```
import os
# Find the latest version of spark 2.0  from http://www-us.apache.org/dist/spark/ and
# For example:
# spark_version = 'spark-3.0.0'
spark_version = 'spark-3.1.2'
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-us.apache.org/dist/spark/$SPARK_VERSION/$SPARK_VERSION-bin-hadoop2
!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 http://ppa.launchpad.net/c2d4u.team/c2d4u4.0+/ubuntu bionic InRelease
Hit:2 http://archive.ubuntu.com/ubuntu bionic InRelease
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Hit:4 http://ppa.launchpad.net/cran/libgit2/ubuntu bionic InRelease
Hit:5 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran40/ InRelease
Get:6 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu bionic InRelease [15.9 kB]
Get:7 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Hit:8 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu bionic InRelease
Get:9 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Ign:10 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_6
Ign:11 https://developer.download.nvidia.com/compute/machine-learning/repos/ubun
Get:12 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_6
Hit:13 https://developer.download.nvidia.com/compute/machine-learning/repos/ubun
Get:14 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_6
Get:15 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [2,61
Get:16 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [
Get:17 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu bionic/main amd64 Packages
Ign:19 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_6
Get:19 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_6
Get:20 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [2,
Get:21 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages
Fetched 9,312 kB in 3s (3,448 kB/s)
Reading package lists... Done
```

```
# Download the Postgres driver that will allow Spark to interact with Postgres.
!wget https://jdbc.postgresql.org/download/postgresql-42.2.16.jar
```

```
--2021-06-20 22:11:37--  https://jdbc.postgresql.org/download/postgresql-42.2.16
```

```
Resolving jdbc.postgresql.org (jdbc.postgresql.org)... 72.32.157.228, 2001:4800:
Connecting to jdbc.postgresql.org (jdbc.postgresql.org)|72.32.157.228|:443... co
HTTP request sent, awaiting response... 200 OK
Length: 1002883 (979K) [application/java-archive]
Saving to: ‘postgresql-42.2.16.jar’

postgresql-42.2.16. 100%[===================>] 979.38K  1.68MB/s    in 0.6s

2021-06-20 22:11:39 (1.68 MB/s) - ‘postgresql-42.2.16.jar’ saved [1002883/100288
```

```
from pyspark.sql import SparkSession
spark = SparkSession.builder.appName("BigData-Challenge").config("spark.driver.extraC
```

Load Amazon Data into Spark DataFrame

```
from pyspark import SparkFiles
url = "https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon_reviews_us_Wireless_v1_
spark.sparkContext.addFile(url)
df = spark.read.option("encoding", "UTF-8").csv(SparkFiles.get(""), sep="\t", header=
df.show()
```

```
+-----------+-----------+--------------+----------+--------------+--------------
|marketplace|customer_id|     review_id|product_id|product_parent|       product_
+-----------+-----------+--------------+----------+--------------+--------------
|         US|   16414143|R3W4P9UBGNGH1U|B00YL0EKWE|     852431543|LG G4 Case Har
|         US|   50800750|R15V54KBMTQWAY|B00XK95RPQ|     516894650|Selfie Stick F
|         US|   15184378| RY8I449HNXSVF|B00SXRXUKO|     984297154|Tribe AB40 Wat
|         US|   10203548|R18TLJYCKJFLSR|B009V5X1CE|     279912704|RAVPower® Elem
