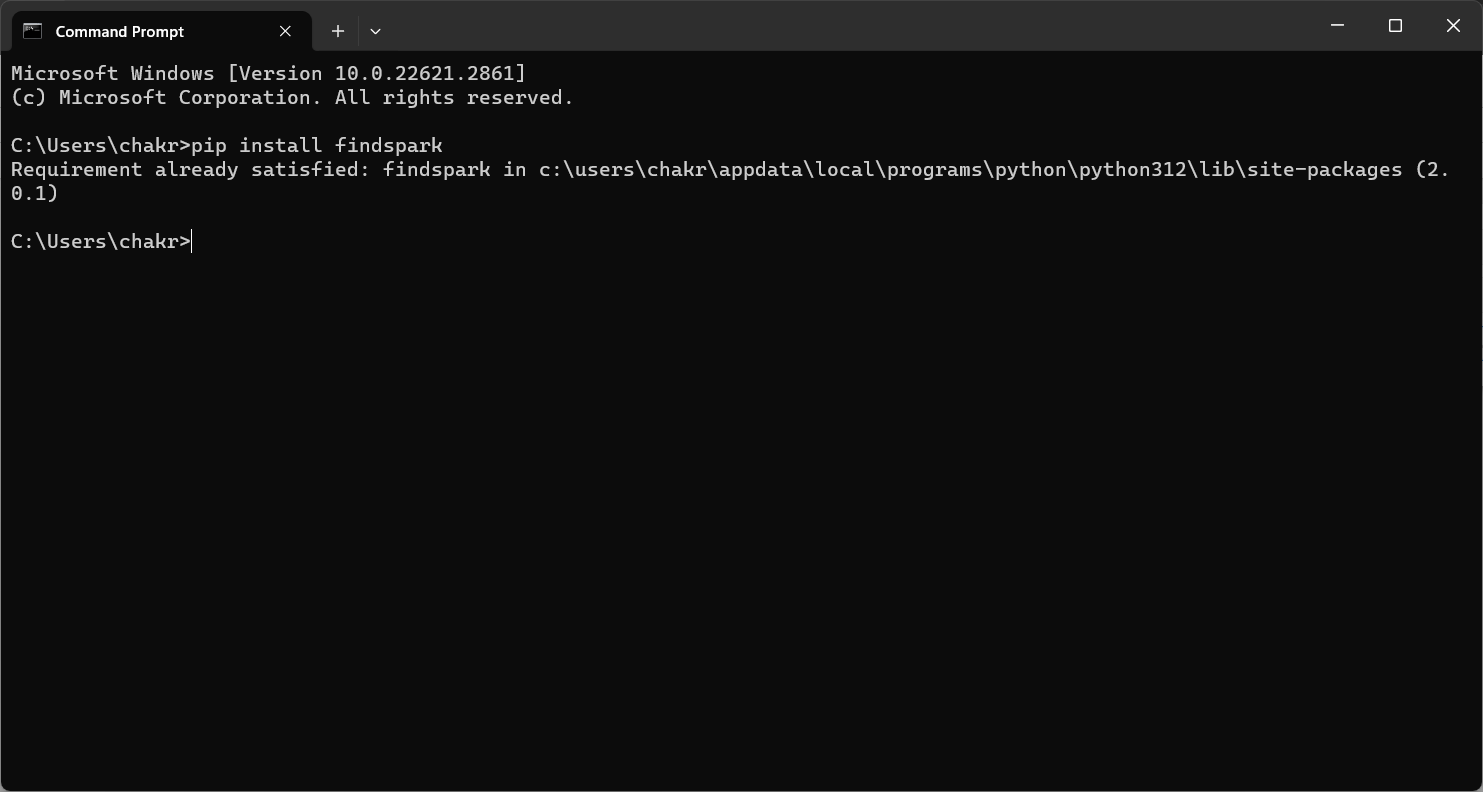
26/12/2023

Chakradhar Bhogapurapu, BIBA

Pyspark

Execution of a program in Jupyter notebook

