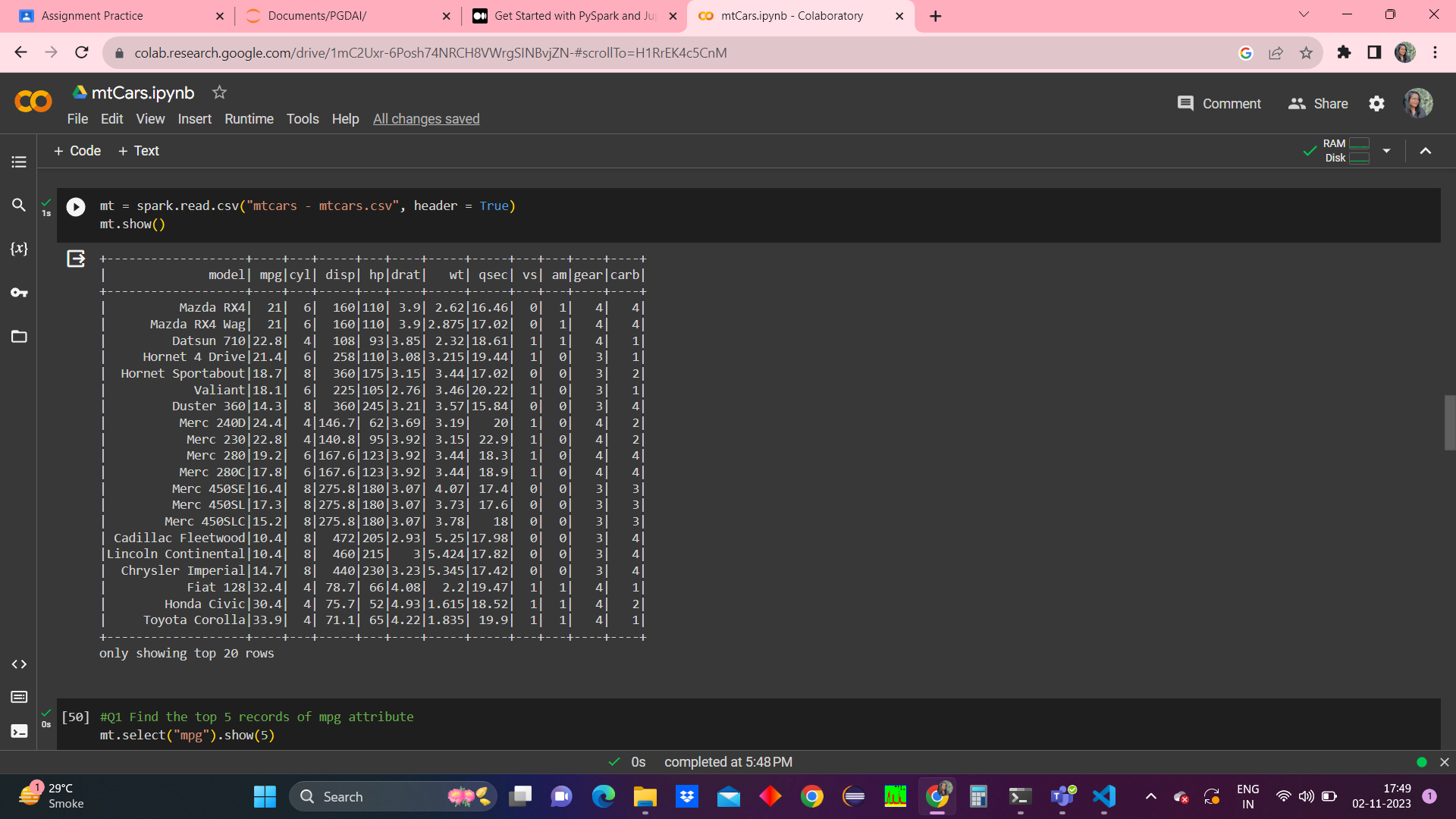
**mtCars**

