Task-2

July 25, 2020

1 Task 2

2 Get only categories for shoes

Select all rows from products where the category name contains the string "shoes"

[1]:
```
from pyspark.sql import SparkSession
from pyspark.sql import Row

df = spark.read \
    .format("jdbc") \
    .option("url", "jdbc:postgresql://managed-pg:5432/products") \
    .option("driver","org.postgresql.Driver") \
    .option("dbtable", "(select * from temp_products where UPPER(category_name)␣
 ↪like UPPER('%shoes%')) as test") \
    .option("user", "postgres") \
    .option("password", "postgres") \
    .load()
```

2.1 Schema of selected DF

[2]:
```
df.printSchema()
```

```
root
 |-- product_id: integer (nullable = true)
 |-- name: string (nullable = true)
 |-- upc_id: string (nullable = true)
 |-- descr: string (nullable = true)
 |-- vendor_catalog_url: string (nullable = true)
 |-- buy_url: string (nullable = true)
 |-- manufacturer_name: string (nullable = true)
 |-- sale_price: decimal(38,18) (nullable = true)
 |-- retail_price: decimal(38,18) (nullable = true)
 |-- manufacturer_part_no: string (nullable = true)
 |-- country: string (nullable = true)
 |-- vendor_id: integer (nullable = true)
```

```
 |-- category_name: string (nullable = true)
 |-- category_code: string (nullable = true)
 |-- category_id: integer (nullable = true)
```

2.2 Save the selected rows in parquet

under hdfs://data/exercise/shoes.parquet

[3]:
```
df.write.mode("overwrite").parquet("hdfs:///data/exercise/shoes.parquet")
```