```
# Activate Spark in our Colab notebook.
import os
# Find the latest version of spark 3.2  from http://www.apache.org/dist/spark/ and enter as the spark version
# For example:
# spark_version = 'spark-3.2.2'
spark_version = 'spark-3.2.3'
os.environ['SPARK_VERSION']=spark_version

# Install Spark and Java
!apt-get update
!apt-get install openjdk-11-jdk-headless -qq > /dev/null
!wget -q http://www.apache.org/dist/spark/$SPARK_VERSION/$SPARK_VERSION-bin-hadoop2.7.tgz
!tar xf $SPARK_VERSION-bin-hadoop2.7.tgz
!pip install -q findspark

# Set Environment Variables
import os
os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-11-openjdk-amd64"
os.environ["SPARK_HOME"] = f"/content/{spark_version}-bin-hadoop2.7"

# Start a SparkSession
import findspark
findspark.init()
```

```
Hit:1 https://cloud.r-project.org/bin/linux/ubuntu focal-cran40/ InRelease
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:3 http://archive.ubuntu.com/ubuntu focal InRelease
Ign:4 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64  InRelease
Hit:5 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64  InRelease
Hit:6 http://ppa.launchpad.net/c2d4u.team/c2d4u4.0+/ubuntu focal InRelease
Hit:7 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64  Release
Hit:8 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Get:9 http://archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Hit:10 http://ppa.launchpad.net/cran/libgit2/ubuntu focal InRelease
Hit:12 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu focal InRelease
Hit:13 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu focal InRelease
Hit:14 http://ppa.launchpad.net/ubuntugis/ppa/ubuntu focal InRelease
Fetched 222 kB in 1s (214 kB/s)
Reading package lists... Done
```

```
# Get postgresql package
!wget https://jdbc.postgresql.org/download/postgresql-42.2.9.jar
```

```
--2023-03-10 23:03:01--  https://jdbc.postgresql.org/download/postgresql-42.2.9.jar
Resolving jdbc.postgresql.org (jdbc.postgresql.org)... 72.32.157.228, 2001:4800:3e1:1::228
Connecting to jdbc.postgresql.org (jdbc.postgresql.org)|72.32.157.228|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 914037 (893K) [application/java-archive]
Saving to: ‘postgresql-42.2.9.jar.3’

postgresql-42.2.9.j 100%[===================>] 892.61K  4.93MB/s    in 0.2s

2023-03-10 23:03:01 (4.93 MB/s) - ‘postgresql-42.2.9.jar.3’ saved [914037/914037]
```

```
# Import Spark and create a SparkSession
from pyspark.sql import SparkSession
spark = SparkSession.builder.appName("BigData-HW-1").config("spark.driver.extraClassPath","/content/postgresql-42.2.9.jar").getOrCreate()
```

