**Experiment 9**

**Aim:**

Experiment to implement Association mining algorithm(Apriori) using Rapid Miner and Python.

**Theory:-**

**Association Mining Rule:**

Association Rule Mining, as the name suggests, association rules are simple If/Then statements that help discover relationships between seemingly independent relational databases or other data repositories. Most machine learning algorithms work with numeric datasets and hence tend to be mathematical. However, association rule mining is suitable for non-numeric, categorical data and requires just a little bit more than simple counting. Association rule mining is a procedure which aims to observe frequently occurring patterns, correlations, or associations from datasets found in various kinds of databases such as relational databases, transactional databases, and other forms of repositories.

The Association rule is a learning technique that helps identify the dependencies between two data items. Based on the dependency, it then maps accordingly so that it can be more profitable. Association rule furthermore looks for interesting associations among the variables of the dataset. It is undoubtedly one of the most important concepts of Machine Learning and has been used in different cases such as association in data mining and continuous production, among others. However, like all other techniques, association in data mining, too, has its own set of disadvantages

**Types of Association Rule Learning**

Association rule learning can be divided into three algorithms:

* **Apriori Algorithm:** This algorithm uses frequent datasets to generate association rules. It is designed to work on the databases that contain transactions. This algorithm uses a breadth-first search and Hash Tree to calculate the itemset efficiently. It is mainly used for market basket analysis and helps to understand the products that can be bought together. It can also be used in the healthcare field to find drug reactions for patients.
* **Eclat Algorithm:** Eclat algorithm stands for Equivalence Class Transformation. This algorithm uses a depth-first search technique to find frequent itemsets in a transaction database. It performs faster than the Apriori Algorithm.
* **F-P Growth Algorithm:** The F-P growth algorithm stands for Frequent Pattern, and it is the improved version of the Apriori Algorithm. It represents the database in the form of a tree structure that is known as a frequent pattern or tree. The purpose of this frequent tree is to extract the most frequent patterns.

The Association mining function computes association rules. The set of the computed association rules is called a rule model. The first part of an association rule is called the rule body. The second part of an association rule is called rule head. You can build an association rule model by using the BuildRuleModel procedure.

* **Transaction tables:** If you use the Associations mining function on transaction tables, for example, retail transaction data, the rule body and the rule head might represent articles that occur in retail transactions, for example, chocolate, or candy.

The association rule might look like this:

If customers buy chocolate, they also buy candy.

* **Relation tables:** If you use the Associations mining function on relational tables, for example, customer data of a bank, the rule body and the rule head might represent column value pairs, for example, online-access=YES.

The association rule might look like this:

If online-access=YES then bankcard=YES

* **Confidence:** An association rule might not always be valid. For example, if the following association rule has a confidence value of 60%, this association rule indicates that if customers buy chocolate only in 60% of the cases they also buy candy:

If chocolate then candy

The degree of validity is indicated by the confidence value of an association rule. The value for confidence is shown in percent. It states how often the association rule head occurs in a transaction given that the rule body occurs in the transaction.

* **Support:** Another attribute of an association rule is the support value. The support value indicates the relative frequency that the conditions in the rule head and in the rule body hold.

For example:

If the following association rule has a support value of 2%, it means that 2% of all sales transactions contain chocolate and candy:

