Talha Khan (2303.009.KHI.DEG)

Muhammad Moiz Khan (2303.022.KHI.DEG)

ASSIGNMNET 5.1

In [3]:

```
from pyspark.sql.functions import *
from pyspark.sql import SparkSession
```

In [5]:

```
spark = SparkSession.builder.appName("task 5.1").getOrCreate()
spark
```

Out[5]: **SparkSession - in-memory**

SparkContext

Spark UI

Version v3.4.0
Master local[*]
AppName task 5.1

In [6]:

```
df_transactions = spark.read.csv('data/store_transactions/transactions_*.csv', header=True, inferSchema=True)
df_transactions.show()
```

```
+-------+-------------+----------+---------+--------+-------------------+
|StoreId|TransactionId|CustomerId|ProductId|Quantity|    TransactionTime|
+-------+-------------+----------+---------+--------+-------------------+
|      3|          454|        35|        3|       3|2022-12-23 17:36:11|
|      3|          524|        37|        9|      11|2022-12-23 22:02:51|
|      3|          562|         4|        3|       4|2022-12-23 02:51:50|
|      3|          581|        35|       14|      56|2022-12-23 17:05:54|
|      3|          200|        34|       15|      24|2022-12-23 07:15:01|
|      3|          506|        41|       24|      19|2022-12-23 21:26:29|
|      3|          278|         5|        1|       5|2022-12-23 16:41:42|
|      3|          849|        36|       23|      13|2022-12-23 13:22:55|
|      3|          992|        34|        7|       3|2022-12-23 16:47:14|
|      3|          703|        19|        7|      13|2022-12-23 22:36:48|
|      3|          719|        48|       18|      12|2022-12-23 10:11:29|
|      3|          526|        13|       14|       3|2022-12-23 11:57:23|
|      3|          997|        20|        1|      14|2022-12-23 04:02:30|
|      3|          281|        11|       15|      25|2022-12-23 16:07:45|
|      3|          691|        48|       23|       2|2022-12-23 08:12:00|
|      3|          762|        17|        5|      26|2022-12-23 16:18:27|
|      3|          106|        24|       23|      11|2022-12-23 07:41:50|
|      3|           21|        32|        9|       2|2022-12-23 21:15:10|
|      3|          626|        14|       18|      14|2022-12-23 12:55:02|
|      3|          219|        11|       15|       5|2022-12-23 13:00:17|
+-------+-------------+----------+---------+--------+-------------------+
only showing top 20 rows
```

In [7]:

```
df_products = spark.read.csv('data/products.csv', header=True, inferSchema=True)
df_products.show()
```

```
+---------+-------------+----------+---------+
|ProductId|         Name|  Category|UnitPrice|
+---------+-------------+----------+---------+
|        1|   Red Shorts|    Shorts|    89.75|
|        2| White Shorts|    Shorts|    89.27|
|        3|  Blue Shorts|    Shorts|   118.88|
|        4| Green Shorts|    Shorts|   121.43|
|        5| Black Shorts|    Shorts|    74.58|
|        6|  Red Sandals|     Shoes|   138.38|
|        7|White Sandals|     Shoes|   160.96|
|        8|Blue Sneakers|     Shoes|    111.7|
|        9|Green Sandals|     Shoes|   137.53|
|       10|Black Sneakers|     Shoes|   146.41|
|       11|        Watch|Accesories|   179.65|
|       12|     Bracelet|Accesories|   160.77|
|       13|     Earrings|Accesories|    185.9|
|       14|  Red t-shirt|  T-Shirts|   121.58|
|       15|White t-shirt|  T-Shirts|   131.13|
|       16| Blue t-shirt|  T-Shirts|   140.68|
|       17|Green t-shirt|  T-Shirts|   130.13|
|       18|Black t-shirt|  T-Shirts|   102.41|
|       19| Green jacket|   Jackets|   223.69|
|       20| Black jacket|   Jackets|   190.01|
+---------+-------------+----------+---------+
only showing top 20 rows
```

In [8]:

```
df_customers = spark.read.csv('data/customers.csv', header=True, inferSchema=True)
df_customers.show()
```

```
+----------+--------------------+--------------------+
|CustomerId|                Name|               Email|
+----------+--------------------+--------------------+
|         1|      Emilia Pedraza|emilia.pedraza@ex...|
|         2|        Thies Blümel|thies.blumel@exam...|
|         3|       بهاره علیزاده|bhrh.aalyzdh@exam...|
|         4|       Alevtin Paska|alevtin.paska@exa...|
|         5|      Charlotte Wong|charlotte.wong@ex...|
|         6|     Vittorio Bonnet|vittorio.bonnet@e...|
|         7|          Dominic Lo|dominic.lo@exampl...|
|         8|        کیان علیزاده|kyn.aalyzdh@examp...|
|         9|       Babür Çörekçi|babur.corekci@exa...|
|        10|          تینا یاسمی|tyn.ysmy@example.com|
|        11|    Angélique Vennix|angelique.vennix@...|
|        12|           Eric King|eric.king@example...|
|        13|      Elizabeth Neal|elizabeth.neal@ex...|
|        14|      Sylvie Lecomte|sylvie.lecomte@ex...|
|        15|           An Jansen|an.jansen@example...|
|        16|      Signe Petersen|signe.petersen@ex...|
|        17|Sevastiana Nester...|sevastiana.nester...|
|        18|          Kiara Brun|kiara.brun@exampl...|
|        19|       Alexia Renaud|alexia.renaud@exa...|
|        20|         Suzy Gibson|suzy.gibson@examp...|
+----------+--------------------+--------------------+
only showing top 20 rows
```

