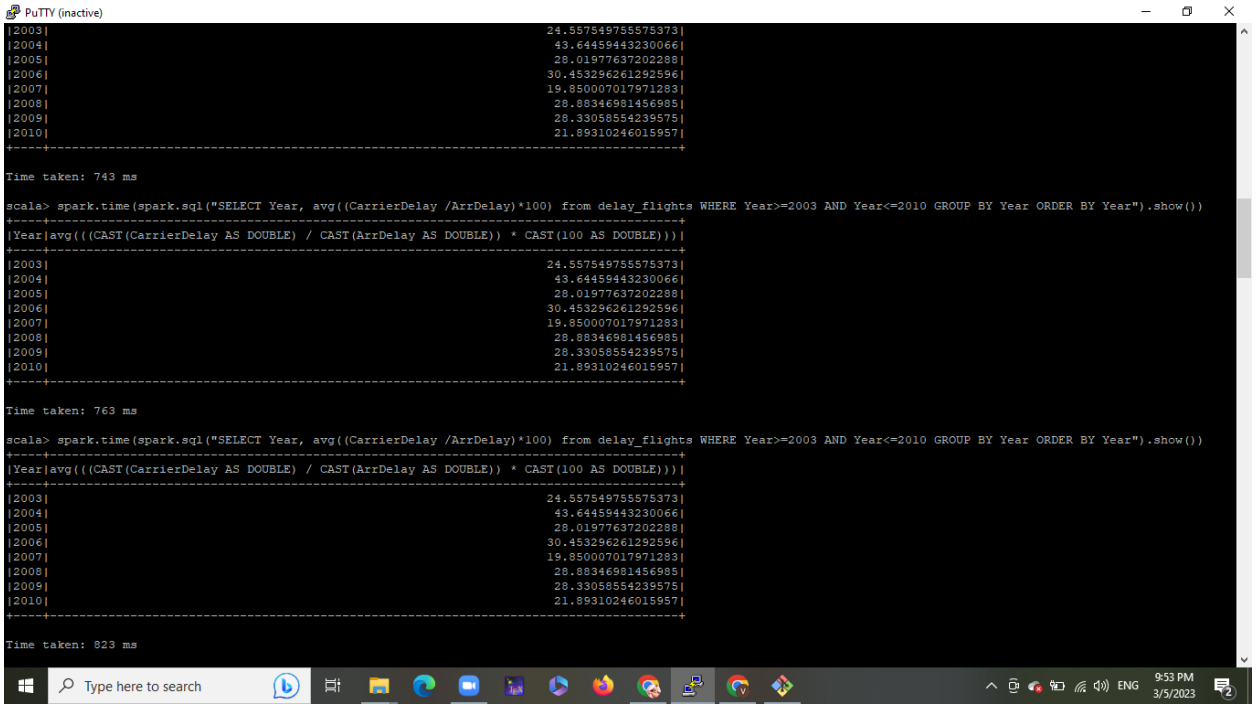
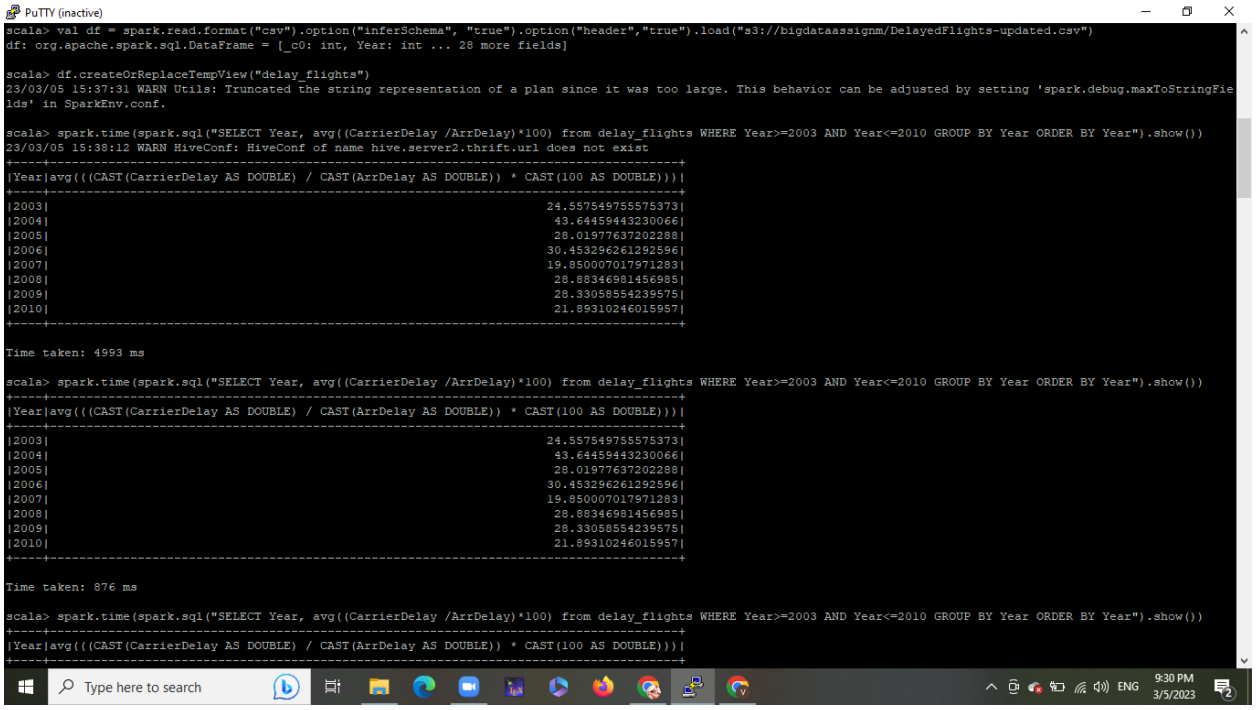
PuTTY (inactive)

```
EE::::EEEEEEEE:::E M:::::M            M:::::M    R:::R      R::::R
E::::::::::::::::::E M:::::M            M:::::M RR:::R      R::::R
EEEEEEEEEEEEEEEEEEE MMMMMMM            MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-77-33 ~]$ spark-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/03/05 15:36:26 WARN Client: Neither spark.yarn.jars nor spark.yarn.archive is set, falling back to uploading libraries under SPARK_HOME.
Spark context Web UI available at http://ip-172-31-77-33.ec2.internal:4040
Spark context available as 'sc' (master = yarn, app id = application_1678026367764_0004).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.4.8-amzn-2
      /_/

Using Scala version 2.11.12 (OpenJDK 64-Bit Server VM, Java 1.8.0_362)
Type in expressions to have them evaluated.
Type :help for more information.

scala> val df = spark.read.format("csv").option("inferSchema", "true").option("header","true").load("s3://bigdataassignm/DelayedFlights-updated.csv")
df: org.apache.spark.sql.DataFrame = [_c0: int, Year: int ... 28 more fields]

scala> df.createOrReplaceTempView("delay_flights")
23/03/05 15:37:31 WARN Utils: Truncated the string representation of a plan since it was too large. This behavior can be adjusted by setting 'spark.debug.maxToStringFie
lds' in SparkEnv.conf.

scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
23/03/05 15:38:12 WARN HiveConf: HiveConf of name hive.server2.thrift.url does not exist
+----+-----------------------------------------------------------------------------------+
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
+----+-----------------------------------------------------------------------------------+
|2003|                                                                 24.5575497555575373|
|2004|                                                                  43.64459443230066|
|2005|                                                                  28.01977637202288|
|2006|                                                                 30.453296261292596|
|2007|                                                                 19.850007017971283|
|2008|                                                                  28.88346981456985|
|2009|                                                                  28.33058554239575|
|2010|                                                                  21.89310246015957|
+----+-----------------------------------------------------------------------------------+
```

hadoop@ip-172-31-77-33:~

```
scala> val df = spark.read.format("csv").option("inferSchema", "true").option("header","true").load("s3://bigdataassignm/DelayedFlights-updated.csv")
df: org.apache.spark.sql.DataFrame = [_c0: int, Year: int ... 28 more fields]

scala> df.createOrReplaceTempView("delay_flights")
23/03/05 15:37:31 WARN Utils: Truncated the string representation of a plan since it was too large. This behavior can be adjusted by setting 'spark.debug.maxToStringFie
lds' in SparkEnv.conf.

scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
23/03/05 15:38:12 WARN HiveConf: HiveConf of name hive.server2.thrift.url does not exist
+----+-----------------------------------------------------------------------------------+
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
+----+-----------------------------------------------------------------------------------+
|2003|                                                                 24.5575497555575373|
|2004|                                                                  43.64459443230066|
|2005|                                                                  28.01977637202288|
|2006|                                                                 30.453296261292596|
|2007|                                                                 19.850007017971283|
|2008|                                                                  28.88346981456985|
|2009|                                                                  28.33058554239575|
|2010|                                                                  21.89310246015957|
+----+-----------------------------------------------------------------------------------+

Time taken: 4993 ms

scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
+----+-----------------------------------------------------------------------------------+
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
+----+-----------------------------------------------------------------------------------+
|2003|                                                                 24.5575497555575373|
|2004|                                                                  43.64459443230066|
|2005|                                                                  28.01977637202288|
|2006|                                                                 30.453296261292596|
|2007|                                                                 19.850007017971283|
|2008|                                                                  28.88346981456985|
|2009|                                                                  28.33058554239575|
|2010|                                                                  21.89310246015957|
+----+-----------------------------------------------------------------------------------+

Time taken: 876 ms

scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
+----+-----------------------------------------------------------------------------------+
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
+----+-----------------------------------------------------------------------------------+
```


PuTTY (inactive)
|2003| 24.5575497555575373|
|2004| 43.64459443230066|
|2005| 28.01977637202288|
|2006| 30.453296261292596|
|2007| 19.8500070179712831
|2008| 28.88346981456985|
|2009| 28.33058554239575|
|2010| 21.89310246015957|
Time taken: 743 ms
scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
|2003| 24.5575497555575373|
|2004| 43.64459443230066|
|2005| 28.01977637202288|
|2006| 30.453296261292596|
|2007| 19.8500070179712831
|2008| 28.88346981456985|
|2009| 28.33058554239575|
|2010| 21.89310246015957|
Time taken: 763 ms
scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
|2003| 24.5575497555575373|
|2004| 43.64459443230066|
|2005| 28.01977637202288|
|2006| 30.453296261292596|
|2007| 19.8500070179712831
|2008| 28.88346981456985|
|2009| 28.33058554239575|
|2010| 21.89310246015957|
Time taken: 823 ms
Type here to search
ENG
9:53 PM
3/5/2023


PuTTY (inactive)
scala> val df = spark.read.format("csv").option("inferSchema", "true").option("header","true").load("s3://bigdataassignm/DelayedFlights-updated.csv")
df: org.apache.spark.sql.DataFrame = [_c0: int, Year: int ... 28 more fields]
scala> df.createOrReplaceTempView("delay_flights")
23/03/05 15:37:31 WARN Utils: Truncated the string representation of a plan since it was too large. This behavior can be adjusted by setting 'spark.debug.maxToStringFie
lds' in SparkEnv.conf.
scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
23/03/05 15:38:12 WARN HiveConf: HiveConf of name hive.server2.thrift.url does not exist
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
|2003| 24.5575497555575373|
|2004| 43.64459443230066|
|2005| 28.01977637202288|
|2006| 30.453296261292596|
|2007| 19.8500070179712831
|2008| 28.88346981456985|
|2009| 28.33058554239575|
|2010| 21.89310246015957|
Time taken: 4993 ms
scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
|2003| 24.5575497555575373|
|2004| 43.64459443230066|
|2005| 28.01977637202288|
|2006| 30.453296261292596|
|2007| 19.8500070179712831
|2008| 28.88346981456985|
|2009| 28.33058554239575|
|2010| 21.89310246015957|
Time taken: 876 ms
scala> spark.time(spark.sql("SELECT Year, avg((CarrierDelay /ArrDelay)*100) from delay_flights WHERE Year>=2003 AND Year<=2010 GROUP BY Year ORDER BY Year").show())
|Year|avg(((CAST(CarrierDelay AS DOUBLE) / CAST(ArrDelay AS DOUBLE)) * CAST(100 AS DOUBLE)))|
Type here to search
ENG
9:30 PM
3/5/2023