schemas for analyzing these processes. Create a fact constellation schema by combining them. Extract data from different data sources, apply suitable transformations and load into destination tables using an ETL tool.

Objective:

Understand the basics of Star/Snowflake/Fact Constellation schema & learn the Rapidminer tool for performing various operations on in-built or external datasets.

Problem Statement:

Design a basic ETL model using Rapid Miner Application.

Outcomes:

1. Students will be able to demonstrate Installation of Rapidminer Tool
2. Students will be able to demonstrate different operators & datasets in Rapidminer
3. Students will be able to demonstrate different operations on available data in Rapidminer

Hardware Requirement:

Any CPU with Pentium Processor or similar, 256 MB RAM or more, 1 GB Hard Disk or more

Software Requirements:

32/64 bit Linux/Windows Operating System, latest Rapidminer Tool

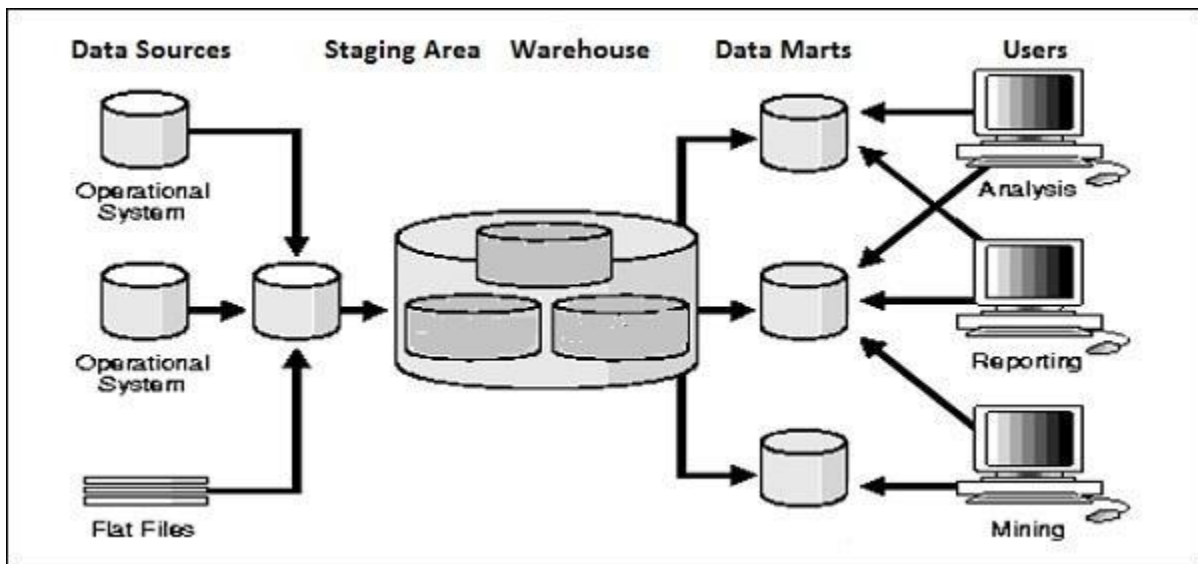
Theory:

What does ETL mean?

ETL stands for Extract, Transform and Load. An ETL tool extracts the data from different RDBMS source systems, transforms the data like applying calculations, concatenate, etc. and then load the data to Data Warehouse system. The data is loaded in the DW system in the form of dimension and fact tables.

Extraction

- A staging area is required during ETL load. There are various reasons why staging area is required.
- The source systems are only available for specific period of time to extract data. This period of time is less than the total data-load time. Therefore, staging area allows you to extract the data from the source system and keeps it in the staging area before the time slot ends.
- Staging area is required when you want to get the data from multiple data sources together or if you want to join two or more systems together. For example, you will not be able to perform a SQL query joining two tables from two physically different databases.
- Data extractions' time slot for different systems vary as per the time zone and operational hours.
- Data extracted from source systems can be used in multiple data warehouse system, Operation Data stores, etc.
- ETL allows you to perform complex transformations and requires extra area to store the data.



Transform

In data transformation, you apply a set of functions on extracted data to load it into the target system. Data, which does not require any transformation is known as direct move or pass through data.

You can apply different transformations on extracted data from the source system. For example, you can perform customized calculations. If you want sum-of-sales revenue and this is not in database, you can apply the **SUM** formula during transformation and load the data.

For example, if you have the first name and the last name in a table in different columns, you can use concatenate before loading.

Load

During Load phase, data is loaded into the end-target system and it can be a flat file or a Data Warehouse system.

Tool for ETL: *RAPID MINER*

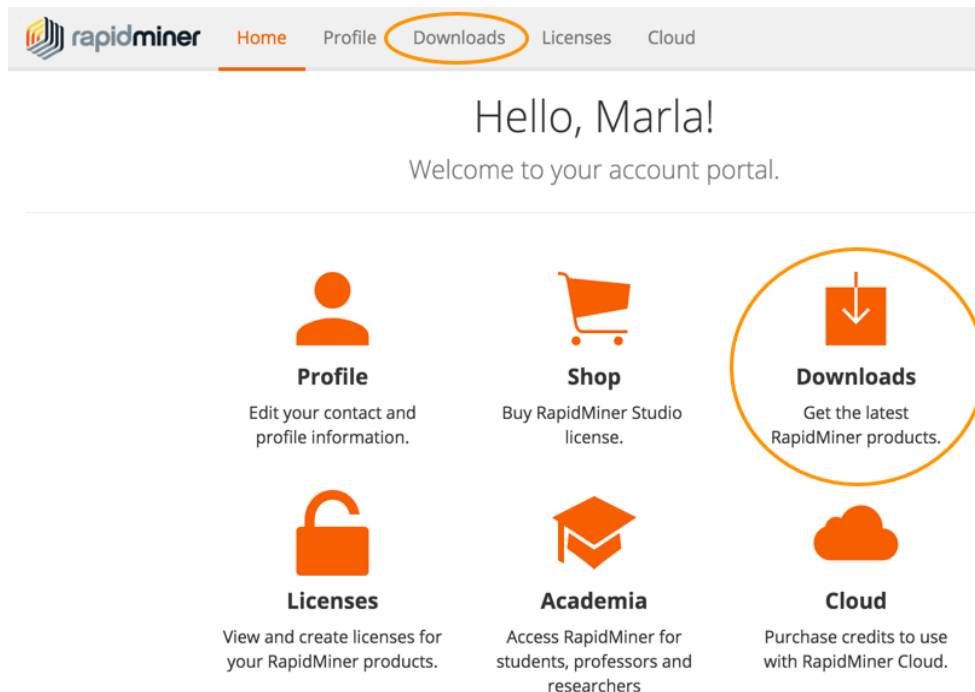
Rapid Miner is a world-leading open-source system for data mining. It is available as a stand-alone application for data analysis and as a data mining engine for the integration into own products. **Rapid Miner is now Rapid Miner Studio** and Rapid Analytics is now called Rapid Miner Server.

In a few words, Rapid Miner Studio is a "downloadable GUI for machine learning, data mining, text mining, predictive analytics and business analytics". It can also be used (for most purposes) in batch mode (command line mode).