Data Preprocessing

```
In [9]:
df_joined = df_transactions.join(
df_products,
df_transactions['ProductId'] == df_products['ProductId'],
'inner')\
.select(df_transactions['*'], df_products.Name.alias('ProductName'), df_products.UnitPrice)
df_joined.printSchema()
```

```
root
 |-- StoreId: integer (nullable = true)
 |-- TransactionId: integer (nullable = true)
 |-- CustomerId: integer (nullable = true)
 |-- ProductId: integer (nullable = true)
 |-- Quantity: integer (nullable = true)
 |-- TransactionTime: timestamp (nullable = true)
 |-- ProductName: string (nullable = true)
 |-- UnitPrice: double (nullable = true)
```

In [10]:

```
df_joined = df_joined.join(
df_customers,
df_joined['CustomerId'] == df_customers['CustomerId'],
'inner')\
.select(df_joined['*'], df_customers.Email)
df_joined.printSchema()
```

```
root
 |-- StoreId: integer (nullable = true)
 |-- TransactionId: integer (nullable = true)
 |-- CustomerId: integer (nullable = true)
 |-- ProductId: integer (nullable = true)
 |-- Quantity: integer (nullable = true)
 |-- TransactionTime: timestamp (nullable = true)
 |-- ProductName: string (nullable = true)
 |-- UnitPrice: double (nullable = true)
 |-- Email: string (nullable = true)
```

In [11]:

```
df_joined = df_joined.withColumn('Sales', df_joined['Quantity']*df_joined['UnitPrice'])
df_joined.select(['Quantity', 'UnitPrice', 'Sales']).show()
```

```
+--------+---------+------------------+
|Quantity|UnitPrice|             Sales|
+--------+---------+------------------+
|       3|   118.88|            356.64|
|      11|   137.53|           1512.83|
|       4|   118.88|            475.52|
|      56|   121.58|           6808.48|
|      24|   131.13|           3147.12|
|      19|    173.1|            3288.9|
|       5|    89.75|            448.75|
|      13|   150.93|1962.0900000000001|
|       3|   160.96|            482.88|
|      13|   160.96|           2092.48|
|      12|   102.41|           1228.92|
|       3|   121.58|            364.74|
|      14|    89.75|            1256.5|
|      25|   131.13|           3278.25|
|       2|   150.93|            301.86|
|      26|    74.58|           1939.08|
|      11|   150.93|           1660.23|
|       2|   137.53|            275.06|
|      14|   102.41|           1433.74|
|       5|   131.13|            655.65|
+--------+---------+------------------+
only showing top 20 rows
```

In [13]:

```
# cast timestamp to datetime
df_joined = df_joined.withColumn('TransactionDate', to_date(df_joined['TransactionTime']))
df_joined.select(['TransactionTime', 'TransactionDate']).show()
```

```
+-------------------+---------------+
|    TransactionTime|TransactionDate|
+-------------------+---------------+
|2022-12-23 17:36:11|     2022-12-23|
|2022-12-23 22:02:51|     2022-12-23|
|2022-12-23 02:51:50|     2022-12-23|
|2022-12-23 17:05:54|     2022-12-23|
|2022-12-23 07:15:01|     2022-12-23|
|2022-12-23 21:26:29|     2022-12-23|
|2022-12-23 16:41:42|     2022-12-23|
|2022-12-23 13:22:55|     2022-12-23|
|2022-12-23 16:47:14|     2022-12-23|
|2022-12-23 22:36:48|     2022-12-23|
|2022-12-23 10:11:29|     2022-12-23|
|2022-12-23 11:57:23|     2022-12-23|
|2022-12-23 04:02:30|     2022-12-23|
|2022-12-23 16:07:45|     2022-12-23|
|2022-12-23 08:12:00|     2022-12-23|
|2022-12-23 16:18:27|     2022-12-23|
|2022-12-23 07:41:50|     2022-12-23|
|2022-12-23 21:15:10|     2022-12-23|
|2022-12-23 12:55:02|     2022-12-23|
|2022-12-23 13:00:17|     2022-12-23|
+-------------------+---------------+
only showing top 20 rows
```

What are the daily total sales for the store with id 1?

In [15]:

```
df_joined.filter(df_joined['StoreId'] == 1)\
.groupBy('TransactionDate')\
.agg(sum('Sales').alias('DailySales'))\
.show()
```

```
+---------------+------------------+
|TransactionDate|        DailySales|
+---------------+------------------+
|     2022-12-23|41264.000000000015|
+---------------+------------------+
```

What are the mean sales for the store with id 2?

In [20]:

```
df_joined.filter(df_joined['StoreId'] == 2)\
.agg(mean('Sales').alias('MeanSales'))\
.show()
```

```
+-----------------+
|        MeanSales|
+-----------------+
|513.4598039215689|
+-----------------+
```

What is the email of the client who spent the most when summing up purchases from all of the stores?

In [21]:

```
df_joined.groupBy('Email')\
.agg(sum('Sales').alias('TotalSpending'))\
.orderBy(col('TotalSpending').desc())\
.limit(1)\
.show(truncate=False)
```

```
+------------------------+-------------+
|Email                   |TotalSpending|
+------------------------+-------------+
|dwayne.johnson@gmail.com|10653.08     |
+------------------------+-------------+
```

Which 5 products are most frequently bought across all stores?

In [22]:

```
df_joined.groupBy(['ProductName'])\
.agg(sum('Quantity').alias('TimesBought'))\
.orderBy(col('TimesBought').desc())\
.limit(5)\
.show()
```

```
+-------------+-----------+
|  ProductName|TimesBought|
+-------------+-----------+
|  Red t-shirt|         82|
|   Blue Jeans|         77|
|White t-shirt|         76|
| Black Shorts|         75|
| Green jacket|         74|
+-------------+-----------+
```

In [ ]: