```
# 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()
```

```
Get:1 https://cloud.r-project.org/bin/linux/ubuntu focal-cran40/ InRelease [3,622 B]
Hit:2 http://ppa.launchpad.net/c2d4u.team/c2d4u4.0+/ubuntu focal InRelease
Hit:3 http://archive.ubuntu.com/ubuntu focal InRelease
Ign:4 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64  InRelease
Get:5 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64  InRelease [1,581 B]
Hit:6 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu2004/x86_64  Release
Get:7 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:8 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:9 http://ppa.launchpad.net/cran/libgit2/ubuntu focal InRelease
Hit:10 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu focal InRelease
Get:11 http://archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Hit:12 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu focal InRelease
Hit:13 http://ppa.launchpad.net/ubuntugis/ppa/ubuntu focal InRelease
Get:14 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64  Packages [908 kB]
Get:16 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [1,015 kB]
Get:17 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [3,014 kB]
Get:18 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [1,311 kB]
Fetched 6,590 kB in 6s (1,149 kB/s)
Reading package lists... Done
```

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

```
--2023-03-10 23:49:42--  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’

postgresql-42.2.9.j 100%[===================>] 892.61K  5.76MB/s    in 0.2s

2023-03-10 23:49:43 (5.76 MB/s) - ‘postgresql-42.2.9.jar’ 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_Tools_v1_00.tsv.gz"

spark.sparkContext.addFile(url)
user_data_df = spark.read.csv(SparkFiles.get("amazon_reviews_us_Tools_v1_00.tsv.gz"), sep="\t", header=True, inferSchema=True)

# Show DataFrame
user_data_df.show(20)
```

```
+-----------+-----------+--------------+----------+--------------+--------------------+----------------+-----------+-------------+-----------+----+-------------
|marketplace|customer_id|     review_id|product_id|product_parent|       product_title|product_category|star_rating|helpful_votes|total_votes|vine|verified_purch
+-----------+-----------+--------------+----------+--------------+--------------------+----------------+-----------+-------------+-----------+----+-------------
|         US|   15785389|R2UM5QMHBHC90Q|B00H5U9ZD6|     115362950|WallPeg 12 sq ft ...|           Tools|          5|            0|          0|   N|
|         US|   47910848|  RF0D1LEIF6L7|B001TJGCS0|     570955425|Nite Ize Nite Daw...|           Tools|          4|            0|          0|   N|
|         US|   36328996|  RM6YKIWQVNSY|B000NIK8JW|     128843593|Stanley 84-058 4 ...|           Tools|          1|            6|          6|   N|
|         US|   51785809|R1RL3L68ASPS36|B008ZYRGUA|     407828107|Powerextra 14.4V ...|           Tools|          4|            0|          0|   N|
|         US|   40757491|R1U4XFBFAG34CY|B00K5CA0GC|     490746675|Waterproof Invisi...|           Tools|          5|            0|          0|   N|
|         US|   35544833|R3KFIK8P0I91PL|B00AIJAA94|     148352067|Crime Scene Do No...|           Tools|          5|            0|          0|   N|
|         US|   16474909| RENOAY76PPK1O|B00JKEI6K8|     331801084|Aweek® 2 Pcs Bicy...|           Tools|          5|            0|          0|   N|
|         US|   22601598| RINV884I0NL5V|B00AGCHVS6|     471514859|Ryobi P102 Genuin...|           Tools|          1|            0|          0|   N|
|         US|   16129808| R5KJH6CXZH2PX|B002S0O7U4|     162253576|Wiha 66995 6-Piec...|           Tools|          5|            0|          0|   N|
|         US|   24382335| RO69JF6QWD0W1|B0084YHXMW|      69530650|TOMTOP LED Submar...|           Tools|          5|            0|          0|   N|
|         US|   49796324|R3L9NQBH3FI55C|B00MLSS1SW|     916693555|Black & Decker BD...|           Tools|          4|            1|          1|   N|
```

```
|         US|   33289687| R4YH95YPHVU0C|B00D4WLS2A|      39333316|Crain 126 Staple ...|          Tools|          5|            0|          0|   N|
|         US|   10916386|R10M1WWDDQBGG2|B00JGCDV5Y|     550596607|Diamond Semi Roun...|          Tools|          2|            0|          0|   N|
|         US|   34071500| RV3KWQBTNIO62|B00N0PS3YM|     735538025|It Mall 9 LED 375...|          Tools|          5|            1|          1|   N|
|         US|   50594486|R1M7YUNLZI0G9F|B0000DD4KV|     506501960|IRWIN Tools Metri...|          Tools|          5|            1|          1|   N|
|         US|   21945887|R2MTL2D4E4HEF4|B0009H5FB8|     268586246|743022-A Backing ...|          Tools|          5|            0|          0|   N|
|         US|   47749608| RXAHWIC1584UQ|B00NKSMPZW|     824618679|ClearArmor 141001...|          Tools|          5|           20|         23|   N|
|         US|   48880662| RMOIQFERVQDWS|B00RBAB92K|     156791442|KKmoon 9cm Mini A...|          Tools|          5|            0|          0|   N|
|         US|    4660265| R710G45MKODY9|B00QGBNZVI|     962324810|Refun E6 High Pow...|          Tools|          2|            1|          1|   N|
|         US|   18397238|R23ZIUGUM7TBMY|B00XXU3CDG|     543062309|Dr.meter S20 Mois...|          Tools|          1|            0|          1|   N|
+-----------+-----------+--------------+----------+--------------+--------------------+---------------+-----------+-------------+-----------+----+--------------
only showing top 20 rows
```

```
# Tak a look at the schema
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: string (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()
```

```
1741100
```