Extract the Amazon Data into Spark DataFrame

```
# Read in the data from an S3 Bucket
from pyspark import SparkFiles
url = "https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon_reviews_us_Camera_v1_00.tsv.gz"

spark.sparkContext.addFile(url)
user_data_df = spark.read.csv(SparkFiles.get("amazon_reviews_us_Camera_v1_00.tsv.gz"), sep="\t", header=True, inferSchema=True)

# Show DataFrame
user_data_df.show()
```

```
+-----------+-----------+--------------+----------+--------------+--------------------+----------------+-----------+-------------+-----------+----+--------------
|marketplace|customer_id|     review_id|product_id|product_parent|       product_title|product_category|star_rating|helpful_votes|total_votes|vine|verified_purch
+-----------+-----------+--------------+----------+--------------+--------------------+----------------+-----------+-------------+-----------+----+--------------
|         US|    2975964|R1NBG94582SJE2|B00I01JQJM|     860486164|GoPro Rechargeabl...|          Camera|          5|            0|          0|   N|
|         US|   23526356|R273DCA6Y0H9V7|B00TCO0ZAA|     292641483|Professional 58mm...|          Camera|          5|            0|          0|   N|
|         US|   52764145| RQVOXO7WUOFK6|B00B7733E0|      75825744|Spy Tec Z12 Motio...|          Camera|          2|            1|          1|   N|
|         US|   47348933|R1KWKSF21PO6HO|B006ZN4U34|     789352955|Celestron UpClose...|          Camera|          5|            0|          0|   N|
|         US|   33680700|R38H3UO1J190GI|B00HUEBGMU|      19067902|Vidpro XM-L Wired...|          Camera|          5|            1|          1|   N|
|         US|   30301059|R3NPIFKLR19NQA|B008MW6Y12|     597683407|NIX 8 inch Hi-Res...|          Camera|          3|            0|          0|   N|
|         US|   28282645|R3MBE6UCH3435E|B00TE8XKIS|      35563334|Polaroid ZIP Mobi...|          Camera|          3|            8|          8|   N|
|         US|     502818|R2E7A4FF0PVY5Q|B00ZKDUFBQ|     555190742|GeekPro 2.0-Inch ...|          Camera|          5|            0|          1|   N|
|         US|    1481233| R3R8JDQ2BF4NM|B010BZ7S2Q|     129544315|Sony HDR-AZ1VR Ac...|          Camera|          5|            0|          2|   N|
|         US|   27885926|R1YND4BS823GN5|B00HRXSSRA|     708418657|ChiliPower DMW-BL...|          Camera|          1|            0|          0|   N|
|         US|    3183883|R2TZNSA18V7YF6|B005C95NM4|     246957815|Zeikos Deluxe Fla...|          Camera|          4|            1|          1|   N|
|         US|   23208852|R22ZVRDPPXIDNL|B00LBIBYBE|     746593019|  GoPro Hero Filters|          Camera|          5|            0|          0|   N|
|         US|   11438825|R1F4O6WOO2W461|B00X3HIM2U|     444991975|Neewer Meike MK-X...|          Camera|          3|            3|          3|   N|
|         US|   50399582| RT1KLS3QSJNUT|B00KDVQF8W|     304104050|LB Photography Ba...|          Camera|          5|            4|          4|   N|
|         US|   36700181|R222VYJL5K5IRS|B00GUZEZL4|     472875794|Waterproof Camera...|          Camera|          5|            0|          0|   N|
```

```
|         US|   47818374|R31LUR7M4PQOLU|B00FB1TBKS|     710827451|Ecolink Z-Wave PI...|         Camera|          5|            0|          0|   N|
|         US|   35272750|R1OD3T3Q042LUQ|B00GVMLPT6|     269896170|FotoTech Male to ...|         Camera|          2|            0|          0|   N|
|         US|   11736306|R2QT68OZTT2YKE|B00L8827BI|     145946775|Nikon D3200 Ultim...|         Camera|          5|            0|          0|   N|
|         US|   52377008| RMFQF59FG3TD1|B00SIM78R0|     972011051|D-Link Wireless D...|         Camera|          4|            0|          0|   N|
|         US|    6465510|R1QNYFW6G31R5T|B00EDCZKJ2|     258297575|ZINK Phone Photo ...|         Camera|          1|            1|          3|   N|
+-----------+-----------+--------------+----------+--------------+--------------------+---------------+-----------+-------------+-----------+----+-------------
only showing top 20 rows
```

```
user_data_df.printSchema()
```

```
root
 |-- marketplace: string (nullable = true)
 |-- customer_id: integer (nullable = true)
 |-- review_id: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- product_parent: integer (nullable = true)
 |-- product_title: string (nullable = true)
 |-- product_category: string (nullable = true)
 |-- star_rating: integer (nullable = true)
 |-- helpful_votes: integer (nullable = true)
 |-- total_votes: integer (nullable = true)
 |-- vine: string (nullable = true)
 |-- verified_purchase: string (nullable = true)
 |-- review_headline: string (nullable = true)
 |-- review_body: string (nullable = true)
 |-- review_date: string (nullable = true)
```

```
# Get the number of rows in the DataFrame.
user_data_df.count()
```

```
1801974
```

Transform the Data

Create the "review_id_table".

```
from pyspark.sql.functions import to_date
# Create the "review_id_df" DataFrame with the appropriate columns and data types.

# Select the columns needed for the review_id_table AND convert the review date to a date.
review_id_table = user_data_df.select(['review_id', 'customer_id', 'product_id', 'product_parent', to_date('review_date')])
review_id_table = review_id_table.withColumnRenamed("to_date(review_date)","review_date")
review_id_table.show()
```

```
+--------------+-----------+----------+--------------+-----------+
|     review_id|customer_id|product_id|product_parent|review_date|
+--------------+-----------+----------+--------------+-----------+
|R1NBG94582SJE2|    2975964|B00I01JQJM|     860486164| 2015-08-31|
|R273DCA6Y0H9V7|   23526356|B00TCO0ZAA|     292641483| 2015-08-31|
| RQVOXO7WUOFK6|   52764145|B00B7733E0|      75825744| 2015-08-31|
|R1KWKSF21PO6HO|   47348933|B006ZN4U34|     789352955| 2015-08-31|
|R38H3UO1J190GI|   33680700|B00HUEBGMU|      19067902| 2015-08-31|
|R3NPIFKLR19NQA|   30301059|B008MW6Y12|     597683407| 2015-08-31|
|R3MBE6UCH3435E|   28282645|B00TE8XKIS|      35563334| 2015-08-31|
|R2E7A4FF0PVY5Q|     502818|B00ZKDUFBQ|     555190742| 2015-08-31|
| R3R8JDQ2BF4NM|    1481233|B010BZ7S2Q|     129544315| 2015-08-31|
|R1YND4BS823GN5|   27885926|B00HRXSSRA|     708418657| 2015-08-31|
|R2TZNSA18V7YF6|    3183883|B005C95NM4|     246957815| 2015-08-31|
|R22ZVRDPPXIDNL|   23208852|B00LBIBYBE|     746593019| 2015-08-31|
|R1F4O6WOO2W461|   11438825|B00X3HIM2U|     444991975| 2015-08-31|
| RT1KLS3QSJNUT|   50399582|B00KDVQF8W|     304104050| 2015-08-31|
|R222VYJL5K5IRS|   36700181|B00GUZEZL4|     472875794| 2015-08-31|
|R31LUR7M4PQOLU|   47818374|B00FB1TBKS|     710827451| 2015-08-31|
|R1OD3T3Q042LUQ|   35272750|B00GVMLPT6|     269896170| 2015-08-31|
|R2QT68OZTT2YKE|   11736306|B00L8827BI|     145946775| 2015-08-31|
| RMFQF59FG3TD1|   52377008|B00SIM78R0|     972011051| 2015-08-31|
|R1QNYFW6G31R5T|    6465510|B00EDCZKJ2|     258297575| 2015-08-31|
+--------------+-----------+----------+--------------+-----------+
only showing top 20 rows
```

```
review_id_table = review_id_table.withColumnRenamed("to_date(review_date)","review_date")
review_id_table.show()
```

```
+--------------+-----------+----------+--------------+-----------+
|     review_id|customer_id|product_id|product_parent|review_date|
+--------------+-----------+----------+--------------+-----------+
|R1NBG94582SJE2|    2975964|B00I01JQJM|     860486164| 2015-08-31|
|R273DCA6Y0H9V7|   23526356|B00TCO0ZAA|     292641483| 2015-08-31|
| RQVOXO7WUOFK6|   52764145|B00B7733E0|      75825744| 2015-08-31|
|R1KWKSF21PO6HO|   47348933|B006ZN4U34|     789352955| 2015-08-31|
|R38H3UO1J190GI|   33680700|B00HUEBGMU|      19067902| 2015-08-31|
|R3NPIFKLR19NQA|   30301059|B008MW6Y12|     597683407| 2015-08-31|
|R3MBE6UCH3435E|   28282645|B00TE8XKIS|      35563334| 2015-08-31|
|R2E7A4FF0PVY5Q|     502818|B00ZKDUFBQ|     555190742| 2015-08-31|
```

```
| R3R8JDQ2BF4NM|    1481233|B010BZ7S2Q|     129544315| 2015-08-31|
|R1YND4BS823GN5|   27885926|B00HRXSSRA|     708418657| 2015-08-31|
|R2TZNSA18V7YF6|    3183883|B005C95NM4|     246957815| 2015-08-31|
|R22ZVRDPPXIDNL|   23208852|B00LBIBYBE|     746593019| 2015-08-31|
|R1F4O6WOO2W461|   11438825|B00X3HIM2U|     444991975| 2015-08-31|
| RT1KLS3QSJNUT|   50399582|B00KDVQF8W|     304104050| 2015-08-31|
|R222VYJL5K5IRS|   36700181|B00GUZEZL4|     472875794| 2015-08-31|
|R31LUR7M4PQOLU|   47818374|B00FB1TBKS|     710827451| 2015-08-31|
|R1OD3T3Q042LUQ|   35272750|B00GVMLPT6|     269896170| 2015-08-31|
|R2QT68OZTT2YKE|   11736306|B00L8827BI|     145946775| 2015-08-31|
| RMFQF59FG3TD1|   52377008|B00SIM78R0|     972011051| 2015-08-31|
|R1QNYFW6G31R5T|    6465510|B00EDCZKJ2|     258297575| 2015-08-31|
+--------------+-----------+----------+--------------+-----------+
only showing top 20 rows
```