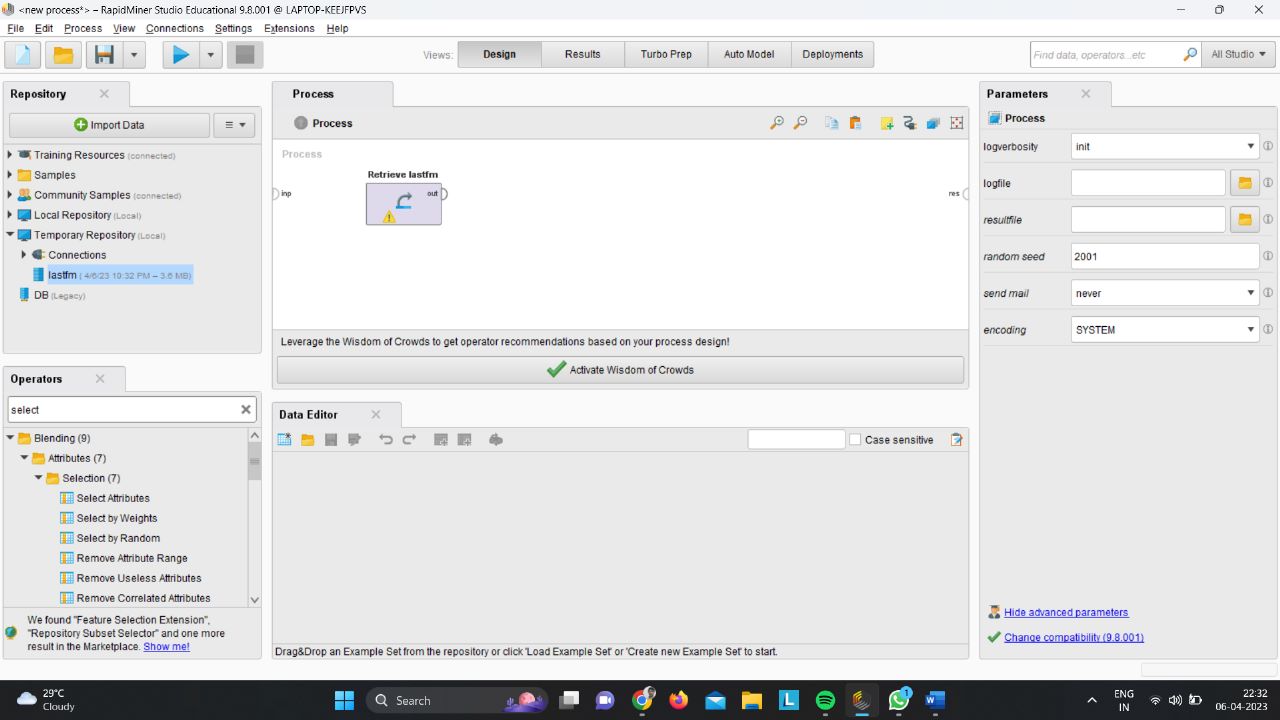
If chocolate then candy

If the following association rule has a support value of 4%, it means that for 4% of all records the values of the ONLINE\_ACCESS column and the BANKCARD column are both YES.