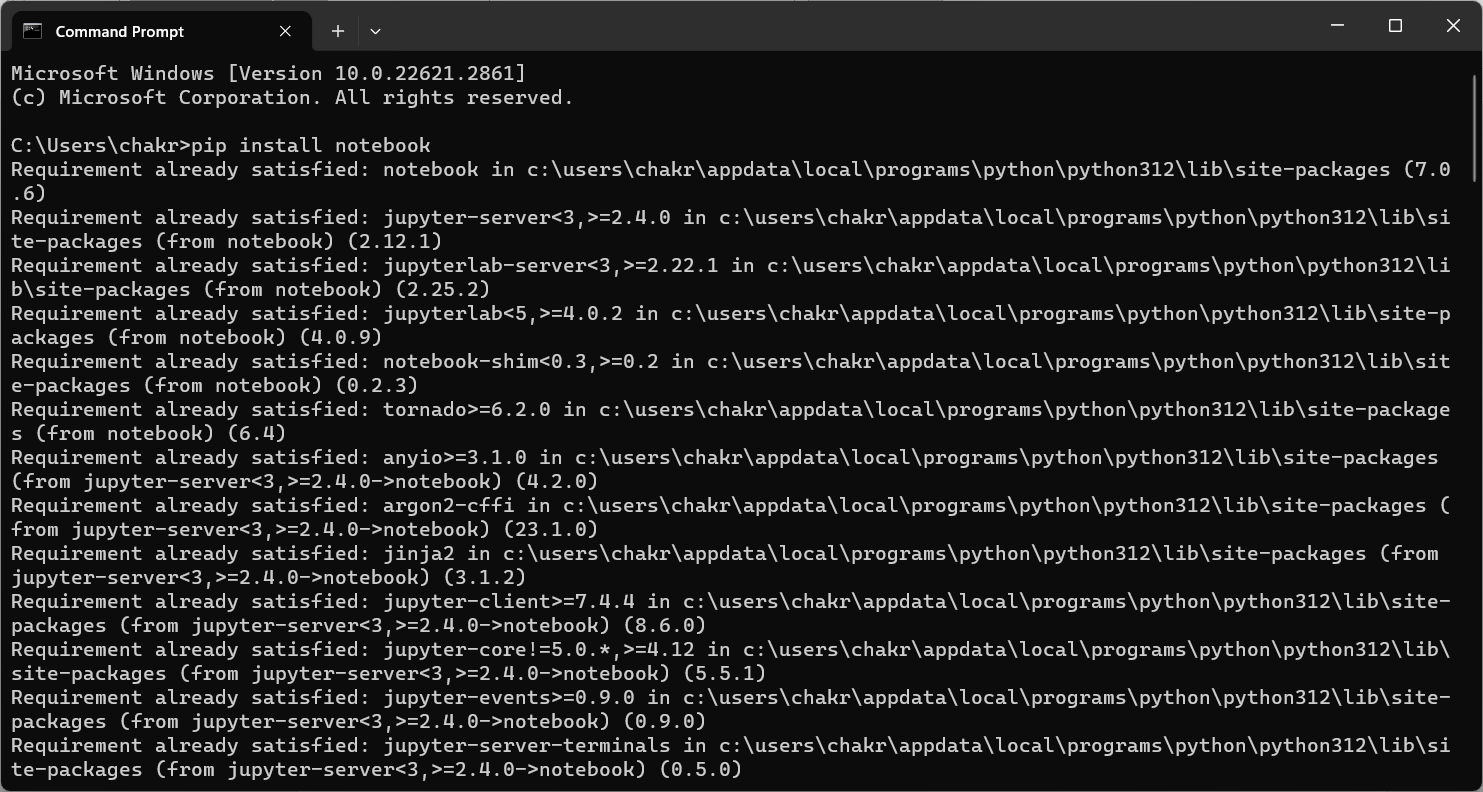
We need to give “pip install findspark” in command prompt in our local system



