Bùi Thanh Tuyền

1050080040

! pip install pyspark

------------------------------------------------

#import libraries

from pyspark import SparkContext

from pyspark.sql import SQLContext

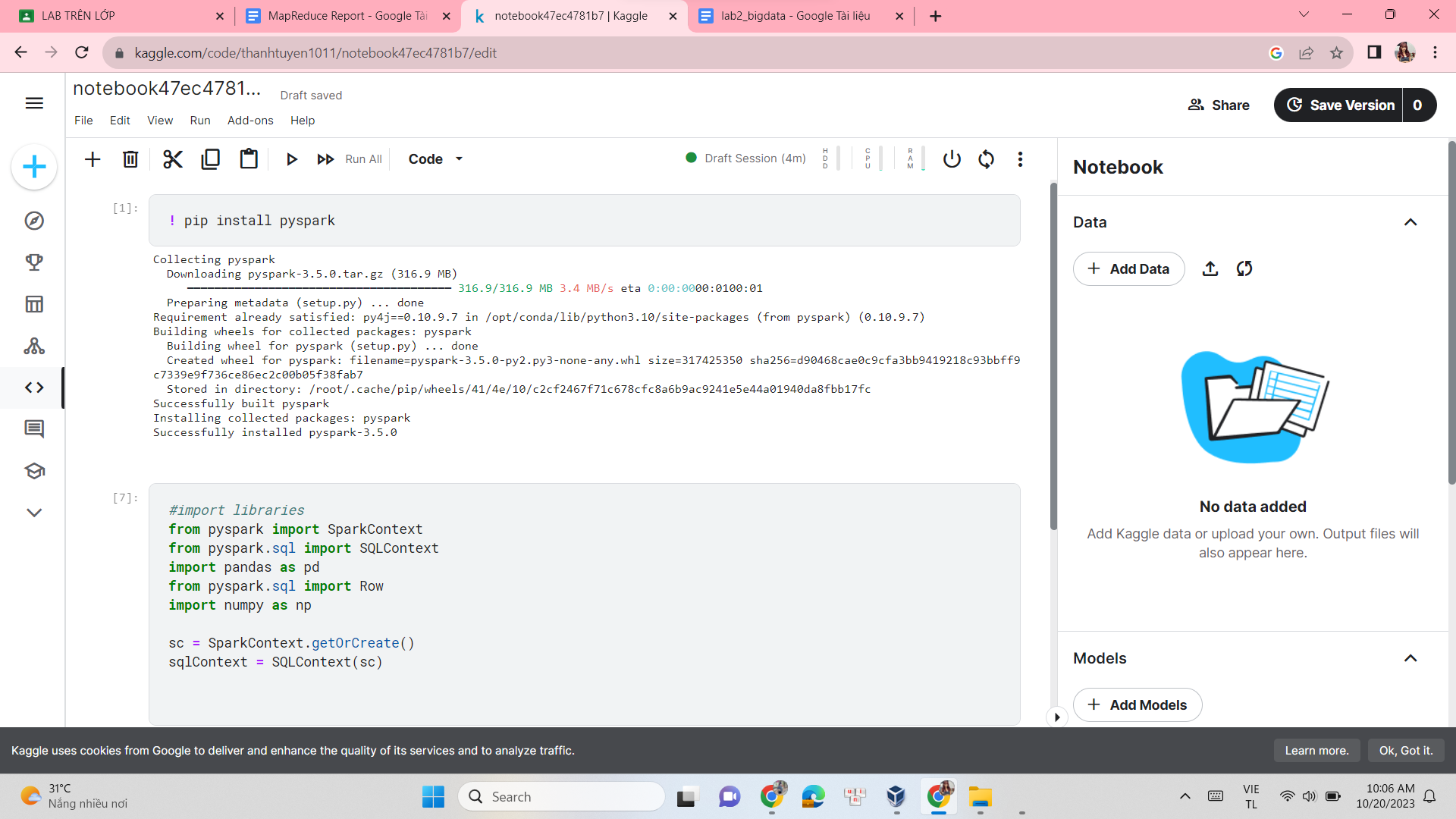
import pandas as pd

from pyspark.sql import Row

import numpy as np

sc = SparkContext.getOrCreate()

sqlContext = SQLContext(sc)