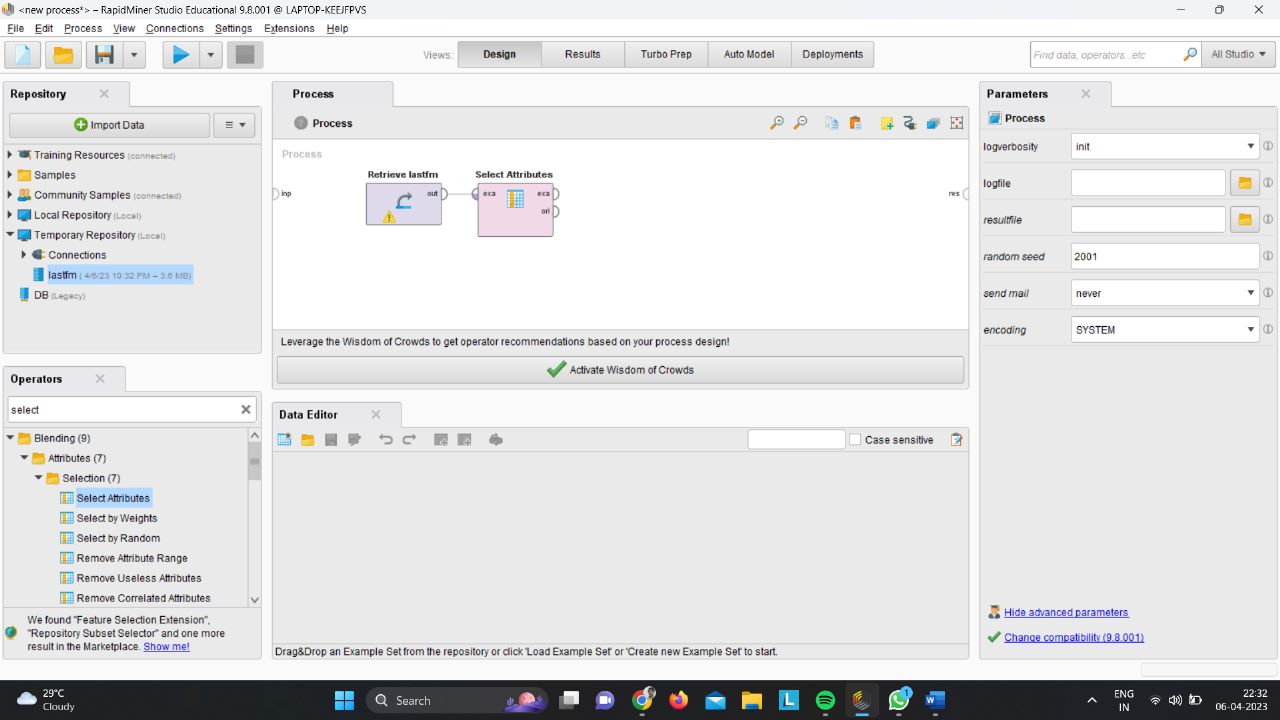
If online-access=YES then bankcard=YES

**Implementing Association Mining using Rapid Miner**

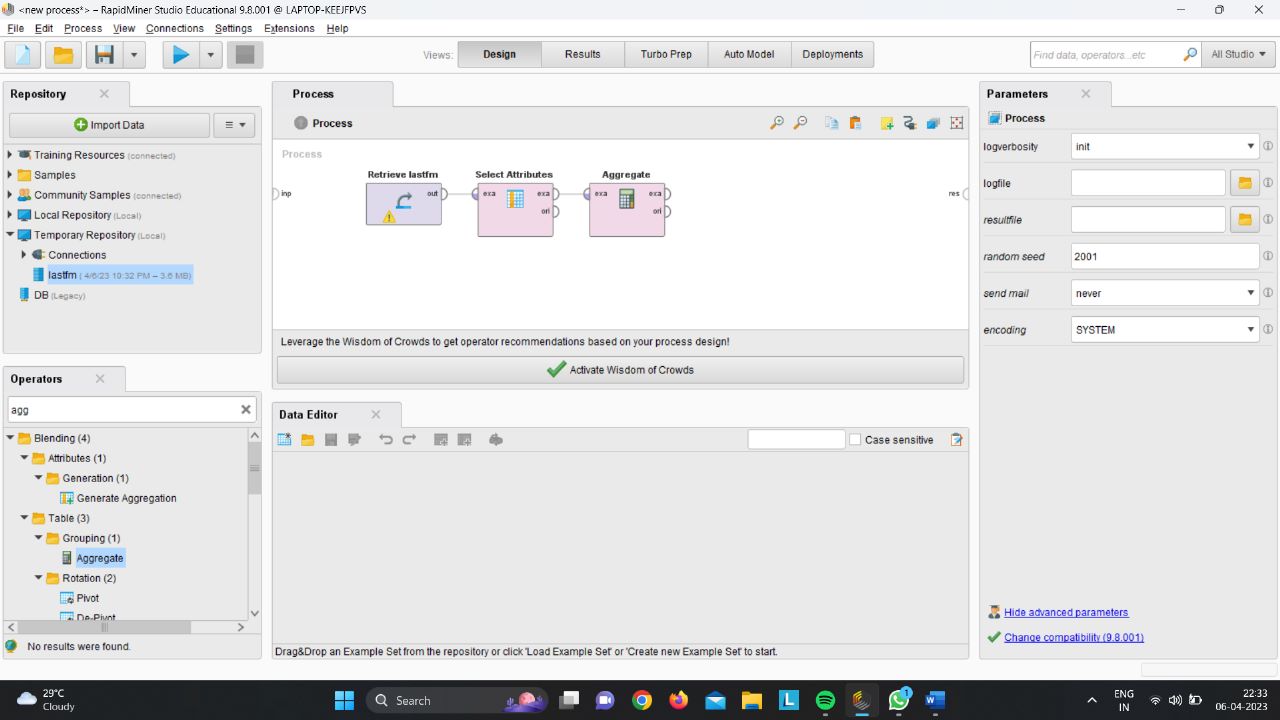
**Import Dataset -**

****

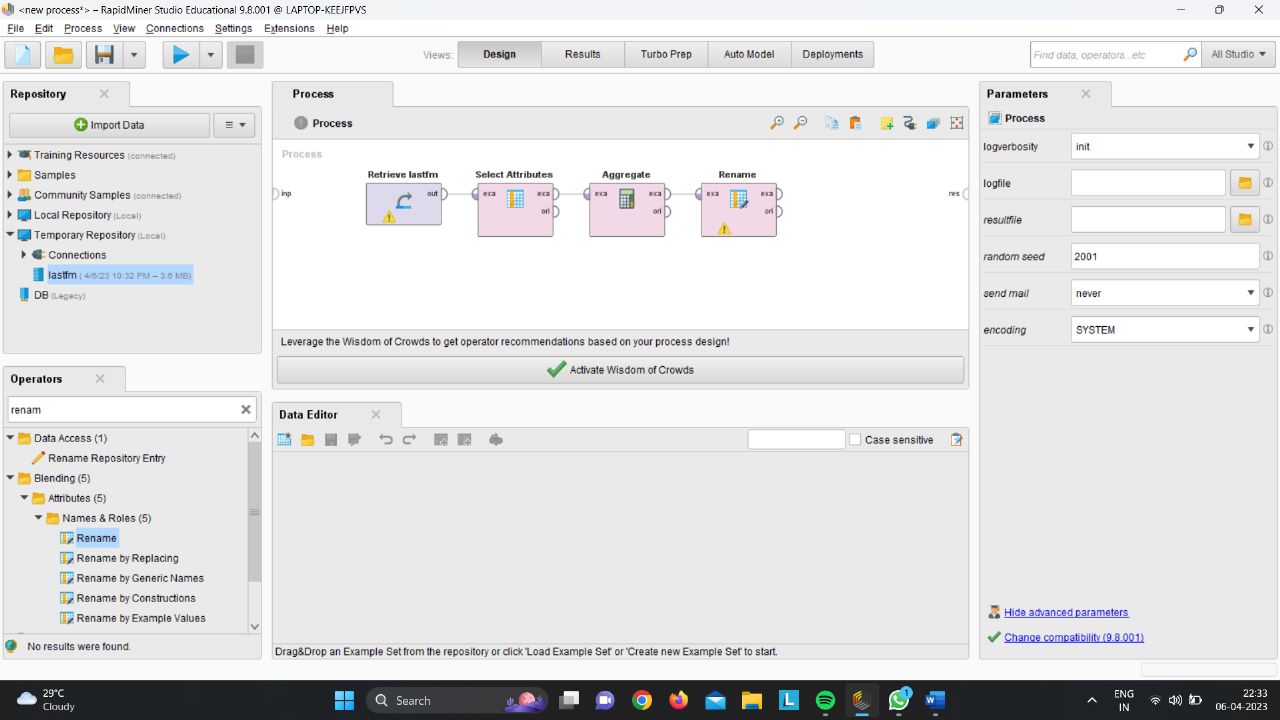