|         US|     488280|R1NK26SWS53B8Q|B00D93OVF0|     662791300|Fosmon Micro U
|         US|   13334021|R11LOHEDYJALTN|B00XVGJMDQ|     421688488|iPhone 6 Case,
|         US|   27520697|R3ALQVQB2P9LA7|B00KQW1X1C|     554285554|Nokia Lumia 63
|         US|   48086021|R3MWLXLNO21PDQ|B00IP1MQNK|     488006702|Lumsing 10400m
|         US|   12738196|R2L15IS24CX0LI|B00HVORET8|     389677711|iPhone 5S Batt
|         US|   15867807|R1DJ8976WPWVZU|B00HX3G6J6|     299654876|HTC One M8 Scr
|         US|    1972249|R3MRWNNR8CBTB7|B00U4NATNQ|     577878727|S6 Case - Bear
|         US|   10956619|R1DS6DKTUXAQK3|B00SZEFDH8|     654620704|BLU Studio X, 
|         US|   14805911| RWJM5E0TWUJD2|B00JRJUL9U|     391166958|EZOPower 5-Por
|         US|   15611116|R1XTJKDYNCRGAC|B00KQ4T0HE|     481551630|iPhone 6S Case
|         US|   39298603|R2UZL3DPWEU1XW|B00M0YWKPM|     685107474| iPhone 6s Plu
|         US|   17552454|R2EZXET9KBFFU3|B00KDZEE68|     148320945|zBoost ZB575-A
|         US|   12218556|R26VY1L1FD3LPU|B00BJN45GM|      47788188|OtterBox Defen
|         US|   21872923|R2SSA4NSFCV18T|B00SA86SXW|     748759272|Aduro PowerUP 
|         US|   16264332|R1G6333JHJNEUQ|B00Q3I68TU|     974085141|LilGadgets Con
|         US|    6042304|R2DRG0UZXJQ0PE|B00TN4J1TA|     716174627|Anker Aluminum
+-----------+-----------+--------------+----------+--------------+--------------
only showing top 20 rows
```

Create DataFrames to match tables

```
from pyspark.sql.functions import to_date
# Read in the Review dataset as a DataFrame
df
```

```
DataFrame[marketplace: string, customer_id: int, review_id: string, product_id: s
```

```
# Create the customers_table DataFrame
customers_df = df.groupby("customer_id").agg({"customer_id":"count"}).withColumnRename
```

```
customers_df.show()
```

```
+-----------+--------------+
|customer_id|customer_count|
+-----------+--------------+
|   46909180|             6|
|   42560427|             7|
|   43789873|             3|
|   22037526|             2|
|   34220092|             2|
|   42801586|             1|
|    9565734|             2|
|   15829398|             1|
|   38247118|             1|
|   32478248|             2|
|   48114630|             1|
|   23085063|             1|
|   32787070|             3|
|   43515569|             1|
|    4919528|             2|
|    5088547|             2|
|   41852407|             3|
|   49703087|             1|
|   12713799|             1|
|   36728141|             8|
+-----------+--------------+
only showing top 20 rows
```

```
# Create the products_table DataFrame and drop duplicates.
products_df = df.select(["product_id", "product_title"]).drop_duplicates()
```

```
products_df.show()
```

```
+----------+--------------------+
|product_id|       product_title|
+----------+--------------------+
|B010LVPJH6|LG G Watch Urbane...|
|B00K5ZNXZ4|Minisuit Sporty A...|
|B00S9RBQOK|iPhone 6 Plus Cas...|
```

```
|B0116N7GYC|Selfie Stick, Por...|
|B009UNH0CY|Plantronics Voyag...|
|B00L8GFYAG|Eallc New Quality...|
|B00R3LMTI0|Kaleidio [Wallop ...|
|B011R0VG36|Galaxy Note 4 Cas...|
|B00BXX0QVQ|iKross Black Dual...|
|B00F4AYI2M|Incipio DualPro C...|
|B00GPI3OHC|Retevis H-777 2 W...|
|B00Y9ZUVU6|Tiwkich 2 in 1 Du...|
|B00W65SYHS|LG G4 case, Caseo...|
|B00V5OU6CW|S5 Leather case,P...|
|B00LP3FSH6|Escort Coiled Sma...|
|B00MIO2KRC|Black Box G1W-C C...|
|B00V5FZM0M|   KoKo Cases 5/5S !|
|B00T1KO2TA|iPhone 6 & 6S Cas...|
|B00YU9XOTQ|Galaxy S5 Screen ...|
|B00PI7IGHE|Soyan Latest DZ09...|
+----------+--------------------+
only showing top 20 rows
```