#List of words

a = ["spark","rdd","python","context","create","class"]

b = ["operation", "apache", "scala", "lambda","parallel","partition"]

-------------------------------------------------------

#Making RDDs

rdd\_A = sc.parallelize(a)

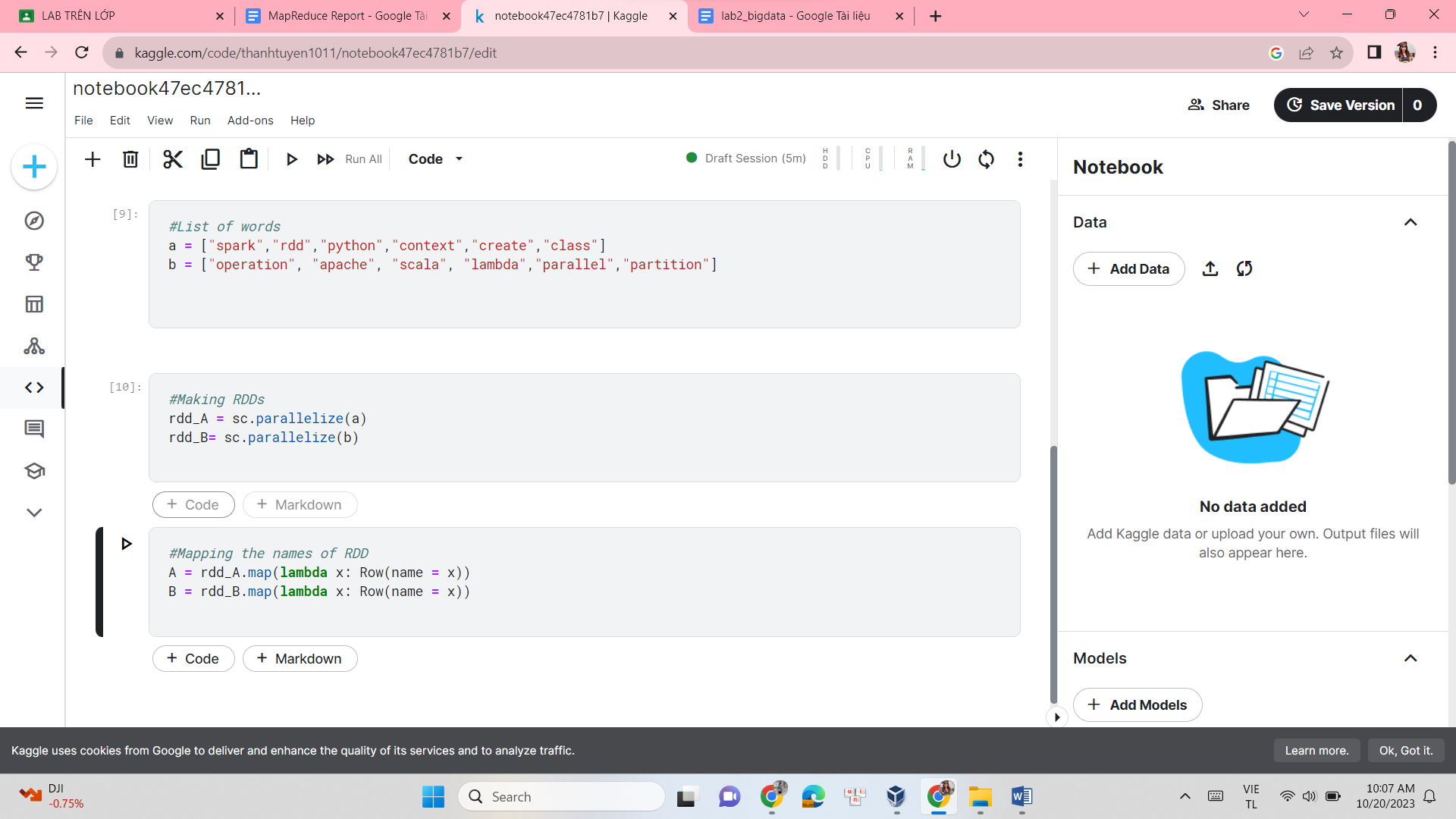
rdd\_B= sc.parallelize(b)

-------------------------------------------

#Mapping the names of RDD

A = rdd\_A.map(lambda x: Row(name = x))

B = rdd\_B.map(lambda x: Row(name = x))



#Creating dataframe from RDD

dfA = sqlContext.createDataFrame(A)

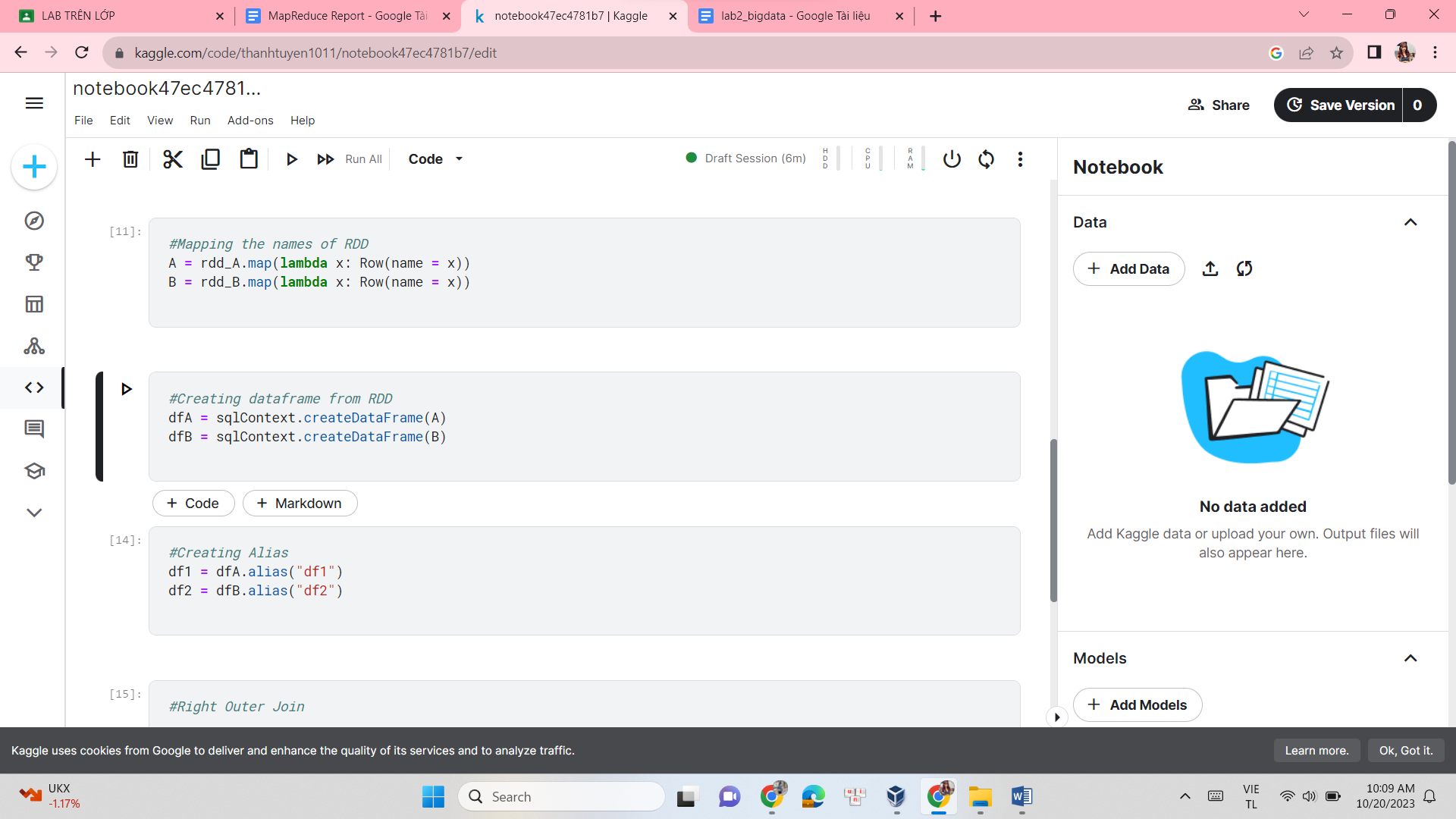
dfB = sqlContext.createDataFrame(B)

----------------------------------

#Creating Alias

df1 = dfA.alias("df1")

df2 = dfB.alias("df2")

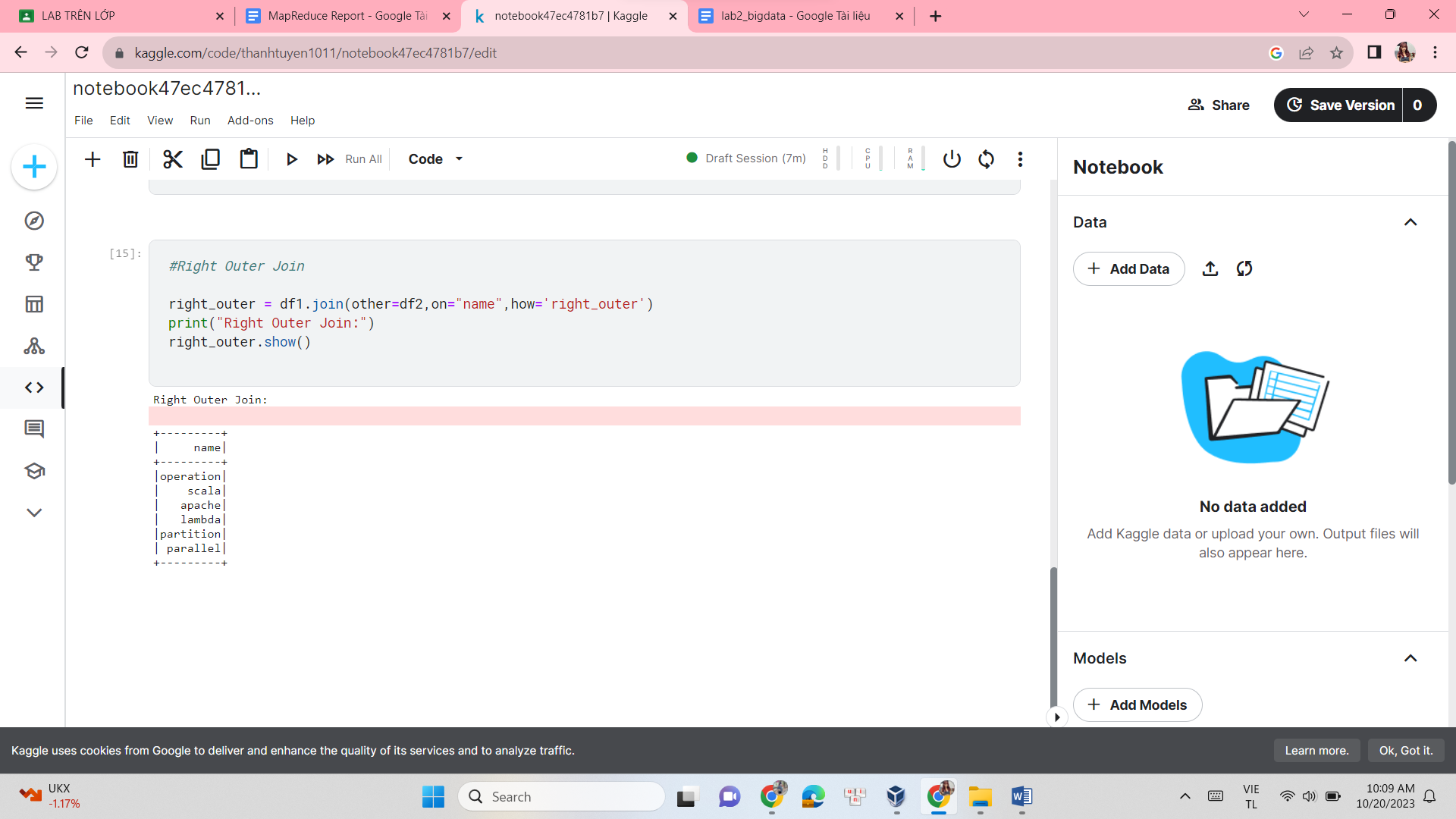


#Right Outer Join

right\_outer = df1.join(other=df2,on="name",how='right\_outer')

print("Right Outer Join:")

right\_outer.show()