**Select Attributes -**

****

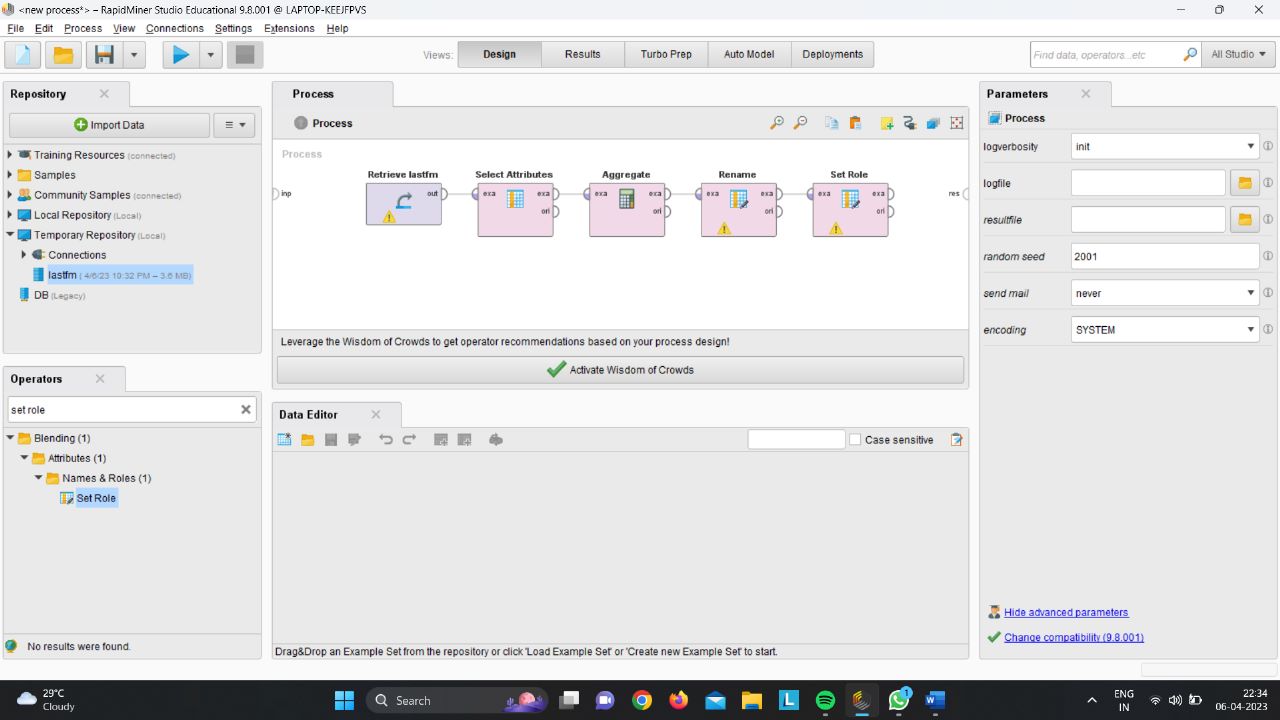
**Data Aggregation -**

****

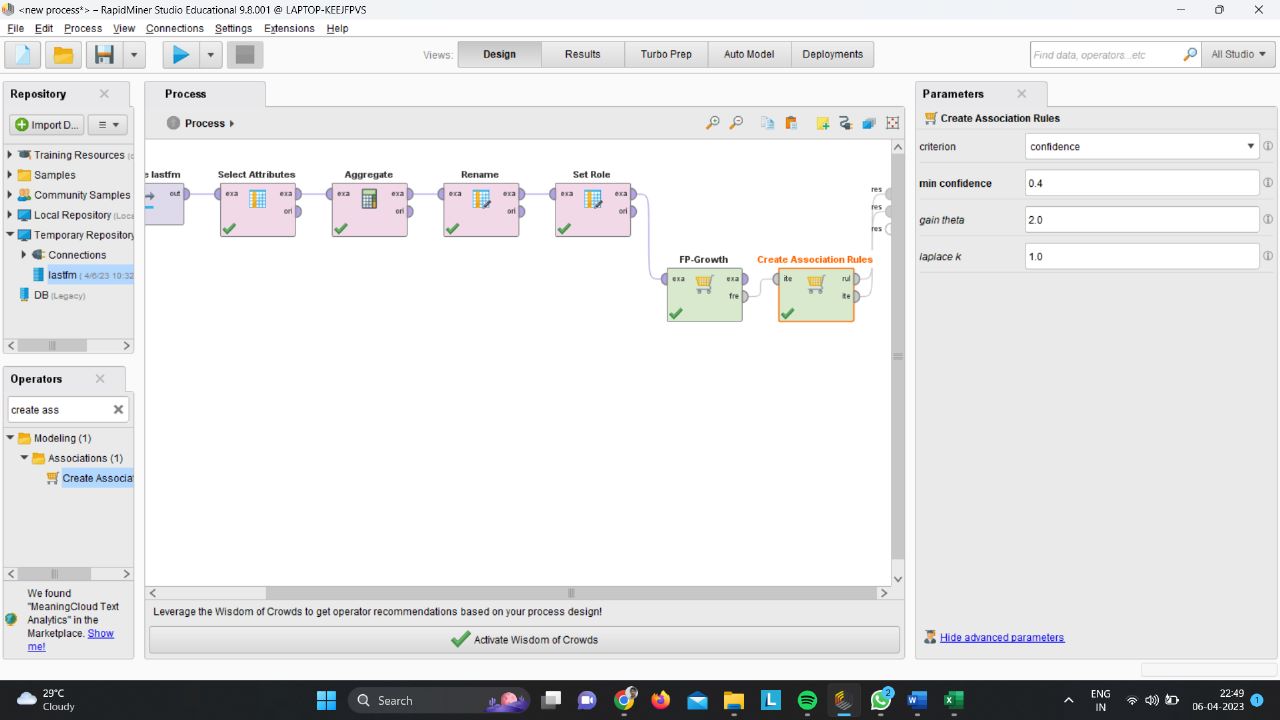
**Rename -**

****

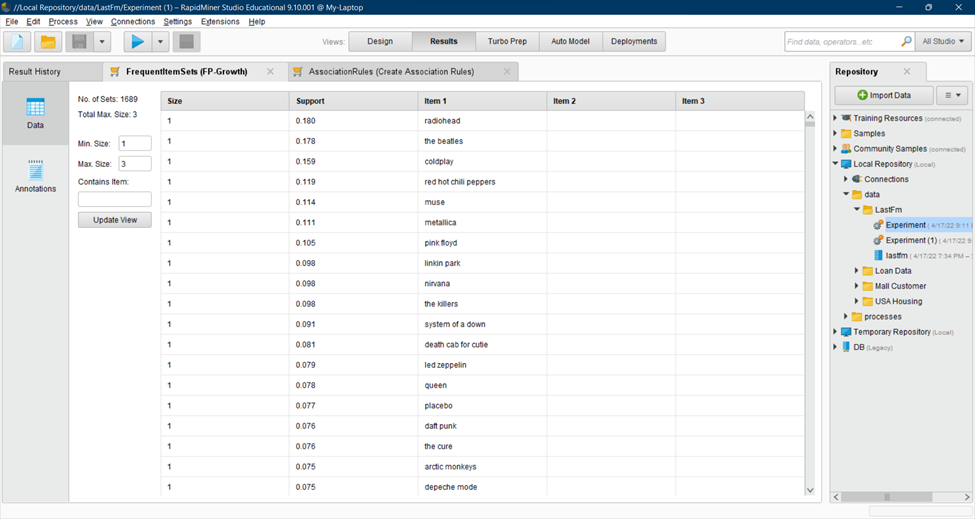
**Set role -**

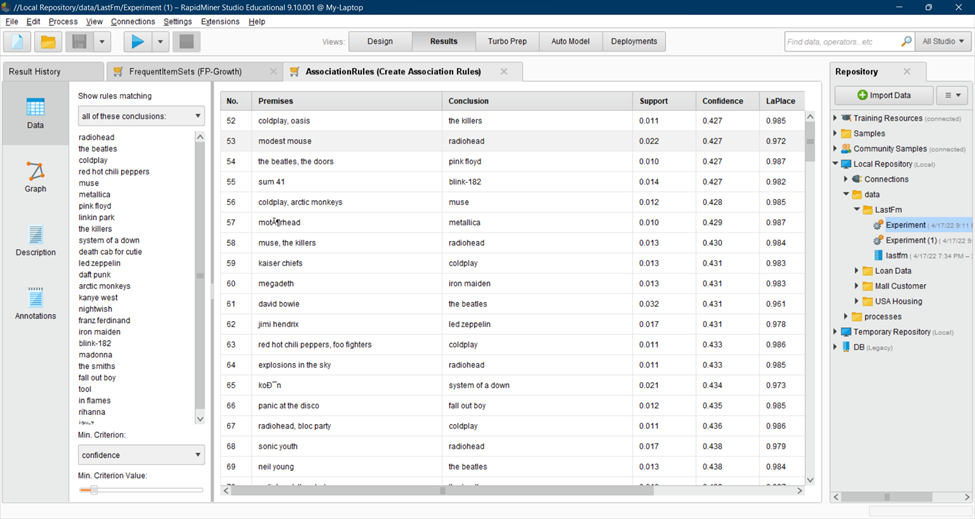