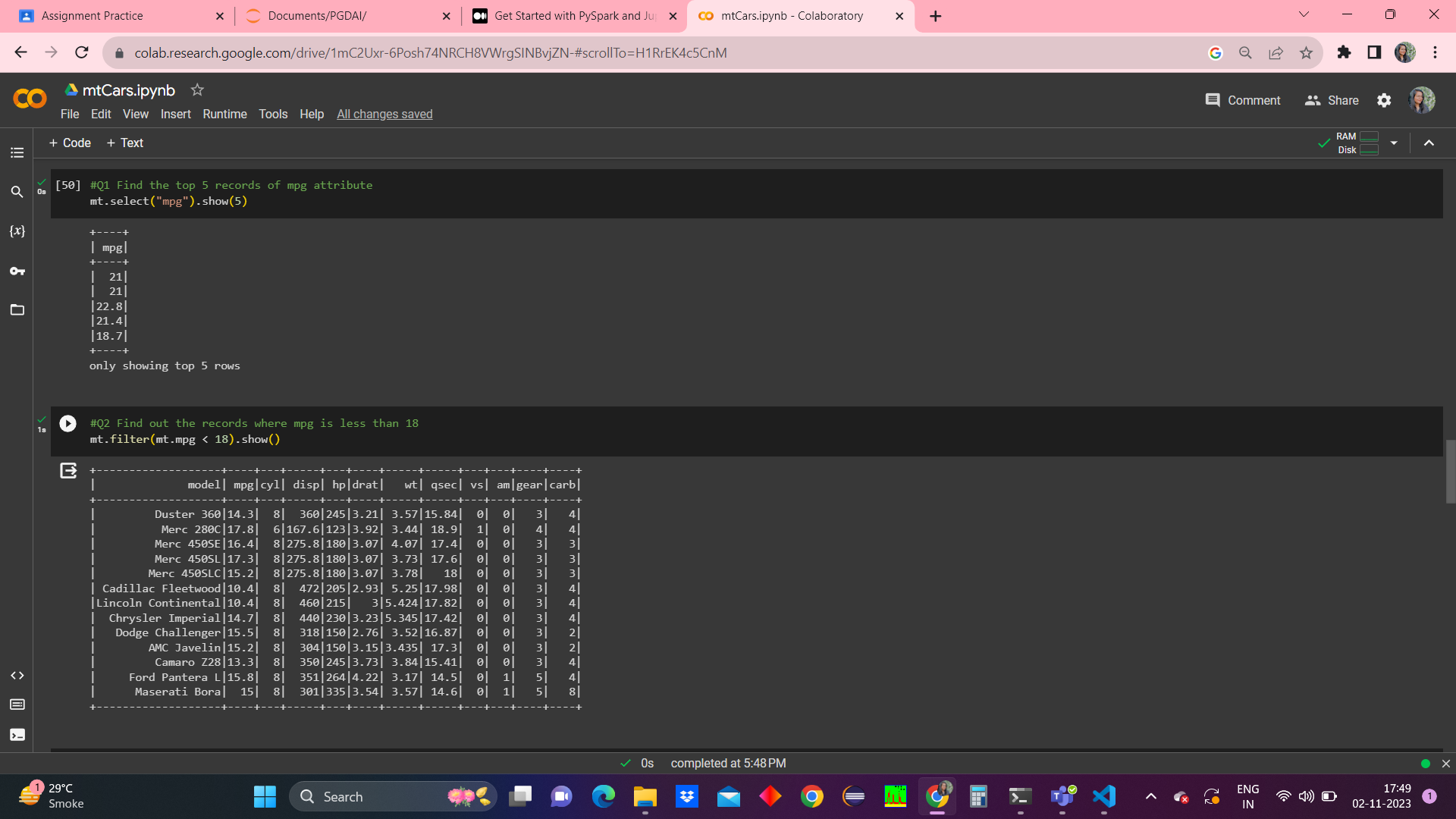
**Name – Ashi Jain**

**PGDAI**