```
# Create the review_id_table DataFrame.
# Convert the 'review_date' column to a date datatype with to_date("review_date", 'yyy
review_id_df = df.select(['review_id', 'customer_id' , 'product_id', 'product_parent',
```

```
review_id_df.show()
```

```
+--------------+-----------+----------+--------------+-----------+
|     review_id|customer_id|product_id|product_parent|review_date|
+--------------+-----------+----------+--------------+-----------+
|R3W4P9UBGNGH1U|   16414143|B00YL0EKWE|     852431543| 2015-08-31|
|R15V54KBMTQWAY|   50800750|B00XK95RPQ|     516894650| 2015-08-31|
| RY8I449HNXSVF|   15184378|B00SXRXUKO|     984297154| 2015-08-31|
|R18TLJYCKJFLSR|   10203548|B009V5X1CE|     279912704| 2015-08-31|
|R1NK26SWS53B8Q|     488280|B00D93OVF0|     662791300| 2015-08-31|
|R11LOHEDYJALTN|   13334021|B00XVGJMDQ|     421688488| 2015-08-31|
|R3ALQVQB2P9LA7|   27520697|B00KQW1X1C|     554285554| 2015-08-31|
|R3MWLXLNO21PDQ|   48086021|B00IP1MQNK|     488006702| 2015-08-31|
|R2L15IS24CX0LI|   12738196|B00HVORET8|     389677711| 2015-08-31|
|R1DJ8976WPWVZU|   15867807|B00HX3G6J6|     299654876| 2015-08-31|
|R3MRWNNR8CBTB7|    1972249|B00U4NATNQ|     577878727| 2015-08-31|
|R1DS6DKTUXAQK3|   10956619|B00SZEFDH8|     654620704| 2015-08-31|
| RWJM5E0TWUJD2|   14805911|B00JRJUL9U|     391166958| 2015-08-31|
|R1XTJKDYNCRGAC|   15611116|B00KQ4T0HE|     481551630| 2015-08-31|
|R2UZL3DPWEU1XW|   39298603|B00M0YWKPM|     685107474| 2015-08-31|
|R2EZXET9KBFFU3|   17552454|B00KDZEE68|     148320945| 2015-08-31|
|R26VY1L1FD3LPU|   12218556|B00BJN45GM|      47788188| 2015-08-31|
|R2SSA4NSFCV18T|   21872923|B00SA86SXW|     748759272| 2015-08-31|
|R1G6333JHJNEUQ|   16264332|B00Q3I68TU|     974085141| 2015-08-31|
|R2DRG0UZXJQ0PE|    6042304|B00TN4J1TA|     716174627| 2015-08-31|
+--------------+-----------+----------+--------------+-----------+
only showing top 20 rows
```

```
# Create the wine table DataFrame
```

```
# Create the vine_table. DataFrame
vine_df = df.select(['review_id', 'star_rating', 'helpful_votes', 'total_votes', 'vine
```

```
vine_df.show()
```

```
+--------------+-----------+-------------+-----------+----+-----------------+
|     review_id|star_rating|helpful_votes|total_votes|vine|verified_purchase|
+--------------+-----------+-------------+-----------+----+-----------------+
|R3W4P9UBGNGH1U|          2|            1|          3|   N|                Y|
|R15V54KBMTQWAY|          4|            0|          0|   N|                N|
| RY8I449HNXSVF|          5|            0|          0|   N|                Y|
|R18TLJYCKJFLSR|          5|            0|          0|   N|                Y|
|R1NK26SWS53B8Q|          5|            0|          0|   N|                Y|
|R11LOHEDYJALTN|          5|            0|          0|   N|                Y|
|R3ALQVQB2P9LA7|          4|            0|          0|   N|                Y|
|R3MWLXLNO21PDQ|          5|            0|          0|   N|                Y|
|R2L15IS24CX0LI|          5|            0|          0|   N|                Y|
|R1DJ8976WPWVZU|          3|            0|          0|   N|                Y|
|R3MRWNNR8CBTB7|          5|            0|          0|   N|                Y|
|R1DS6DKTUXAQK3|          5|            0|          0|   N|                Y|
| RWJM5E0TWUJD2|          5|            0|          0|   N|                Y|
|R1XTJKDYNCRGAC|          1|            0|          0|   N|                Y|
|R2UZL3DPWEU1XW|          5|            0|          0|   N|                Y|
|R2EZXET9KBFFU3|          1|            0|          0|   N|                Y|
|R26VY1L1FD3LPU|          5|            0|          0|   N|                Y|
|R2SSA4NSFCV18T|          5|            0|          0|   N|                N|
|R1G6333JHJNEUQ|          5|            0|          0|   N|                Y|
|R2DRG0UZXJQ0PE|          5|            0|          0|   N|                Y|
+--------------+-----------+-------------+-----------+----+-----------------+
only showing top 20 rows
```

Connect to the AWS RDS instance and write each DataFrame to its table.

```
# Configure settings for RDS
mode = "append"
from getpass import getpass
password = getpass('Enter database password')
jdbc_url="jdbc:postgresql://dataviz.ch4kxtyqixht.us-east-2.rds.amazonaws.com:5432/post
config = {"user":"postgres",
          "password": password,
          "driver":"org.postgresql.Driver"}
```

```
Enter database password··********
```

```
# Write review_id_df to table in RDS
review_id_df.write.jdbc(url=jdbc_url, table='review_id_table', mode=mode, properties=c
```

```
# Write products_df to table in RDS
# about 3 min
products_df.write.jdbc(url=jdbc_url, table='products_table', mode=mode, properties=con
```

```
products_df.write.jdbc(url=jdbc_url, table='products_table', mode=mode, properties=co


# Write customers_df to table in RDS
# 5 min 14 s
customers_df.write.jdbc(url=jdbc_url, table='customers_table', mode=mode, properties=c


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

5m 14s completed at 7:31 PM