****

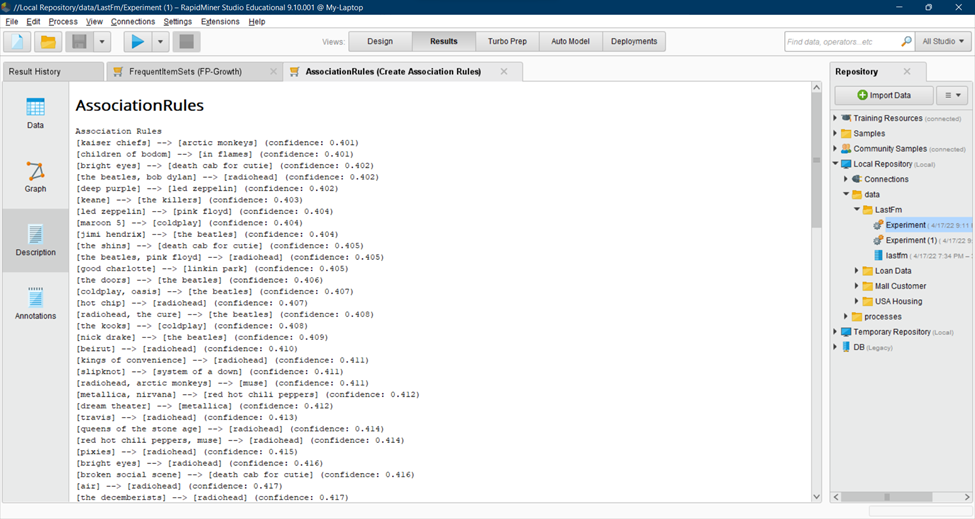
**Create Association Rules -**

****

**Results -**

****

****

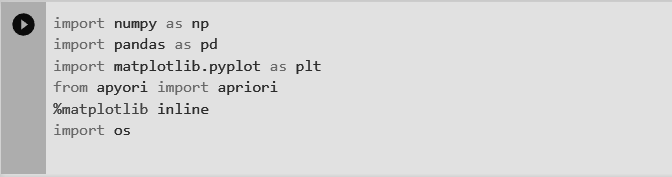
****

**Implementing Association Mining using Python**

**First install Apriori**

****

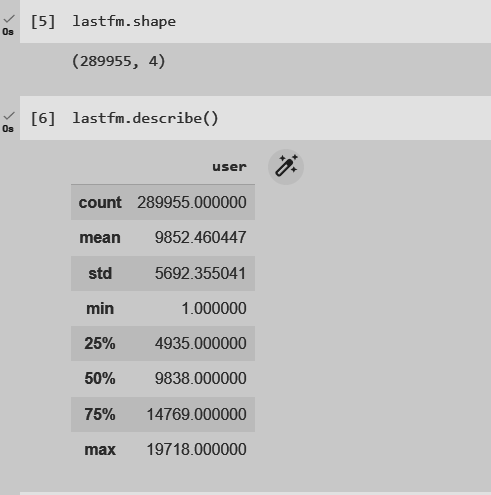
**Importing Library**

****

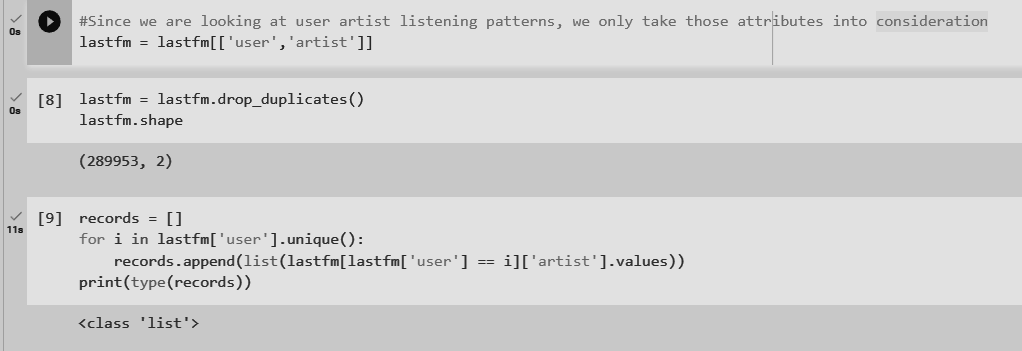
**Importing dataset -**

****

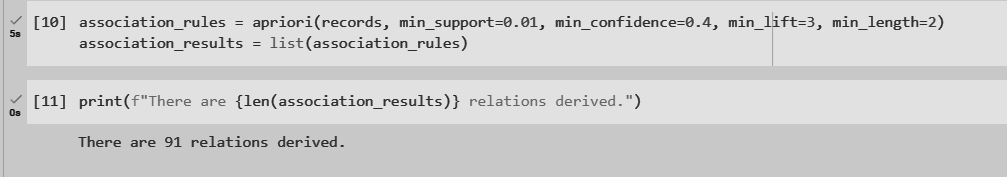
**Shape and description -**

****

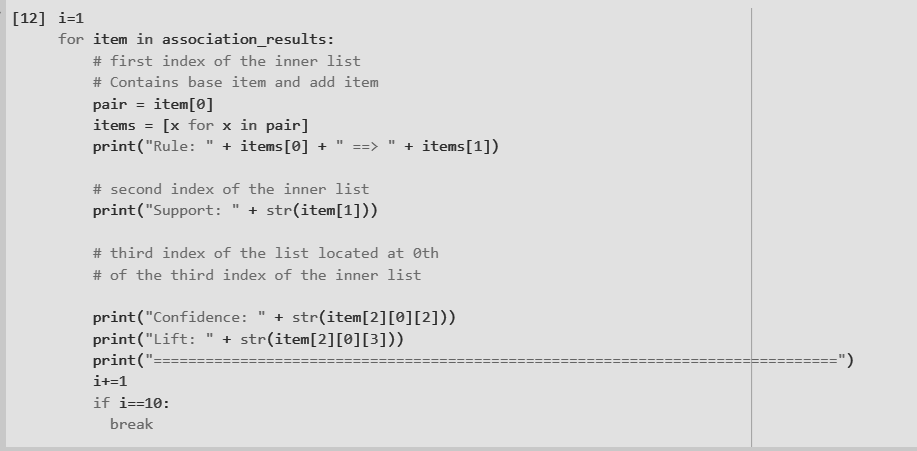
**Removing Duplicates**

****

**Using Apriori Algorithm -**

****

**Results of few rules-**

****

****

**Comparison of these implementations**

At min support count of 0.01 and confidence of 0.4 the following table represents the number of rules which are generated.

| **Python** | **91** |
| --- | --- |
| **Rapid Miner** | **212** |

**Conclusion**

Thus we have learnt what association mining is, how it is important and also implemented different ways to perform association mining.