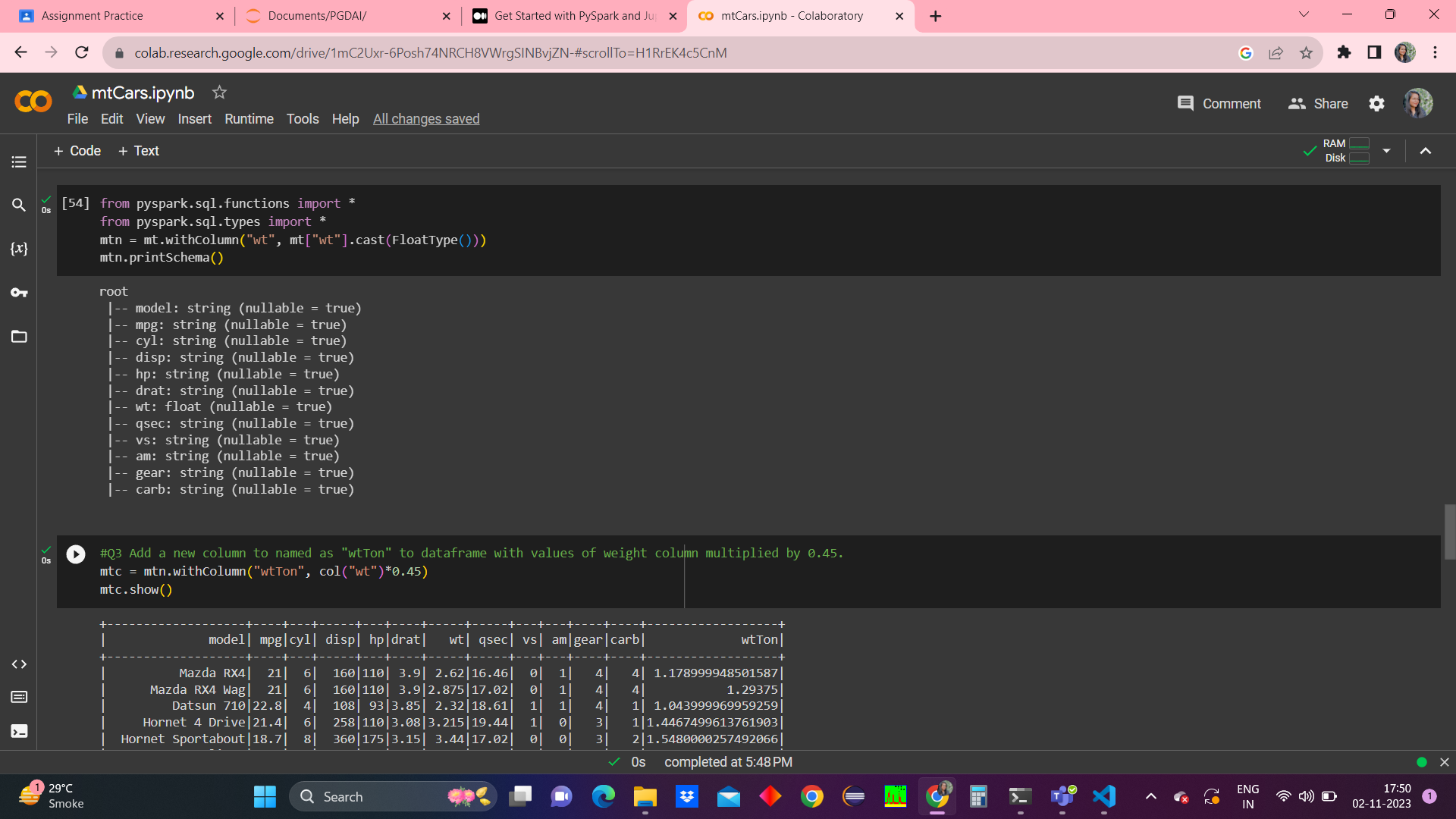
#Q1 Find the top five records of mpg attribute.

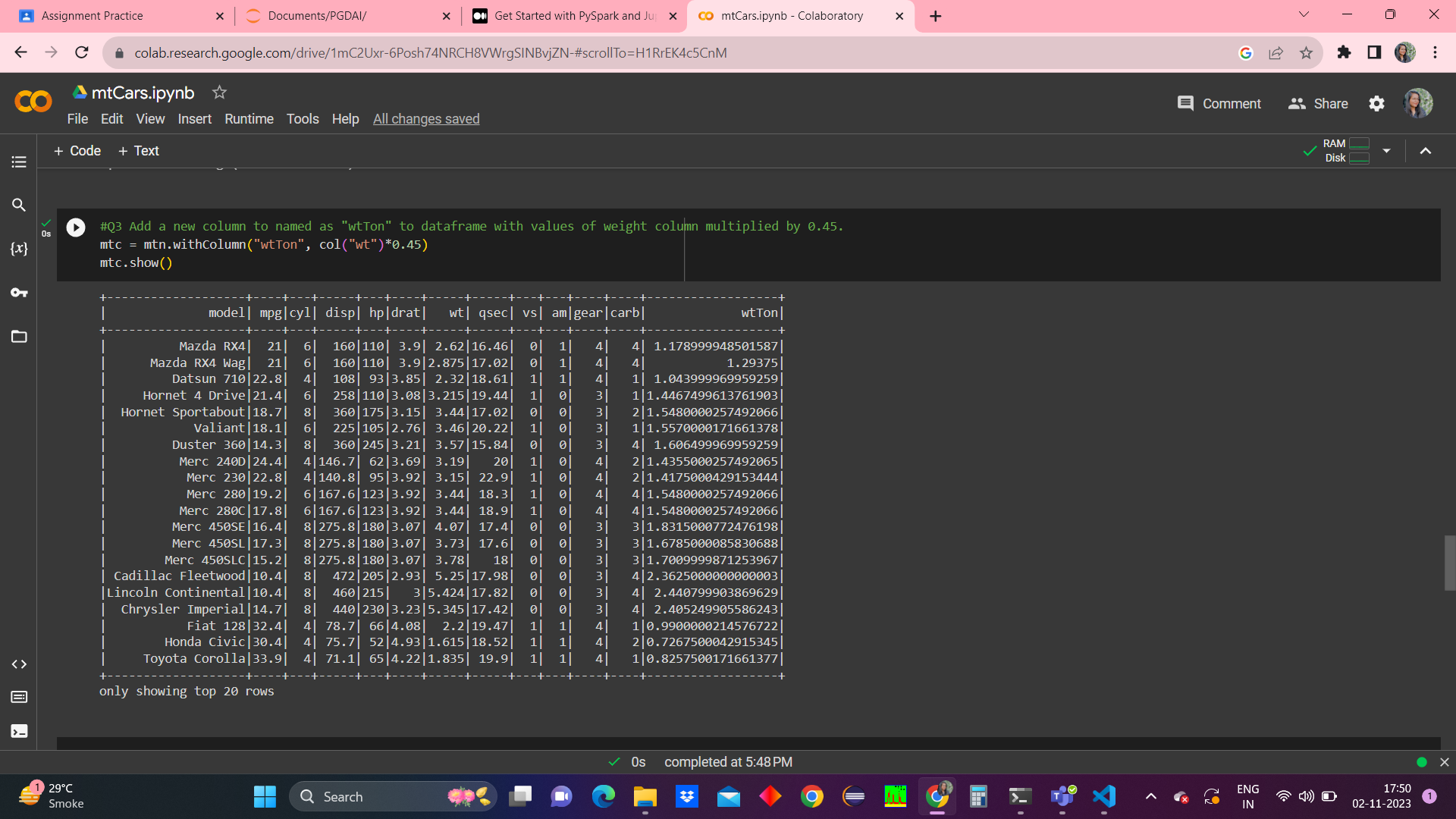


#Q2 Find out the records where the value of mpg is less than 18.

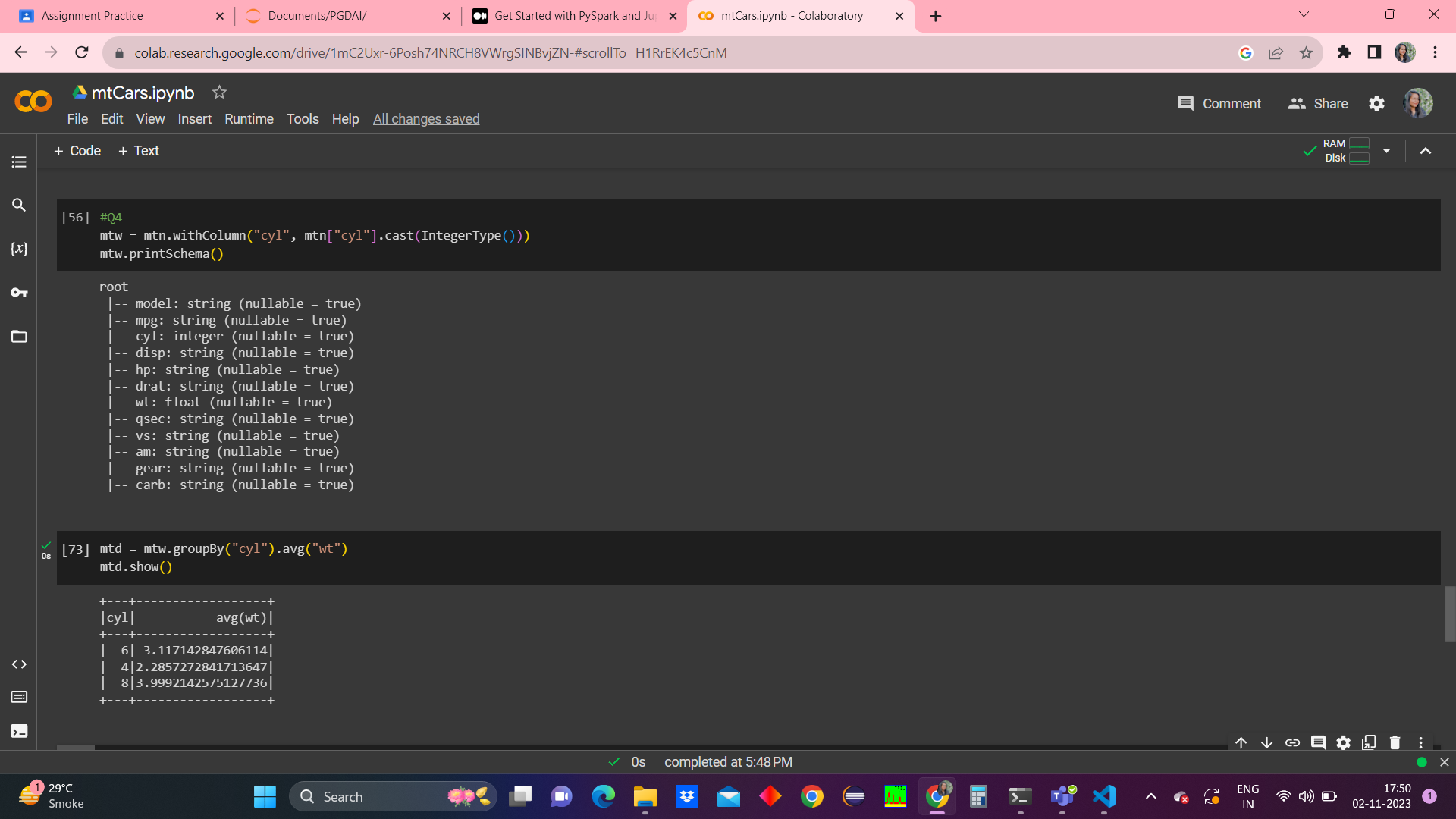


#Q3 Add a new column to named as "wtTon" to dataframe with values of weight column multiplied by 0.45.