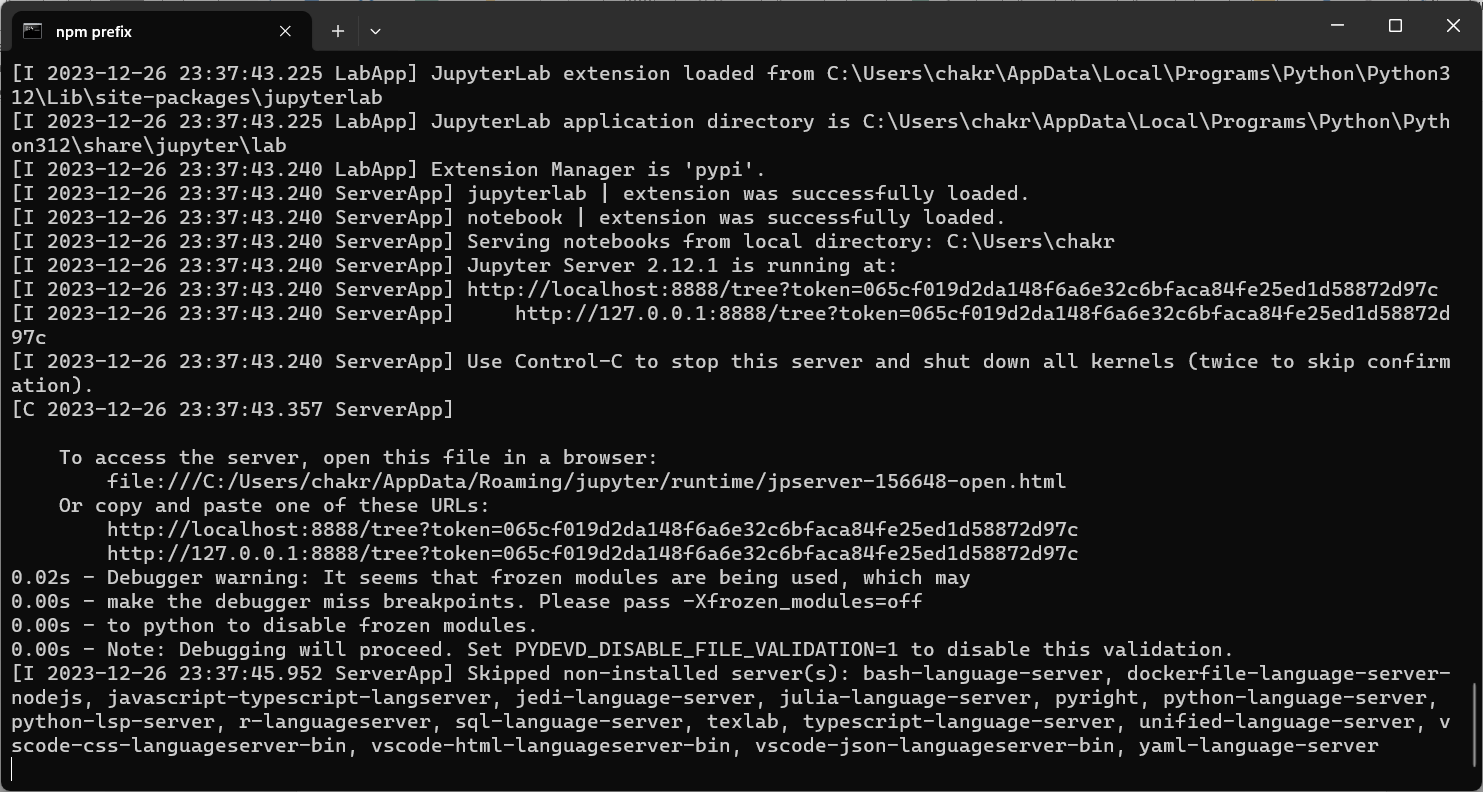
To execute a program in jupyter notebook, we need to have a command to install jupyter. i.e.,