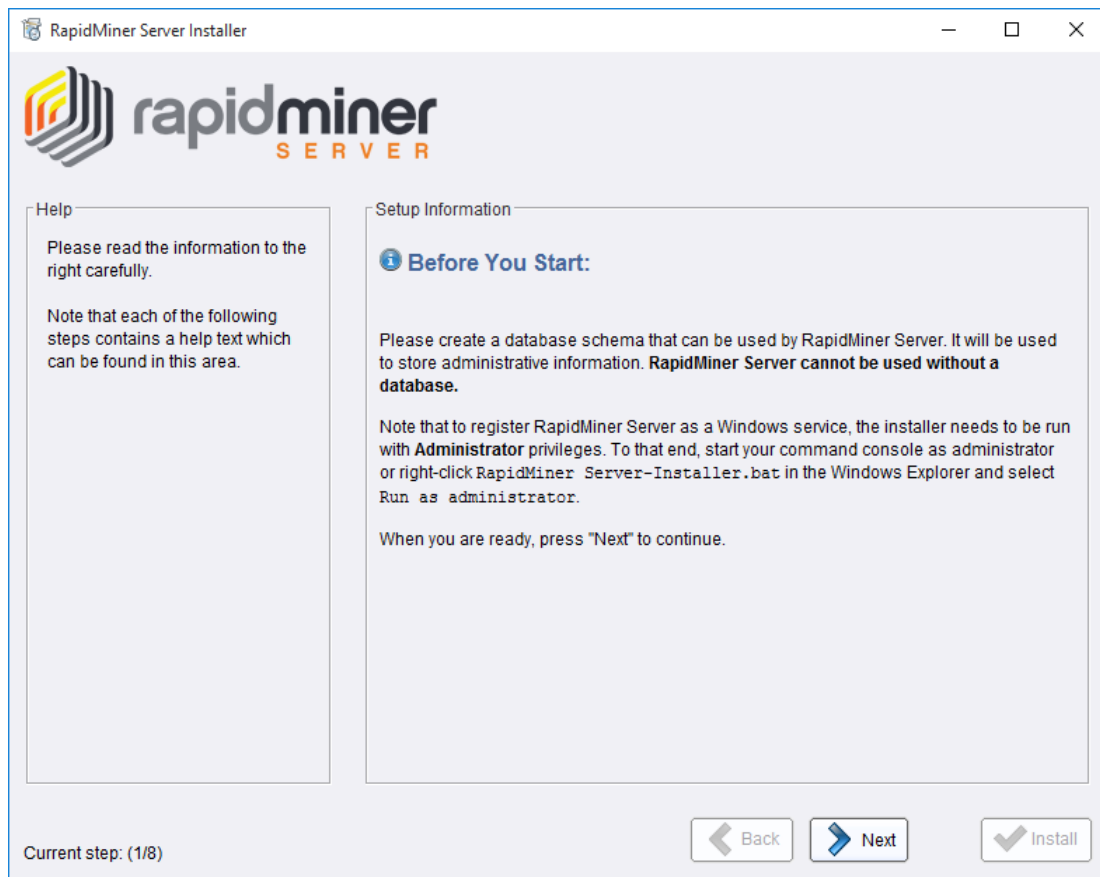
Rapid Miner Support to Nominal, Numerical values, Integers, Real numbers, 2-value nominal, multi-value nominal etc.

STEPS FOR INSTALLATION:

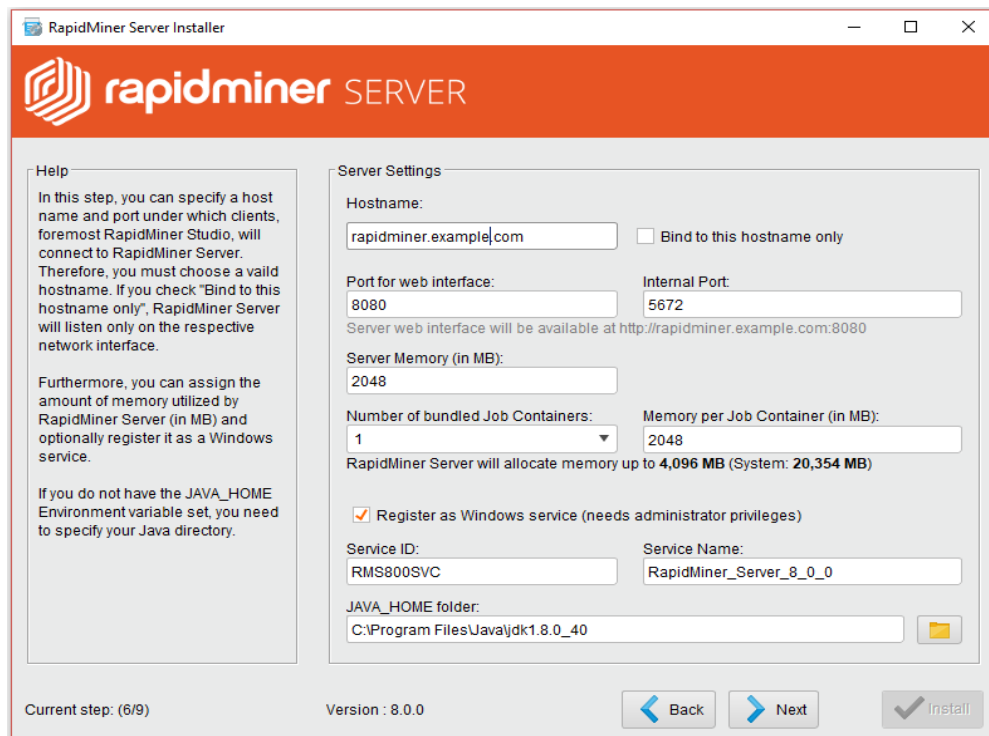
1. Downloading Rapid Miner Server



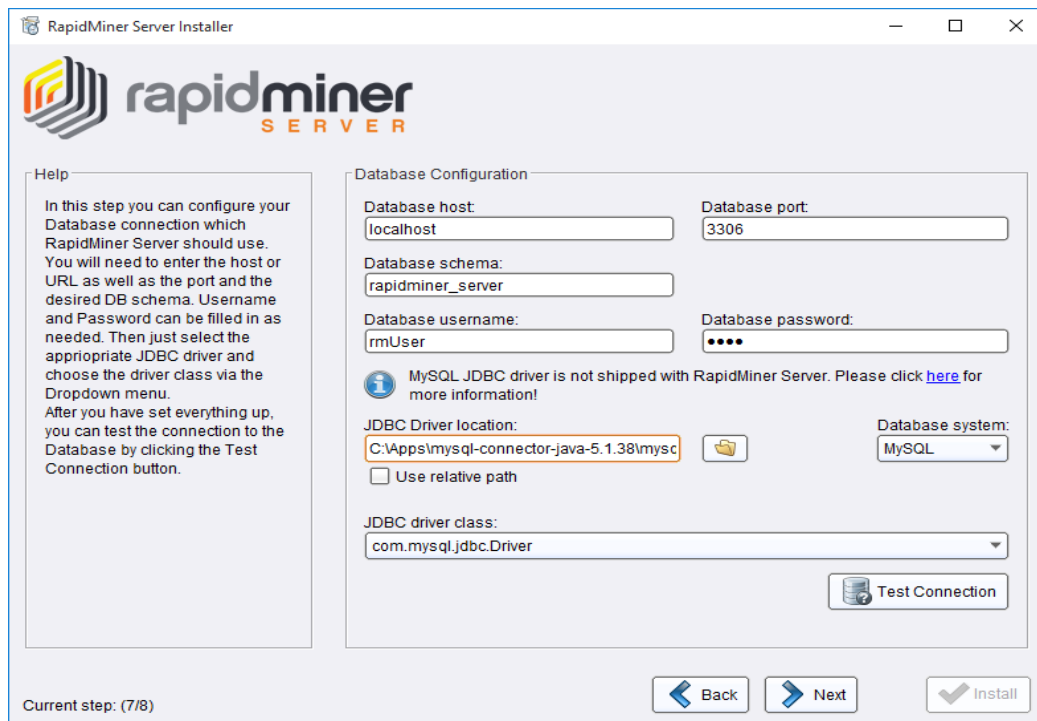
2. Installing Rapid Miner Server



3. Configuring Rapid Miner Server settings



4. Configuring Rapid Miner Server's database connection



The screenshot shows the 'Database Configuration' step of the RapidMiner Server Installer. The window title is 'RapidMiner Server Installer'. The RapidMiner logo is at the top left. On the left, a 'Help' box explains the step. The main area contains fields for 'Database host' (localhost), 'Database port' (3306), 'Database schema' (rapidminer_server), 'Database username' (rmUser), and 'Database password' (masked with dots). A note indicates the MySQL JDBC driver is not shipped and provides a link. Below, 'JDBC Driver location' is set to 'C:\Apps\mysql-connector-java-5.1.38\mysql', with a checkbox for 'Use relative path'. The 'Database system' is set to 'MySQL' in a dropdown. The 'JDBC driver class' is 'com.mysql.jdbc.Driver'. A 'Test Connection' button is at the bottom right. Navigation buttons 'Back', 'Next', and 'Install' are at the bottom. The status bar shows 'Current step: (7/8)'.

