Peer Members

Maaz Javaid Siddique (2303.KHI.DEG.004)

Syed Muhammad Raqim Ali Shah (2303.KHI.DEG.008)

Qadeer Hussain (2303.KHI.DEG.006)

In [1]:
```
from pyspark.sql import SparkSession
from pyspark.sql.types import IntegerType
from pyspark.sql.functions import col,sum,count,mean
```

In [2]:
```
spark = SparkSession.builder.appName("PySpark Assignment").getOrCreate()
```

```
23/05/16 08:32:23 WARN Utils: Your hostname, all-MS-7D35 resolves to a loopback address: 127.0.1.1; using 192.168.1.
154 instead (on interface enp2s0)
23/05/16 08:32:23 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/05/16 08:32:23 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-jav
a classes where applicable
```

In [3]:
```
transaction_1 = spark.read.csv("./store_transactions/transactions_1.csv", header=True)
transaction_2 = spark.read.csv("./store_transactions/transactions_2.csv", header=True)
transaction_3 = spark.read.csv("./store_transactions/transactions_3.csv", header=True)
customers = spark.read.csv("customers.csv", header=True)
products = spark.read.csv("products.csv", header=True)
```

Problem 1

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

In [4]:
```
store_1_transaction=transaction_1.join(products,transaction_1.ProductId ==  products.ProductId,"inner")
result_table = store_1_transaction.withColumn("TotalPrice", col("Quantity") * col("UnitPrice"))
total_price_sum = result_table.agg(sum("TotalPrice")).collect()[0][0]
print("Total sum of Store Id 1:",total_price_sum)
```

```
Total sum of Store Id 1: 41264.000000000015
```

Problem 2

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

In [5]:
```
store_2_transaction=transaction_2.join(products,transaction_2.ProductId ==  products.ProductId,"inner")
result_table = store_2_transaction.withColumn("TotalPrice", col("Quantity") * col("UnitPrice"))
mean_product_price = result_table.agg(mean("TotalPrice")).collect()[0][0]
print("Mean of Store Id 2:",mean_product_price)
```

```
Mean of Store Id 2: 513.4598039215689
```

Problem 3

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

In [6]:
```
transaction= transaction_1.union(transaction_2).union(transaction_3)
transaction_join=transaction.join(customers,transaction.CustomerId==customers.CustomerId).join(products,transaction
transaction_totalSales=transaction_join.withColumn("TotalSales",col("Quantity")*col("UnitPrice"))
transaction_group_withEmail=transaction_totalSales.groupBy("Email").agg(sum("TotalSales").alias("TotalSales"))
transaction_Email=transaction_group_withEmail.orderBy(col("TotalSales").desc())
email = transaction_Email.select("Email").first()[0]
print("Email of the client:",email)
```

```
Email of the client: dwayne.johnson@gmail.com
```

Problem 4

Which 5 products are most frequently bought across all stores?

In [7]:
```
transaction= transaction_1.union(transaction_2).union(transaction_3)
transaction_join=transaction.join(products,transaction.ProductId==products.ProductId)
transaction_count=transaction_join.groupBy("Name").count()
transaction_order=transaction_count.orderBy(col("count").desc())
transaction_order.show(5)
```

```
+-------------+-----+
|         Name|count|
+-------------+-----+
| White Shorts|   20|
| Black Shorts|    9|
| Green jacket|    9|
|White t-shirt|    8|
|   Red Shorts|    7|
+-------------+-----+
only showing top 5 rows
```