pip install notebook

This command will install notebook in local system

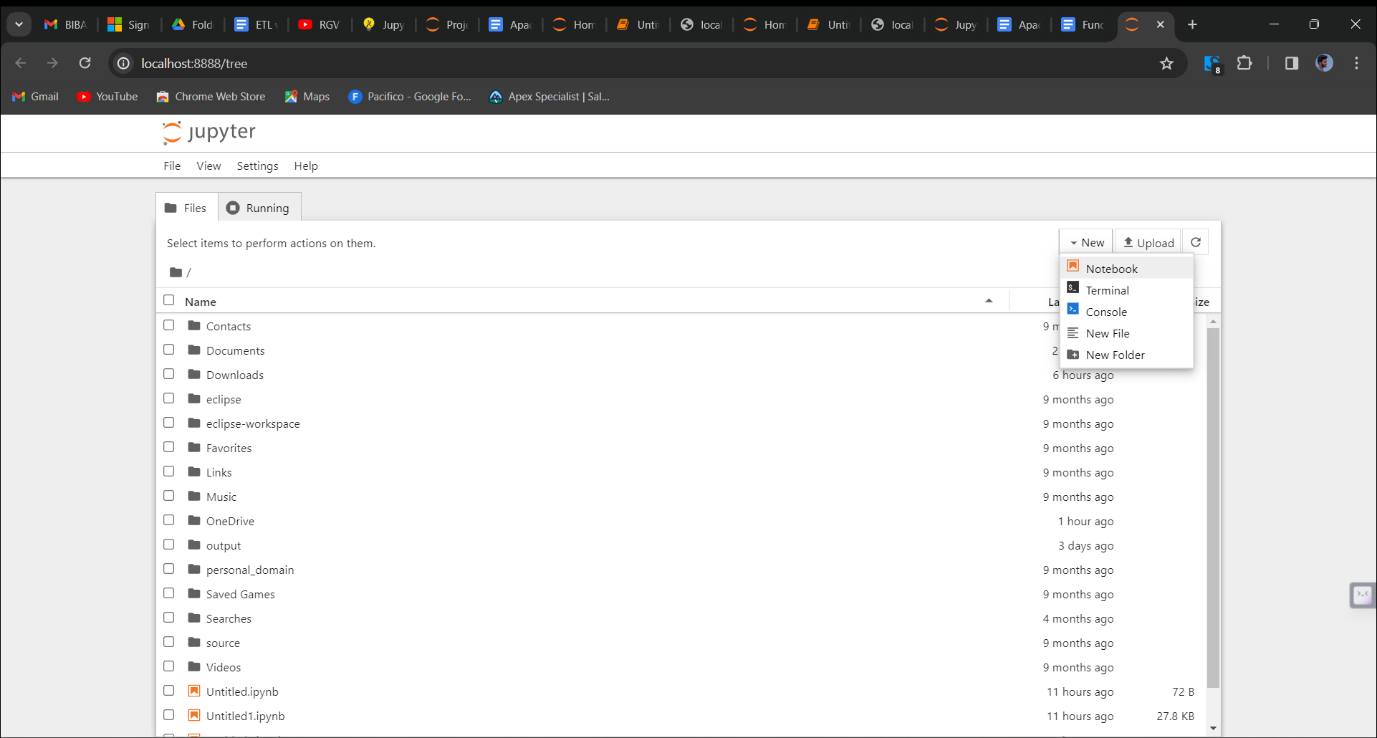


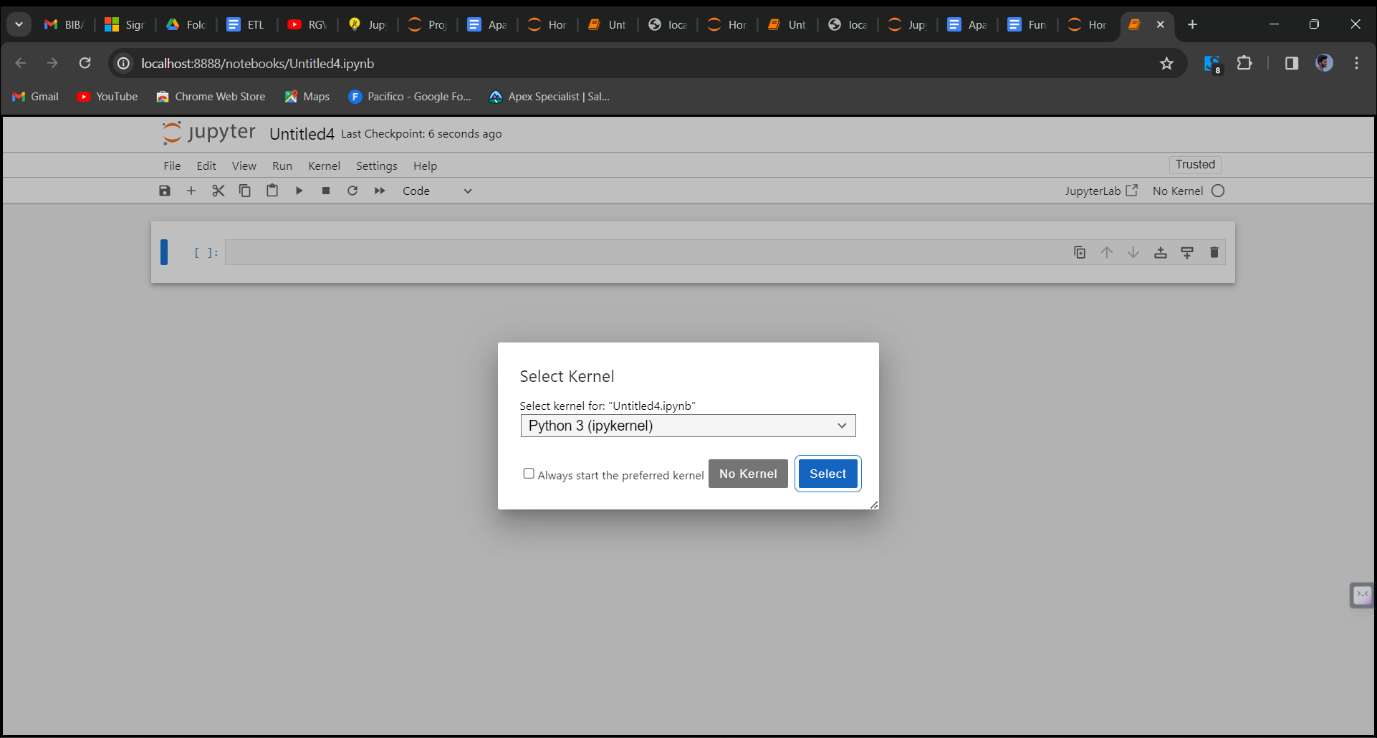
Now give the command as ”jupyter notebook” in command prompt



