databricksExercise7-Assessment1

```
import pyspark
from pyspark.sql.types import
StructType,StringType,IntegerType,DoubleType,FloatType,DateType

ex5_schema = StructType() \
        .add("InvoiceNo",StringType(),False) \
        .add("StockCode",StringType(),False) \
        .add("Description",StringType(),True) \
        .add("Quantity",IntegerType(),True) \
        .add("InvoiceDate",StringType(),True) \
        .add("UnitPrice",DoubleType(),True) \
        .add("CustomerID",IntegerType(),True) \
        .add("Country",StringType(),True)

kaggledata = spark.read.format("csv") \
          .option("header",True) \
          .schema(ex5_schema) \
          .load("FileStore/tables/data.csv")

kaggledata.printSchema()
kaggledata.show(5)
```

```
root
 |-- InvoiceNo: string (nullable = true)
 |-- StockCode: string (nullable = true)
 |-- Description: string (nullable = true)
 |-- Quantity: integer (nullable = true)
 |-- InvoiceDate: string (nullable = true)
 |-- UnitPrice: double (nullable = true)
 |-- CustomerID: integer (nullable = true)
 |-- Country: string (nullable = true)

+---------+---------+--------------------+--------+--------------+---------+--
--------+--------------+
|InvoiceNo|StockCode|         Description|Quantity|   InvoiceDate|UnitPrice|Cu
stomerID|       Country|
+---------+---------+--------------------+--------+--------------+---------+--
--------+--------------+
|   536365|   85123A|WHITE HANGING HEA...|       6|12/1/2010 8:26|     2.55|
17850|United Kingdom|
|   536365|    71053| WHITE METAL LANTERN|       6|12/1/2010 8:26|     3.39|
17850|United Kingdom|
|   536365|   84406B|CREAM CUPID HEART...|       8|12/1/2010 8:26|     2.75|
17850|United Kingdom|
|   536365|   84029G|KNITTED UNION FLA...|       6|12/1/2010 8:26|     3.39|
17850|United Kingdom|
|   536365|   84029E|RED WOOLLY HOTTIE...|       6|12/1/2010 8:26|     3.39|
17850|United Kingdom|
+---------+---------+--------------------+--------+--------------+---------+--
--------+--------------+
only showing top 5 rows
```

```
import pyspark
from pyspark.sql.functions import col

kaggledf_amount = kaggledata.withColumn("Amount_Spent", col("Quantity") *
col("UnitPrice"))
kaggledf_amount.show(5)
filtered_kaggle_df = kaggledf_amount.filter("Quantity > 0 and InvoiceNo not
like '%C'")
filtered_kaggle_df.show(5)
#kaggledf_amount = kaggledata.withColumn("Amount_Spent", col("Quantity") *
col("UnitPrice"))
#kaggledf_amount.show(5)
#kaggle_df_group =
filtered_kaggle_df.groupBy("Country","CustomerID").sum("Amount_Spent").withColu
mnRenamed("sum(Amount_spent)","Amount Spent")
#display(kaggle_df_group)
```

```
+---------+---------+--------------------+--------+--------------+---------+--
--------+--------------+------------------+
|InvoiceNo|StockCode|         Description|Quantity|   InvoiceDate|UnitPrice|Cu
stomerID|       Country|      Amount_Spent|
+---------+---------+--------------------+--------+--------------+---------+--
--------+--------------+------------------+
|   536365|   85123A|WHITE HANGING HEA...|       6|12/1/2010 8:26|     2.55|
17850|United Kingdom|15.299999999999999|
|   536365|    71053| WHITE METAL LANTERN|       6|12/1/2010 8:26|     3.39|
17850|United Kingdom|             20.34|
|   536365|   84406B|CREAM CUPID HEART...|       8|12/1/2010 8:26|     2.75|
17850|United Kingdom|              22.0|
|   536365|   84029G|KNITTED UNION FLA...|       6|12/1/2010 8:26|     3.39|
17850|United Kingdom|             20.34|
|   536365|   84029E|RED WOOLLY HOTTIE...|       6|12/1/2010 8:26|     3.39|
17850|United Kingdom|             20.34|
+---------+---------+--------------------+--------+--------------+---------+--
--------+--------------+------------------+
only showing top 5 rows
```

```
+---------+---------+---------------------+--------+---------------+---------+--
```

```
from pyspark.sql.functions import split
split_date = split(filtered_kaggle_df["InvoiceDate"]," ")

dfprojected =
filtered_kaggle_df.withColumn("Invoice_Date",split_date.getItem(0))
split_date_1 = split(dfprojected["Invoice_Date"],"/")
from pyspark.sql.functions import concat, lit
#Extracting Year and Month out
dfprojected1 =
dfprojected.withColumn("Month",split_date_1.getItem(0)).withColumn("Year",split
_date_1.getItem(2)).withColumn("Month-year",concat( col("Year"), lit("-"),
col("Month")))
#Extracting the Month out
dfprojected2 = dfprojected.withColumn("Month",split_date_1.getItem(0))
```

	StockCode	Month	Mean_Of_Amount_Totally_Priced
1	22688	9	493.31666666666666
2	22117	8	398.27500000000003
3	22827	10	290
4	22782	2	247
5	23134	7	238.494
6	48111	9	228.2290909090909
7	22655	3	187.5
8	22823	2	125

Showing the first 1000 rows.