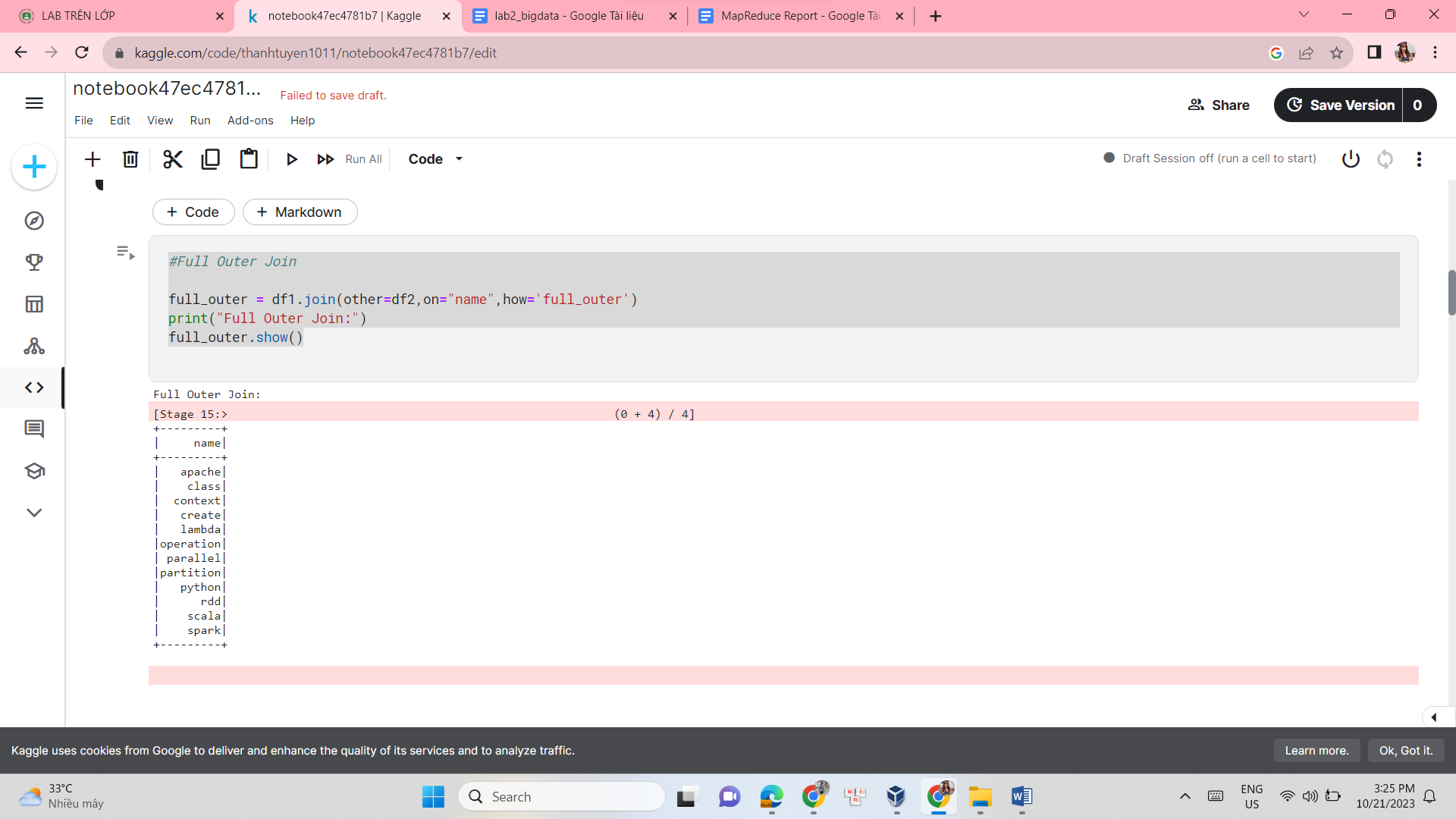


#Full Outer Join

full\_outer = df1.join(other=df2,on="name",how='full\_outer')

print("Full Outer Join:")

full\_outer.show()