Help

In this step you can configure your Database connection which RapidMiner Server should use. You will need to enter the host or URL as well as the port and the desired DB schema. Username and Password can be filled in as needed. Then just select the appropriate JDBC driver and choose the driver class via the Dropdown menu. After you have set everything up, you can test the connection to the Database by clicking the Test Connection button.

Database Configuration

Database host: localhost Database port: 3306

Database schema: rapidminer_server

Database username: rmUser Database password:

MySQL JDBC driver is not shipped with RapidMiner Server. Please click [here](#) for more information!

JDBC Driver location: C:\Apps\mysql-connector-java-5.1.38\mysql Use relative path

Database system: MySQL

JDBC driver class: com.mysql.jdbc.Driver

Test Connection

Current step: (7/8) Back Next Install

5. Installing Radoop Proxy



The screenshot shows the 'Radoop Proxy installation' step of the RapidMiner Server Installer. The window title is 'RapidMiner Server Installer'. The RapidMiner logo is at the top left. On the left, a 'Help' box explains the step. The main area has a checkbox for 'Enable Radoop Proxy' and a 'Port' field set to '1081'. Navigation buttons 'Back', 'Next', and 'Install' are at the bottom. The status bar shows 'Current step: (9/9)'.

Help

In this step you can enable the Radoop Proxy component that provides a proxy server for RapidMiner Studio users with Radoop extension installed. If you want to know more about Radoop Proxy and its use cases, please visit the <http://docs.rapidminer.com> website.

Radoop Proxy installation

☐ Enable Radoop Proxy

Port: 1081

Current step: (9/9) Back Next Install

6. Completing the installation

Once logged in, complete the final installation steps.

1. From the **SQL Dialect** pull-down, verify that the database type displayed is the one you used to create the Rapid Miner Server database.
2. Verify the setting for the integrated Quartz scheduler, which is enabled by default.

3. Specify the path to the plug in directory. You can install additional RapidMiner extensions by placing them in, or saving them to, this directory. Note that all extensions bundled with RapidMiner Studio are also bundled with Rapid Miner Server (no installation is necessary). These bundled extensions are stored in a separate directory that is independent of the path specified here. Be sure that you have write permission to the directory.
4. Specify the path to upload directory. This is the directory where RapidMiner Server stores temporary files needed for processes. The installation process creates a local uploads directory in the installation folder. However, if you install Rapid Miner Server on a relatively small hard disk and, for example, use many file objects in processes or if you have large resulting files, consider creating a directory elsewhere in the cluster to store the temporary files. Be sure that you have write permission to the directory.
5. Click **Start installation now**.
6. Installation gets completed.

Data Warehousing Schemas

1. Star Schema
2. Snowflake Schema
3. Fact Constellation

Star Schema

For example, as you can see in the above-given image that fact table is at the center which contains keys to every dimension table like Deal_ID, Model ID, Date_ID, Product_ID, Branch_ID & other attributes like Units sold and revenue.

Characteristics of Star Schema:

- Every dimension in a star schema is represented with the only one-dimension table.
- The dimension table should contain the set of attributes.
- The dimension table is joined to the fact table using a foreign key
- The dimension table are not joined to each other
- Fact table would contain key and measure
- The Star schema is easy to understand and provides optimal disk usage.
- The dimension tables are not normalized. For instance, in the above figure, Country_ID does not have Country lookup table as an OLTP design would have.
- The schema is widely supported by BI Tools

Snowflake Schema

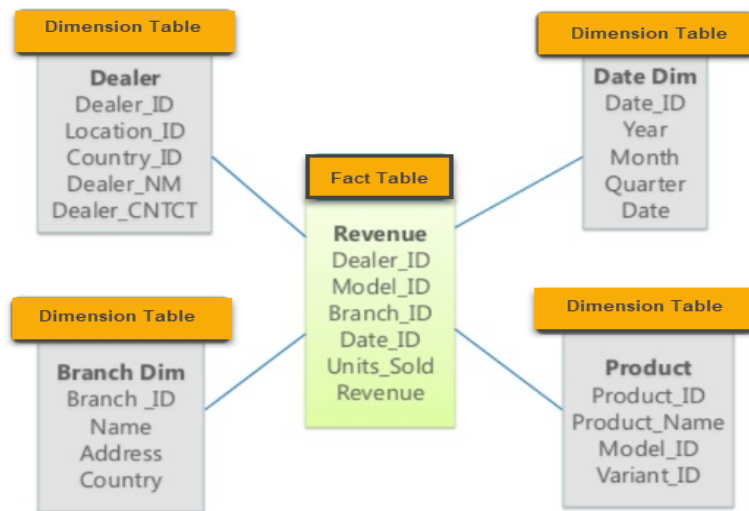
A Snowflake Schema is an extension of a Star Schema, and it adds additional dimensions. It is called snowflake because its diagram resembles a Snowflake.

The dimension tables are normalized which splits data into additional tables. In the following example, Country is further normalized into an individual table.

Characteristics of Snowflake Schema:

- The main benefit of the snowflake schema it uses smaller disk space.
- Easier to implement a dimension is added to the Schema
- Due to multiple tables query performance is reduced
- The primary challenge that you will face while using the snowflake Schema is that you need to perform more maintenance efforts because of the more lookup tables.

Star Schema	Snow Flake Schema
Hierarchies for the dimensions are stored in the dimensional table.	Hierarchies are divided into separate tables.
It contains a fact table surrounded by dimension tables.	One fact table surrounded by dimension table which are in turn surrounded by dimension table
In a star schema, only single join creates the relationship between the fact table and any dimension tables.	A snowflake schema requires many joins to fetch the data.
Simple DB Design.	Very Complex DB Design.
Denormalized Data structure and query also run faster.	Normalized Data Structure.
High level of Data redundancy	Very low-level data redundancy
Single Dimension table contains aggregated data.	Data Split into different Dimension Tables.
Cube processing is faster.	Cube processing might be slow because of the complex join
Offers higher performing queries using Star Join Query Optimization. Tables may be connected with multiple dimensions.	The Snow Flake Schema is represented by centralized fact table which unlikely connected with multiple dimensions.

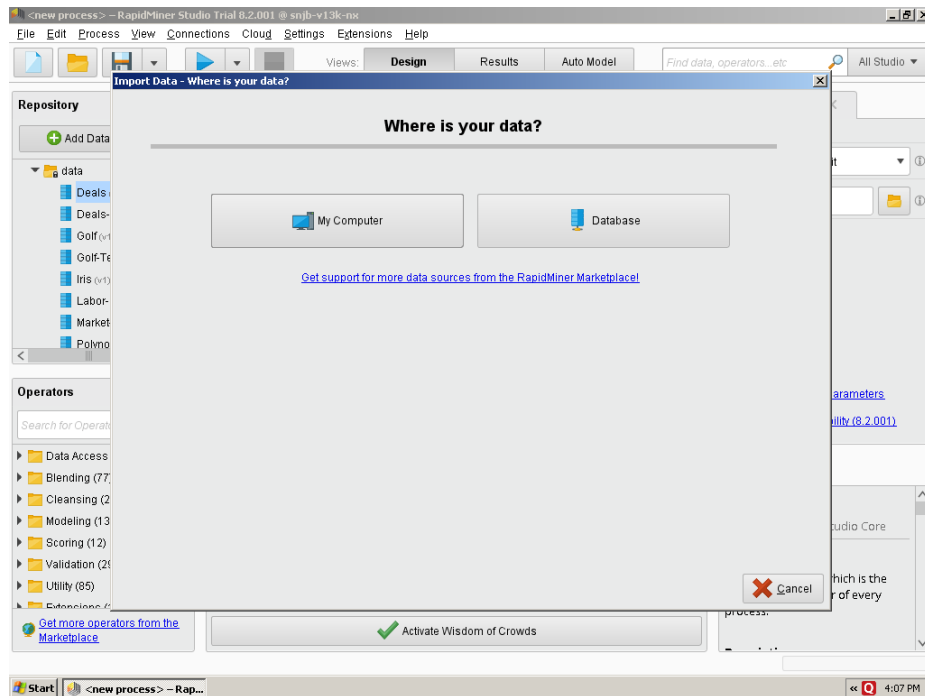


Star Schema

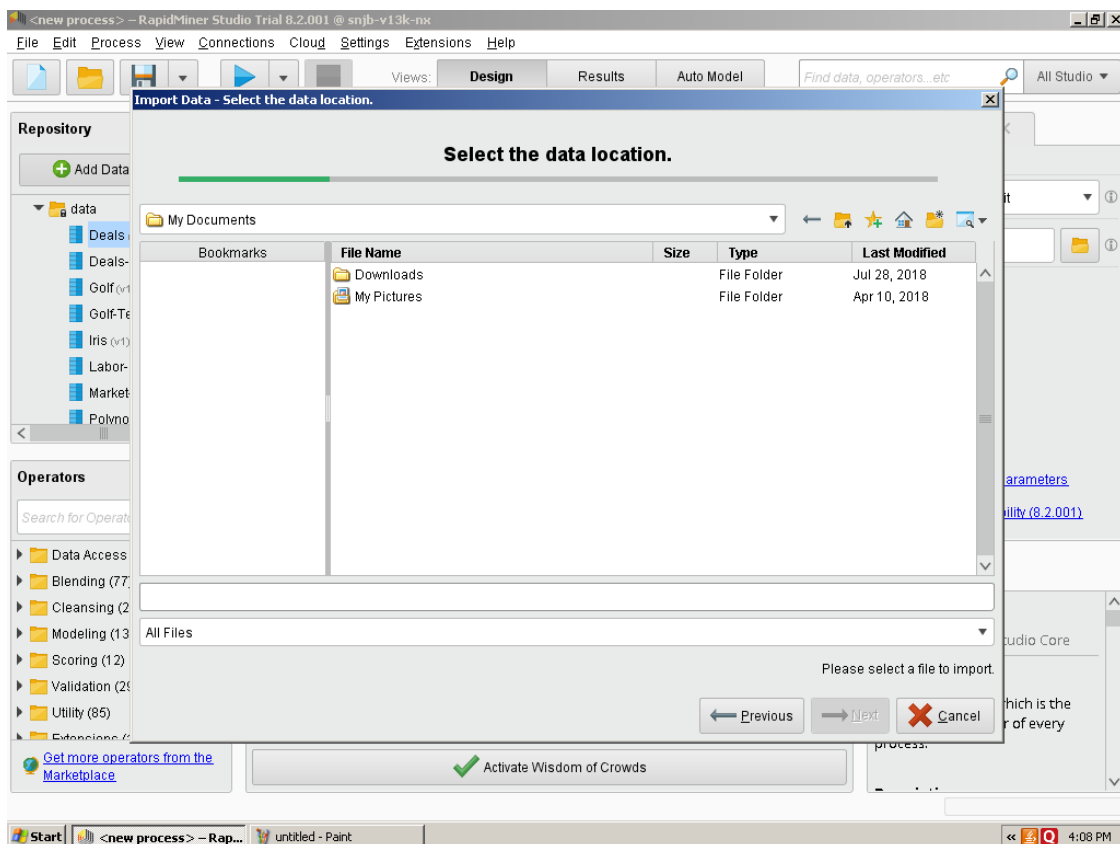
1. Design Model



Step-1 Import Data from Source



Step-2 Select Data Location



Step-3 Open Sample Data Set eg. Iris dataset available inbuilt with tool

The screenshot shows the RapidMiner Studio interface with the 'Results' tab selected. The 'ExampleSet (/Samples/data/Iris)' is displayed, showing 150 examples with 2 special attributes and 4 regular attributes. The data is filtered to all 150 examples. The table below represents the data shown in the interface:

Row No.	id	label	a1	a2	a3	a4
1	id_1	Iris-setosa	5.100	3.500	1.400	0.200
2	id_2	Iris-setosa	4.900	3	1.400	0.200
3	id_3	Iris-setosa	4.700	3.200	1.300	0.200
4	id_4	Iris-setosa	4.600	3.100	1.500	0.200
5	id_5	Iris-setosa	5	3.600	1.400	0.200
6	id_6	Iris-setosa	5.400	3.900	1.700	0.400
7	id_7	Iris-setosa	4.600	3.400	1.400	0.300
8	id_8	Iris-setosa	5	3.400	1.500	0.200
9	id_9	Iris-setosa	4.400	2.900	1.400	0.200
10	id_10	Iris-setosa	4.900	3.100	1.500	0.100
11	id_11	Iris-setosa	5.400	3.700	1.500	0.200
12	id_12	Iris-setosa	4.800	3.400	1.600	0.200
13	id_13	Iris-setosa	4.800	3	1.400	0.100
14	id_14	Iris-setosa	4.300	3	1.100	0.100
15	id_15	Iris-setosa	5.800	4	1.200	0.200
16	id_16	Iris-setosa	5.700	4.400	1.500	0.400
17	id_17	Iris-setosa	5.400	3.900	1.300	0.400

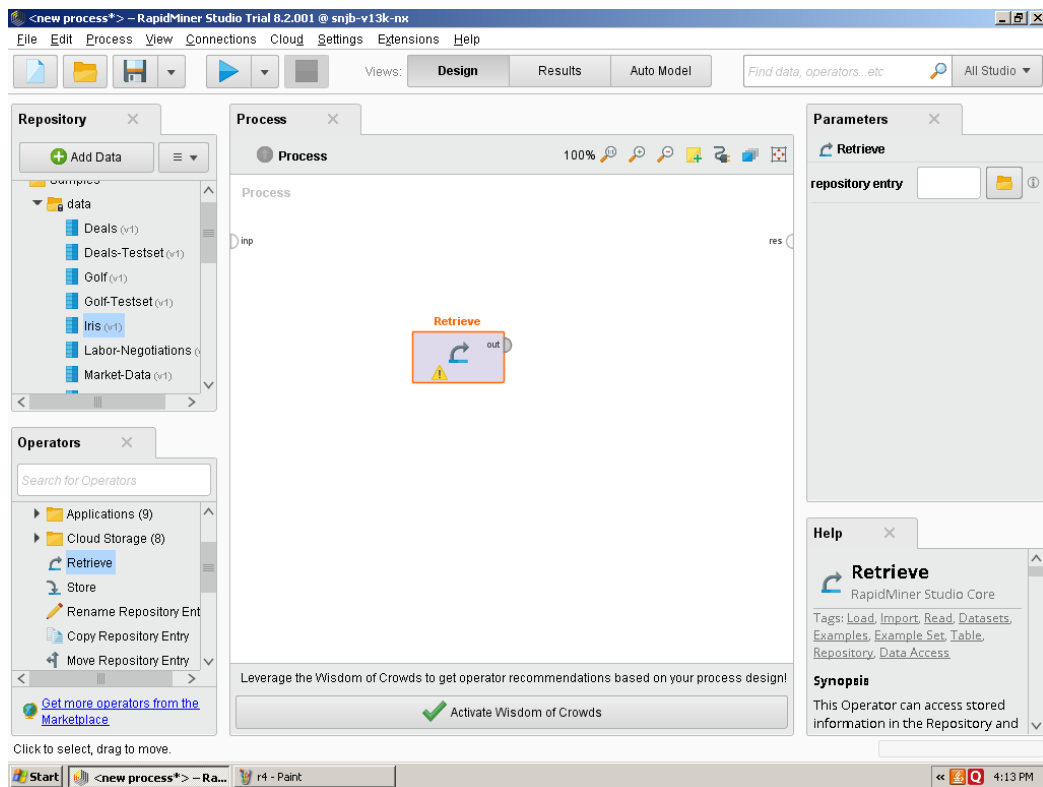
Step-4 Click on retrieve Operator Drag in Process View