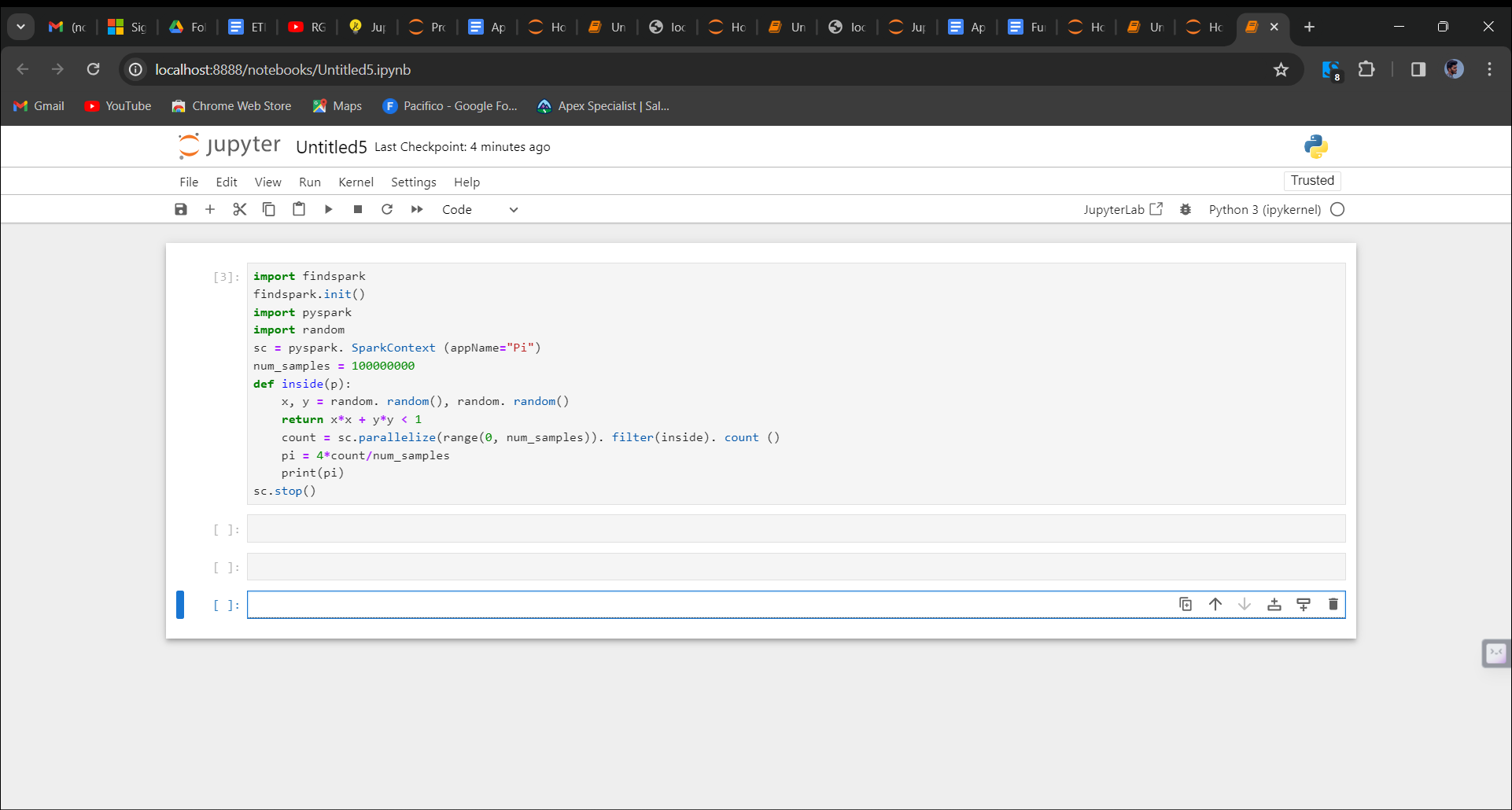
Now We get a web page opened in browser directly

Click on new file on notebook





The program is executed in the notebook

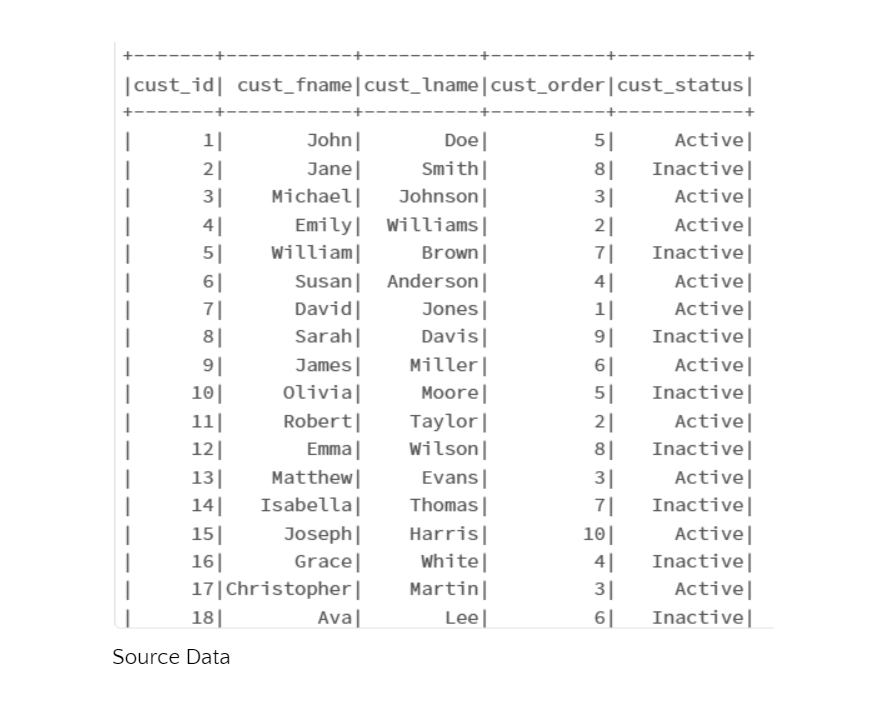


ETL(Extract, Transform, Load)

Extract : Retrieve data from various sources like databases, files or API’s.

Transform : Clean, aggregate, and manipulate data to fit your analysis needs.

Load : Store the transformed data inyodatabase or data warehouse for analysis.



🡪Initialize a spark session

Spark = SparkSession.builder.appName(“ComplexETL”).getOrCreate()

🡪Define external source and target paths

Source\_path = ‘your\_path’ target\_path = “your\_output\_path”

🡪Extract: Read data from ean external CSV file.

df = spark.read.csv()source\_path, header=True, schema = ‘cust\_id’ int, first\_name string, last\_name string, cust\_order int, cust\_status string’)

🡪Transformation 1:

To concatenate first and last names

df = df.withColumn(“full\_name”, concat(col(“first\_name”),lit(“ ”), col(“last\_name”)))

🡪Transformation 2:

To calculate net salary(subtract 10% as taxes)

df = df.withColumn(“net\_salary”, floor(lit(10000)+rand()\*lit(50)))

🡪Adding age column

df = df.withColumn(“age”, floor(lit(20)+rand()\*lit(31)))

🡪Transformation 3:

To filter by age

df = df.filter(col(“age”)>=30)

🡪Transformation 4:

To group by age and calculate average salary

Avg\_salary\_by\_age = df.groupBy(“age”).agg({“net\_salary”:”avg”}).withColumnRenamed(“avg(salary)”,”avg\_salary”)

🡪Transformation 5:

df = df.orderBy(“age”)

🡪Save the transformed data to an external CSV file

df.write.csv(target\_path, mode=”overwrite”, header = True)