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|
+--------------+-----------+----------+--------------+-----------+
|R2UM5QMHBHC90Q|   15785389|B00H5U9ZD6|     115362950| 2015-08-31|
|  RF0D1LEIF6L7|   47910848|B001TJGCS0|     570955425| 2015-08-31|
|  RM6YKIWQVNSY|   36328996|B000NIK8JW|     128843593| 2015-08-31|
|R1RL3L68ASPS36|   51785809|B008ZYRGUA|     407828107| 2015-08-31|
|R1U4XFBFAG34CY|   40757491|B00K5CA0GC|     490746675| 2015-08-31|
|R3KFIK8P0I91PL|   35544833|B00AIJAA94|     148352067| 2015-08-31|
| RENOAY76PPK1O|   16474909|B00JKEI6K8|     331801084| 2015-08-31|
| RINV884I0NL5V|   22601598|B00AGCHVS6|     471514859| 2015-08-31|
| R5KJH6CXZH2PX|   16129808|B002S0O7U4|     162253576| 2015-08-31|
| RO69JF6QWD0W1|   24382335|B0084YHXMW|      69530650| 2015-08-31|
|R3L9NQBH3FI55C|   49796324|B00MLSS1SW|     916693555| 2015-08-31|
| R4YH95YPHVU0C|   33289687|B00D4WLS2A|      39333316| 2015-08-31|
|R10M1WWDDQBGG2|   10916386|B00JGCDV5Y|     550596607| 2015-08-31|
| RV3KWQBTNIO62|   34071500|B00N0PS3YM|     735538025| 2015-08-31|
|R1M7YUNLZI0G9F|   50594486|B0000DD4KV|     506501960| 2015-08-31|
|R2MTL2D4E4HEF4|   21945887|B0009H5FB8|     268586246| 2015-08-31|
| RXAHWIC1584UQ|   47749608|B00NKSMPZW|     824618679| 2015-08-31|
| RMOIQFERVQDWS|   48880662|B00RBAB92K|     156791442| 2015-08-31|
| R710G45MKODY9|    4660265|B00QGBNZVI|     962324810| 2015-08-31|
|R23ZIUGUM7TBMY|   18397238|B00XXU3CDG|     543062309| 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|
+----------+--------------------+
|0258231246|Himalaya Shuddha ...|
|0328305030|Illinois Industri...|
|057802697X|Build A Sculpture...|
|0578060604|Build A Maloof In...|
|0615247881|Pen Turning with ...|
|0645230227|2 X Vicco Narayan...|
|0829164383|Himalaya Manjisht...|
|0958057133|Fastener Black Bo...|
|0970704615|Build Your Own Lo...|
|1012151026|Dabur Pure Indian...|
|1036987434|Divya Kesh Tail (...|
|111556000X|Divya Dant Kanti ...|
|112233446X|Dabur Amla Gold H...|
|1304757439|Dentist Office Pa...|
|1456987682|Maybelline the Co...|
|1465799281|Sesa Oil (For Lon...|
|1465799796|Confido Tablets P...|
|1558706879|The Pocket Hole D...|
|1582095744|Stainless Steel 1...|
|160085947X|2012 Taunton Fine...|
+----------+--------------------+
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|
+-----------+--------------+
|   45978717|             2|
|   43622307|             1|
|     740678|             4|
|   29045703|             1|
|   52484548|             3|
|   42560427|             4|
|   17067926|             1|
|   10093406|             1|
|   44979559|             1|
|   19432125|             1|
|   26079415|             1|
|   29931671|             1|
|   12945150|             3|
|   12036434|             1|
|   14230926|             1|
|   45015535|             1|
|   39064792|             1|
|   20709090|             1|
|   45074906|             1|
|   26767269|             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 = vine_df.withColumn("star_rating",vine_df["star_rating"].cast('integer'))
vine_df.show(10)
```

```
+--------------+-----------+-------------+-----------+----+
|     review_id|star_rating|helpful_votes|total_votes|vine|
+--------------+-----------+-------------+-----------+----+
|R2UM5QMHBHC90Q|          5|            0|          0|   N|
|  RF0D1LEIF6L7|          4|            0|          0|   N|
|  RM6YKIWQVNSY|          1|            6|          6|   N|
|R1RL3L68ASPS36|          4|            0|          0|   N|
|R1U4XFBFAG34CY|          5|            0|          0|   N|
|R3KFIK8P0I91PL|          5|            0|          0|   N|
| RENOAY76PPK1O|          5|            0|          0|   N|
| RINV884I0NL5V|          1|            0|          0|   N|
| R5KJH6CXZH2PX|          5|            0|          0|   N|
| RO69JF6QWD0W1|          5|            0|          0|   N|
+--------------+-----------+-------------+-----------+----+
only showing top 10 rows
```

Load

```
# Configure settings for RDS
mode = "append"
jdbc_url="jdbc:postgresql://[redacted]:5432/amazon_review_db"
config = {"user":"root",
          "password": "[redacted]",
          "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)
```

✓ 4m 17s completed at 7:16 PM