Create the "products" Table

```
# Create the "products_df" DataFrame that drops the duplicates in the "product_id" and "product_title columns.
products_df = user_data_df.select(['product_id', 'product_title'])
products_df = products_df.dropDuplicates(['product_id'])
```

```
products_df.show()
```

```
+----------+--------------------+
|product_id|       product_title|
+----------+--------------------+
|0011300000|Genuine Geovision...|
|0974096512|DVD: Digital Phot...|
|0984445145|Tamara Lackey Cap...|
|0984445161|"Tamara Lackey's ...|
|0984920242|Samys Camera The ...|
|1123000034|NB-1LH NB-1L BATT...|
|1210300001|Orange Sources 12...|
|1379178304|Dengpin MC-DC2 Re...|
|1921453575|Pebble 8+2 Iridol...|
|198347598X|7 Colors Filters ...|
|3100028120|3 Years Warranty ...|
|3490204816|Orange Sources 3M...|
|5135000011|NP-BG1 NP-FG1 Dig...|
|6000005822|NB-2L Charger For...|
|6000006179|NB-6L Charger For...|
|6000006853|Best NP-40 NP-40N...|
|6000007388|KLIC-7004 Battery...|
|6000008775|NP-BK1 NPBK1 Type...|
|6000011474|EN-EL12 1500mAh B...|
|6000013159|2X EN-EL5 ENEL5 C...|
+----------+--------------------+
only showing top 20 rows
```

Create the "customers" Table

```
# Create the "customers_df" DataFrame that groups the data on the "customer_id" by the number of times a customer reviewed a product.
customers_df = user_data_df.groupby("customer_id").count()
customers_df = customers_df.withColumnRenamed("count","customer_count")
customers_df.show()
```

```
+-----------+--------------+
|customer_id|customer_count|
+-----------+--------------+
|   52695798|             1|
|   48363612|             5|
|   46909180|             1|
|   45595220|             1|
|   50372387|             2|
|    9731896|             1|
|   24540309|             1|
|    2019000|             1|
|   50798385|             5|
|   37669883|             1|
|   19718301|             1|
|   45616772|             1|
|    2167730|             1|
|   47027968|             7|
|    5459822|             1|
|   52484883|             1|
|   12425248|             1|
|   15460750|             1|
|   46944960|             1|
|   37502310|             2|
+-----------+--------------+
only showing top 20 rows
```

Create the "vine_table".

```
# Create the "vine_df" DataFrame that has the "review_id", "star_rating", "helpful_votes", "total_votes", and "vine" columns.
```

```
# Select the columns needed for the products_df
vine_df = user_data_df.select(['review_id', 'star_rating', 'helpful_votes', 'total_votes', 'vine'] )
vine_df.show(10)
```

```
+--------------+-----------+-------------+-----------+----+
|     review_id|star_rating|helpful_votes|total_votes|vine|
+--------------+-----------+-------------+-----------+----+
|R1NBG94582SJE2|          5|            0|          0|   N|
|R273DCA6Y0H9V7|          5|            0|          0|   N|
| RQVOXO7WUOFK6|          2|            1|          1|   N|
|R1KWKSF21PO6HO|          5|            0|          0|   N|
|R38H3UO1J190GI|          5|            1|          1|   N|
|R3NPIFKLR19NQA|          3|            0|          0|   N|
|R3MBE6UCH3435E|          3|            8|          8|   N|
|R2E7A4FF0PVY5Q|          5|            0|          1|   N|
| R3R8JDQ2BF4NM|          5|            0|          2|   N|
|R1YND4BS823GN5|          1|            0|          0|   N|
+--------------+-----------+-------------+-----------+----+
only showing top 10 rows
```

Load

```
# Configure settings for RDS
mode = "append"
jdbc_url="jdbc:postgresql://mypostgresdb.cjrjjt1yssrb.us-east-2.rds.amazonaws.com:5432/amazon_review_db"
config = {"user":"root",
          "password": "jsp-challenge-22",
          "driver":"org.postgresql.Driver"}
```

```
# Write review_id_df to table in RDS (this is AWS)
review_id_table.write.jdbc(url=jdbc_url, table='review_id_table', mode=mode, properties=config)
```

```
# Write products_df to table in RDS
products_df.write.jdbc(url=jdbc_url, table='products', mode=mode, properties=config)
```

```
# Write customers_df to table in RDS
customers_df.write.jdbc(url=jdbc_url, table='customers', mode=mode, properties=config)
```

```
# Write vine_df to table in RDS
vine_df.write.jdbc(url=jdbc_url, table='vine_table', mode=mode, properties=config)
```