The screenshot shows the RapidMiner Studio interface with the 'Design' tab selected. The 'Process' view is empty, and the 'Operators' panel is open. The 'Retrieve' operator is highlighted in the 'Operators' panel. A tooltip for the 'Retrieve' operator is displayed, showing its description and a keyboard shortcut.

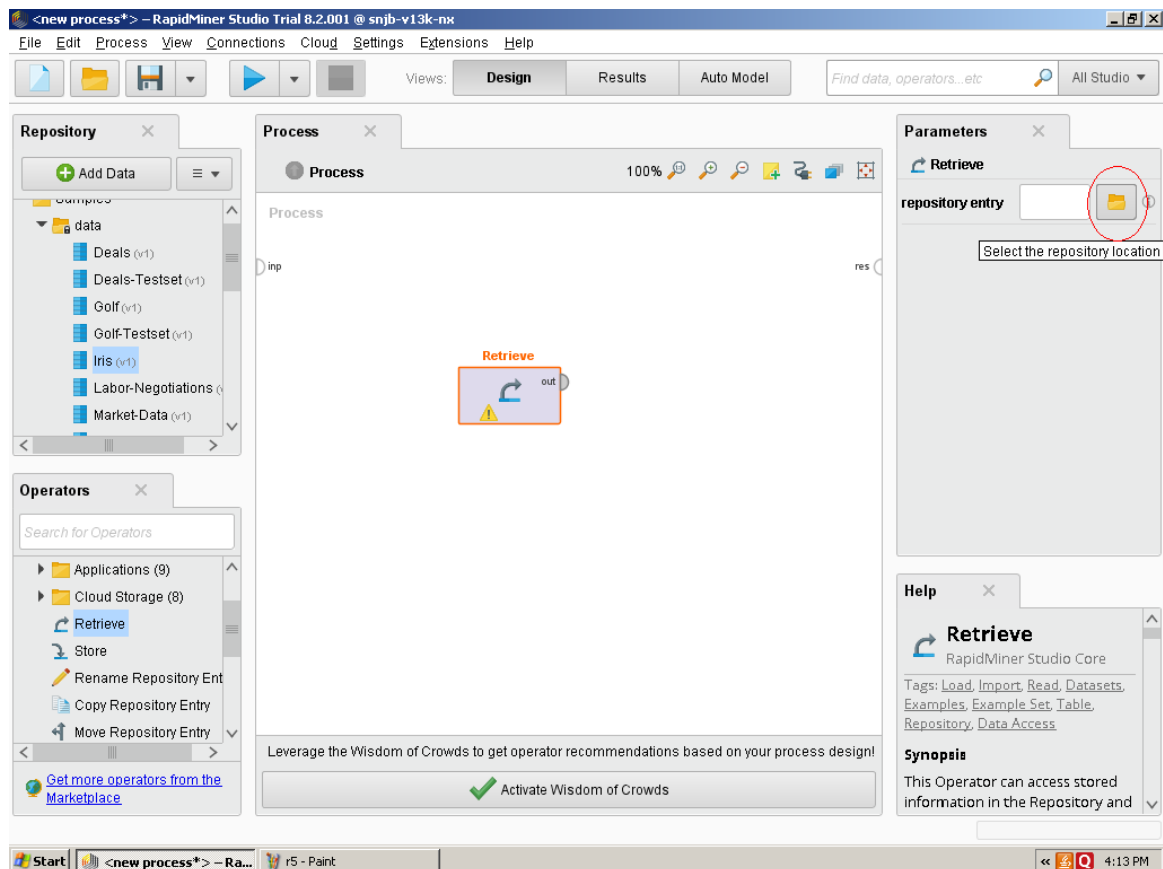
Retrieve
Reads an object from the data repository.
Press "F3" for focus.

The 'Repository' panel on the left shows the 'Iris (v1)' dataset selected. The 'Parameters' panel on the right shows the 'Process' operator parameters, including 'logverbosity' set to 'init' and 'logfile' set to an empty field. The 'Help' panel on the right shows the 'Process' operator synopsis.

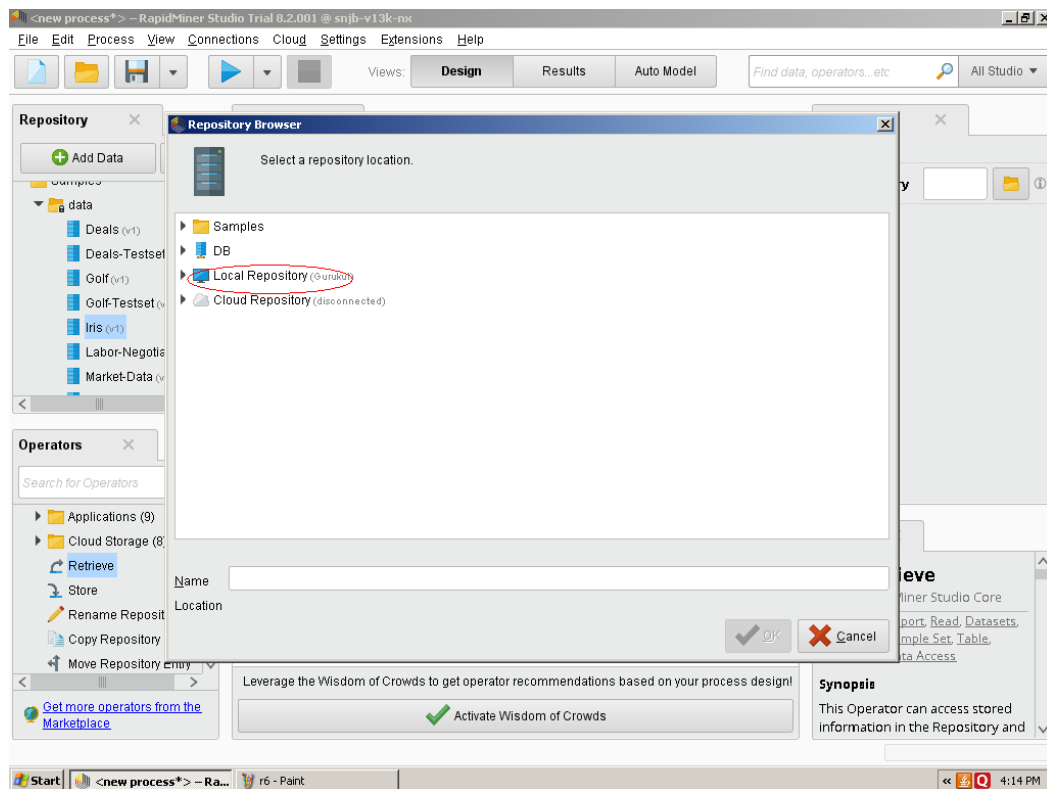
Step-5 Retrieve icon shows in Process View it has input and out Operator