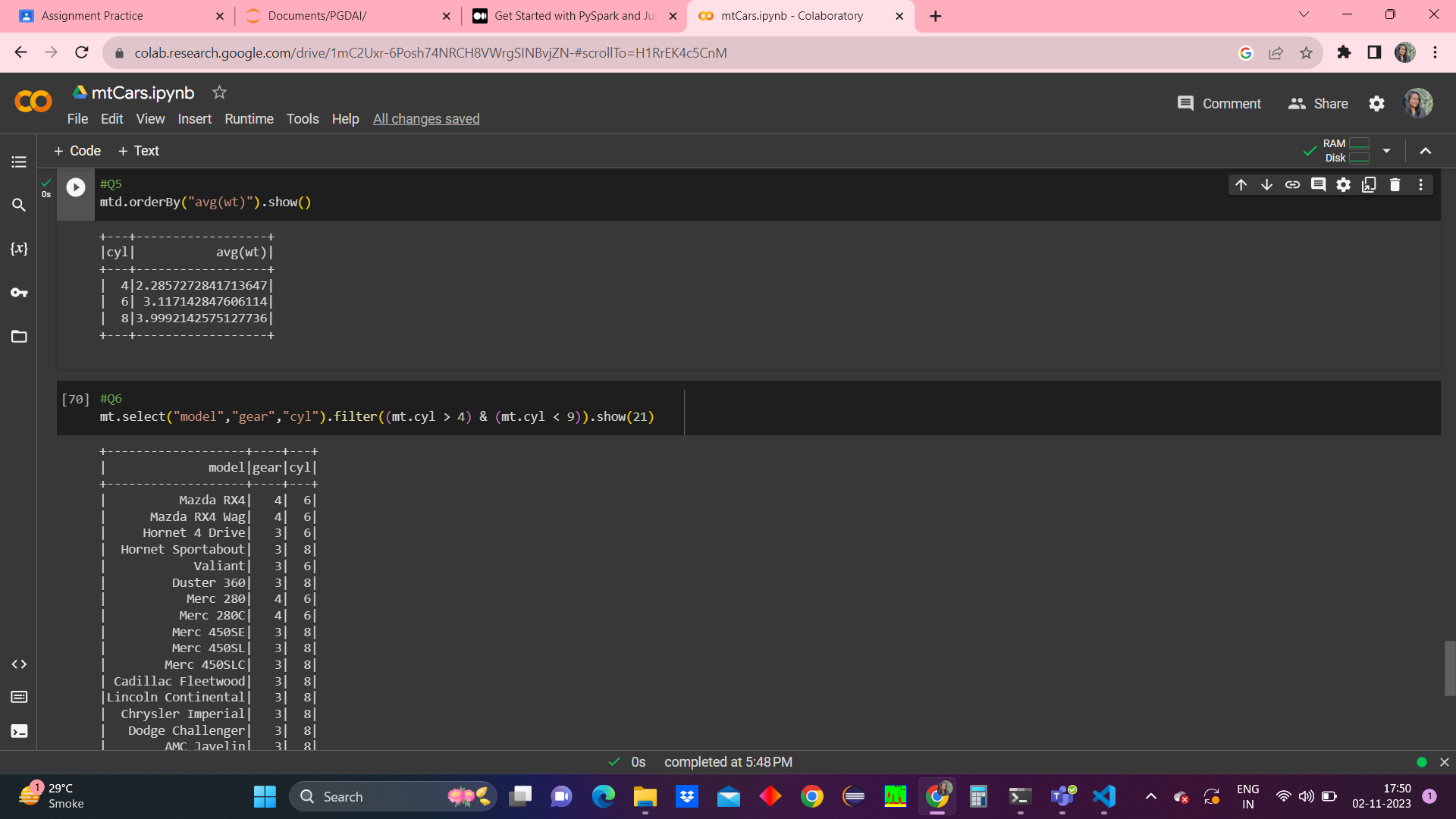




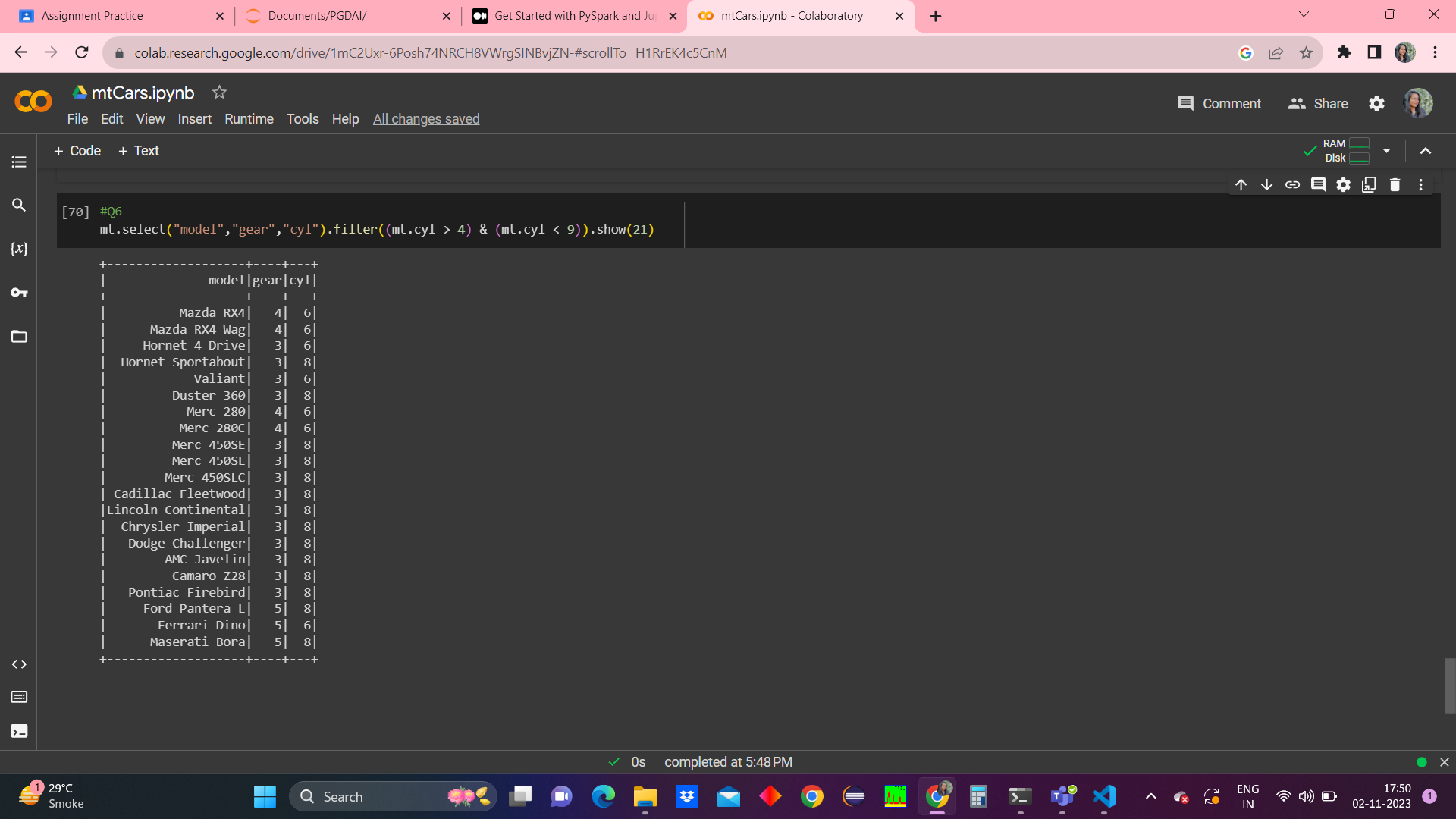
#Q4 Compute the average weight of cars by their cylinders.



#Q5 Also order the output of q.4



#Q6 Write a sql query to find gears where the value of cylinder lies between 4 and 9.



**Codes**

!pip install pyspark

!pip install -U -q PyDrive

!apt install openjdk-8-jdk-headless -qq

import os

os.environ["JAVA\_HOME"] = "/usr/lib/jvm/java-8-openjdk-amd64"

from pyspark.sql import SparkSession

from pyspark import SparkContext, SparkConf

# create the session

conf = SparkConf().set("spark.ui.port", "4050")

# create the context

import pyspark

sc = pyspark.SparkContext(conf=conf)

spark = SparkSession.builder.getOrCreate()

!wget https://bin.equinox.io/c/4VmDzA7iaHb/ngrok-stable-linux-amd64.zip

!unzip ngrok-stable-linux-amd64.zip

get\_ipython().system\_raw('./ngrok http 4050 &')

!sleep 10

!curl -s http://localhost:4040/api/tunnels | python3 -c \

    "import sys, json; print(json.load(sys.stdin)['tunnels'][0]['public\_url'])"

# to check if pyspark is installed

!pyspark --version

"""Olympics File"""

#Uploading the file

from google.colab import files

files.upload()

mt = spark.read.csv("mtcars - mtcars.csv", header = True)

mt.show()

#Q1 Find the top 5 records of mpg attribute

mt.select("mpg").show(5)

#Q2 Find out the records where mpg is less than 18

mt.filter(mt.mpg < 18).show()

from pyspark.sql.functions import \*

from pyspark.sql.types import \*

mtn = mt.withColumn("wt", mt["wt"].cast(FloatType()))

mtn.printSchema()

#Q3 Add a new column to named as "wtTon" to dataframe with values of weight column multiplied by 0.45.

mtc = mtn.withColumn("wtTon", col("wt")\*0.45)

mtc.show()

#Q4

mtw = mtn.withColumn("cyl", mtn["cyl"].cast(IntegerType()))

mtw.printSchema()

mtd = mtw.groupBy("cyl").avg("wt")

mtd.show()

#Q5

mtd.orderBy("avg(wt)").show()

#Q6

mt.select("model","gear","cyl").filter((mt.cyl > 4) & (mt.cyl < 9)).show(21)