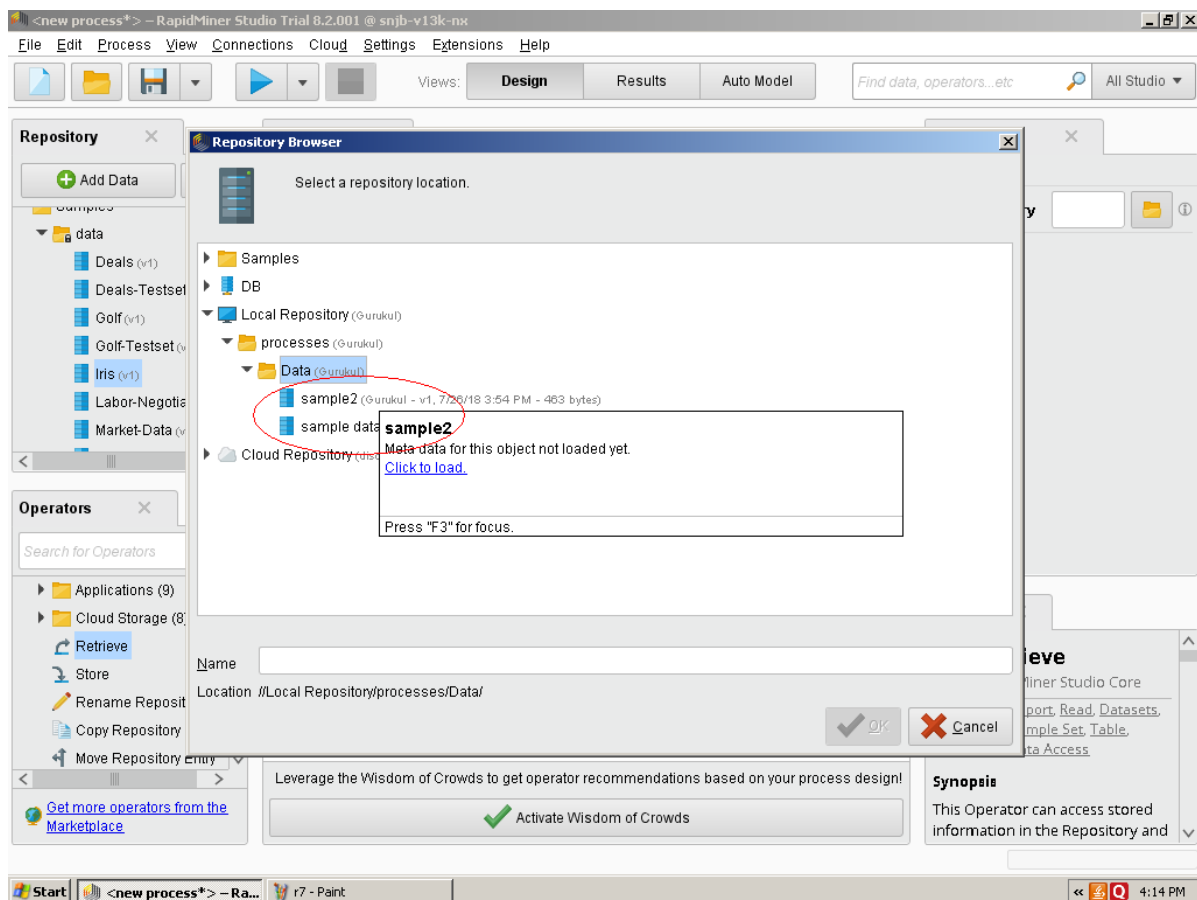
Step-6 Click on repository entry



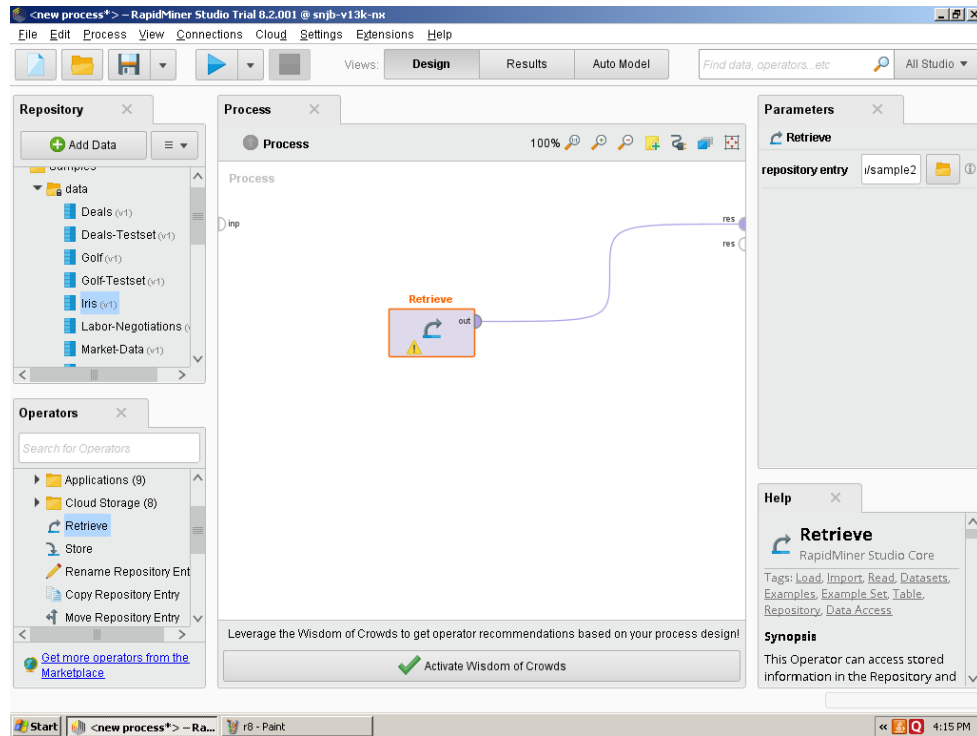
Step-7 Select Local Repository



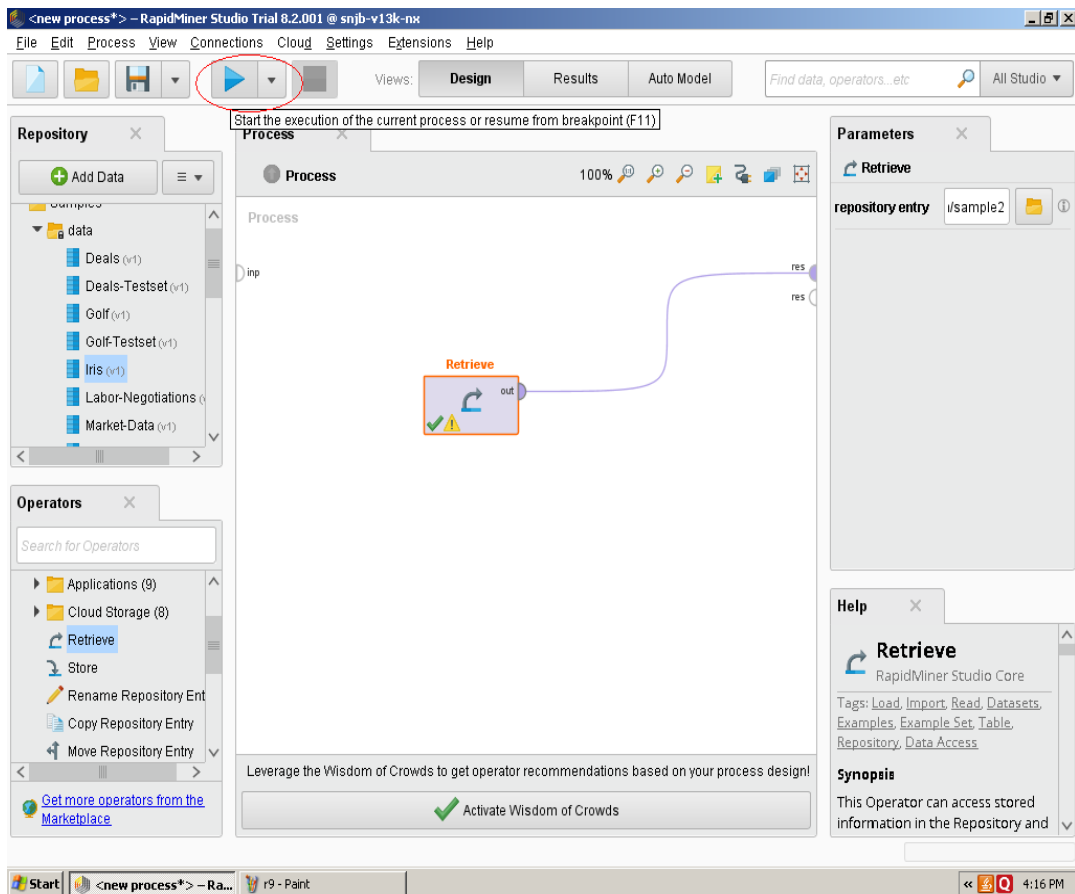
Step-8 Select Sample file



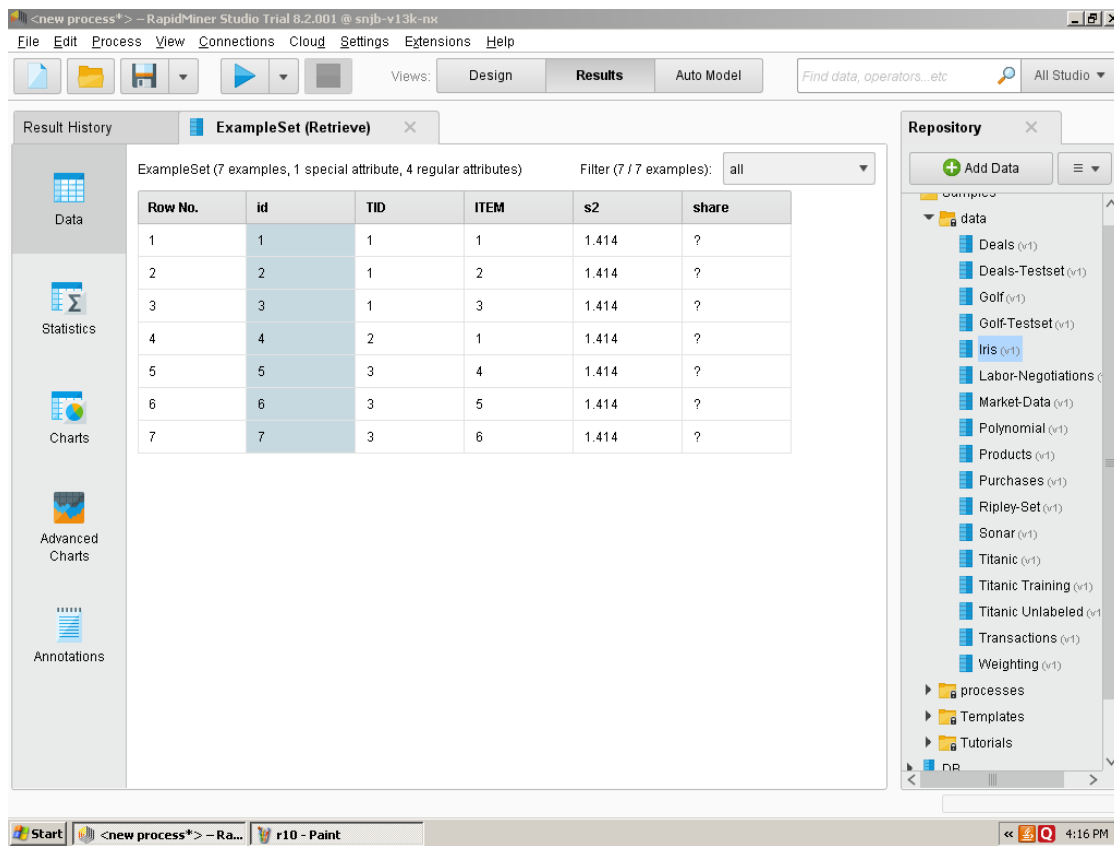
Step-9 Join Out Operator to result Operator



Step-10 Start Execution of Current Process



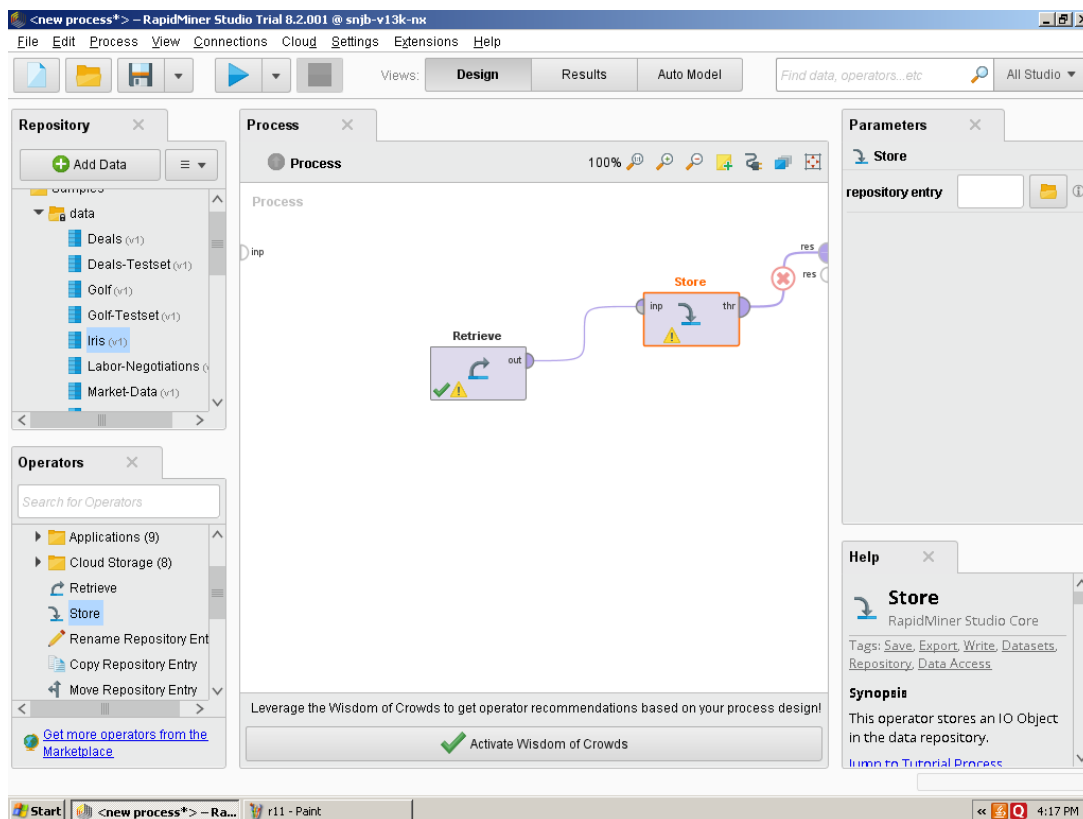
Step-11 Output Result Generated after Execution of Current Process



ExampleSet (7 examples, 1 special attribute, 4 regular attributes) Filter (7 / 7 examples): all

Row No.	id	TID	ITEM	s2	share
1	1	1	1	1.414	?
2	2	1	2	1.414	?
3	3	1	3	1.414	?
4	4	2	1	1.414	?
5	5	3	4	1.414	?
6	6	3	5	1.414	?
7	7	3	6	1.414	?

Step-12 Now you can add Store Operator and connect to result operator



Process

100%

Process

inp

res

res

thr

Store

Retrieve

out

Parameters

Store

repository entry

Help

Store

RapidMiner Studio Core

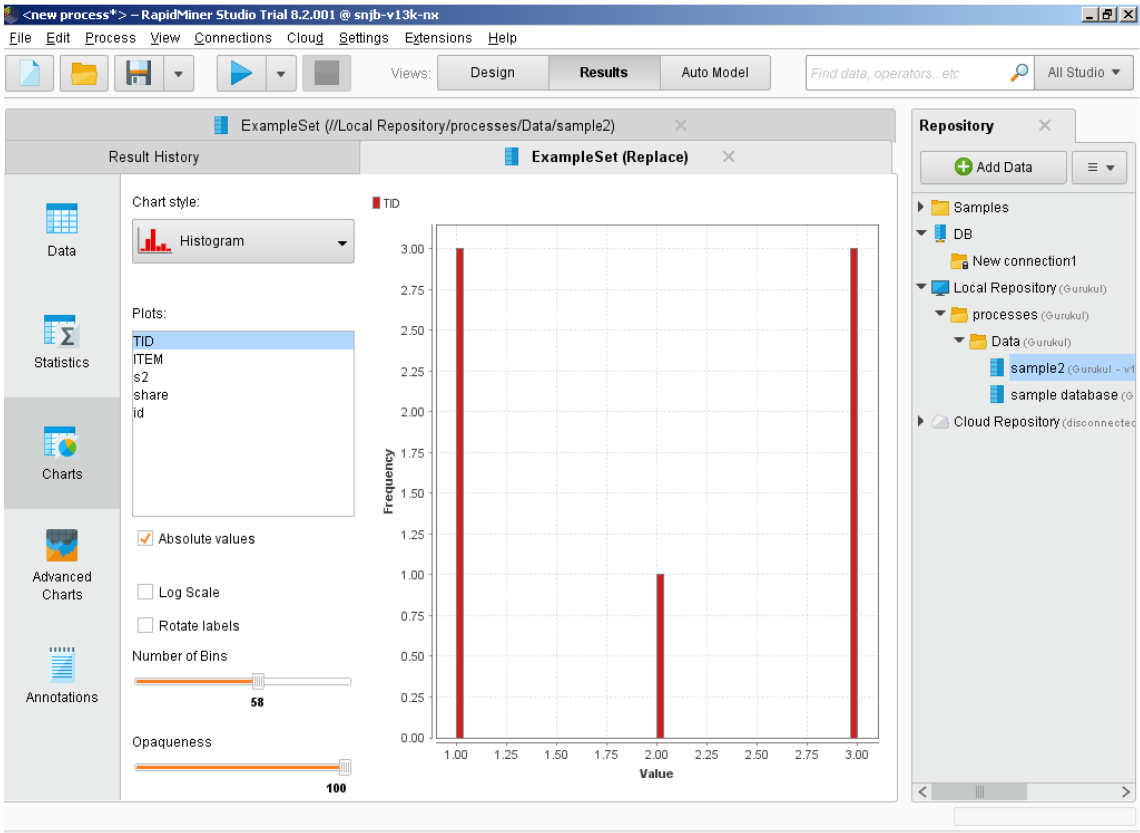
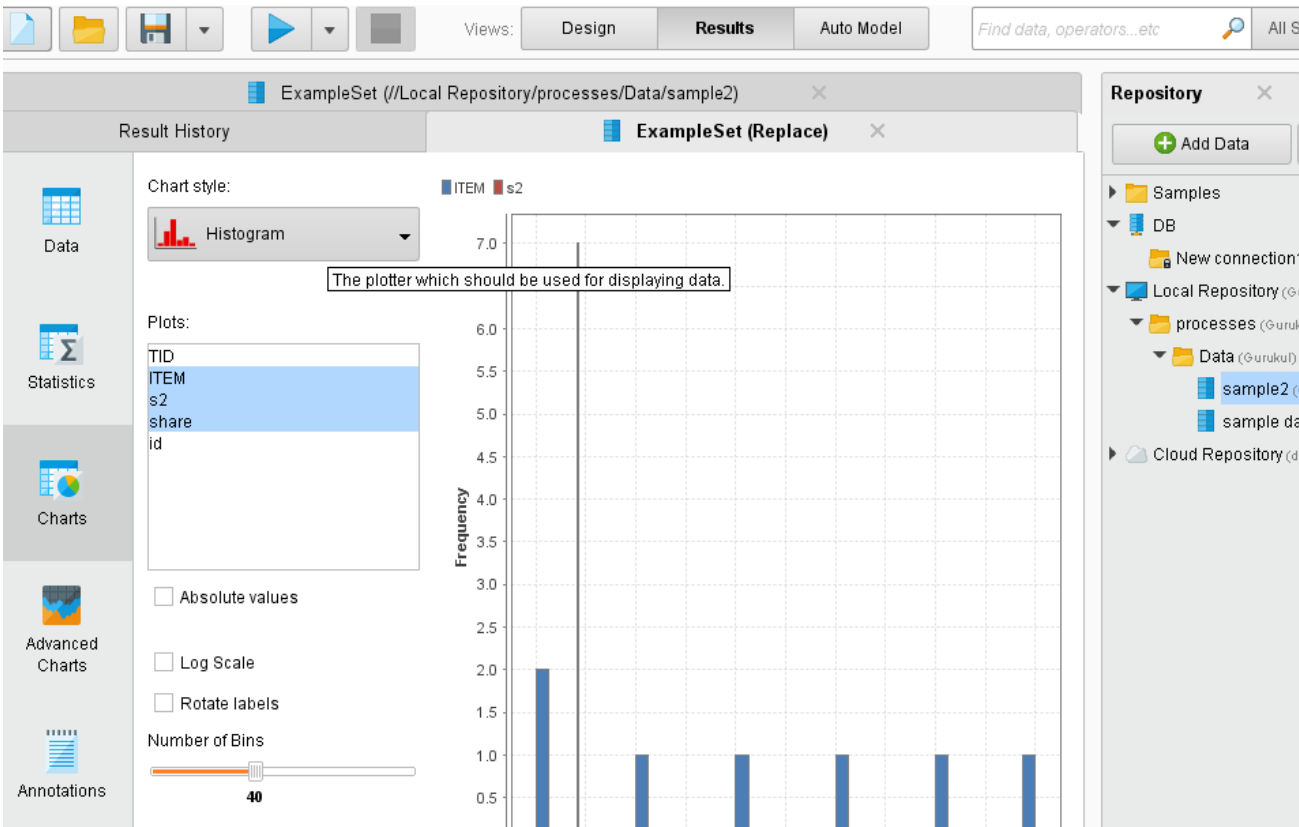
Tags: Save, Export, Write, Datasets, Repository, Data Access

Synopsis

This operator stores an IO Object in the data repository.

Link to Tutorial Process

Step-13 You can also plot Charts of Sample Data set



A nice functionality for data preparation, called [RapidMiner Turbo Prep](#), is where you simply drag and drop data to create amazing interfaces.

Conclusion:

Hence we are able to study Rapidminer. Using these tools, we can perform ETL operations on sample datasets and can perform analysis on sample datasets.