#Mapping the RDD

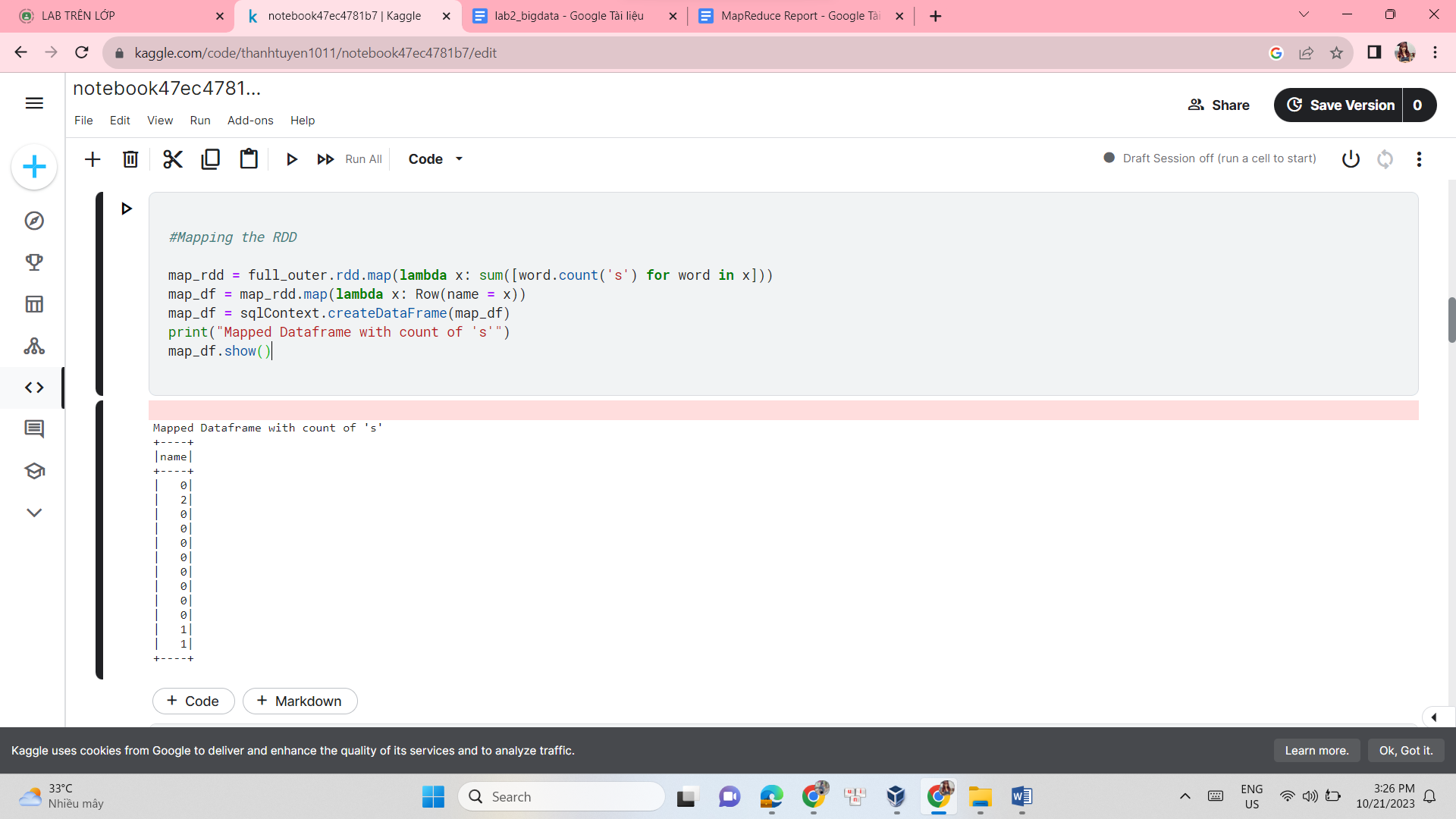
map\_rdd = full\_outer.rdd.map(lambda x: sum([word.count('s') for word in x]))

map\_df = map\_rdd.map(lambda x: Row(name = x))

map\_df = sqlContext.createDataFrame(map\_df)

print("Mapped Dataframe with count of 's'")

map\_df.show()



from pyspark.sql import SparkSession

spark = SparkSession.builder.appName("YourAppName").getOrCreate()

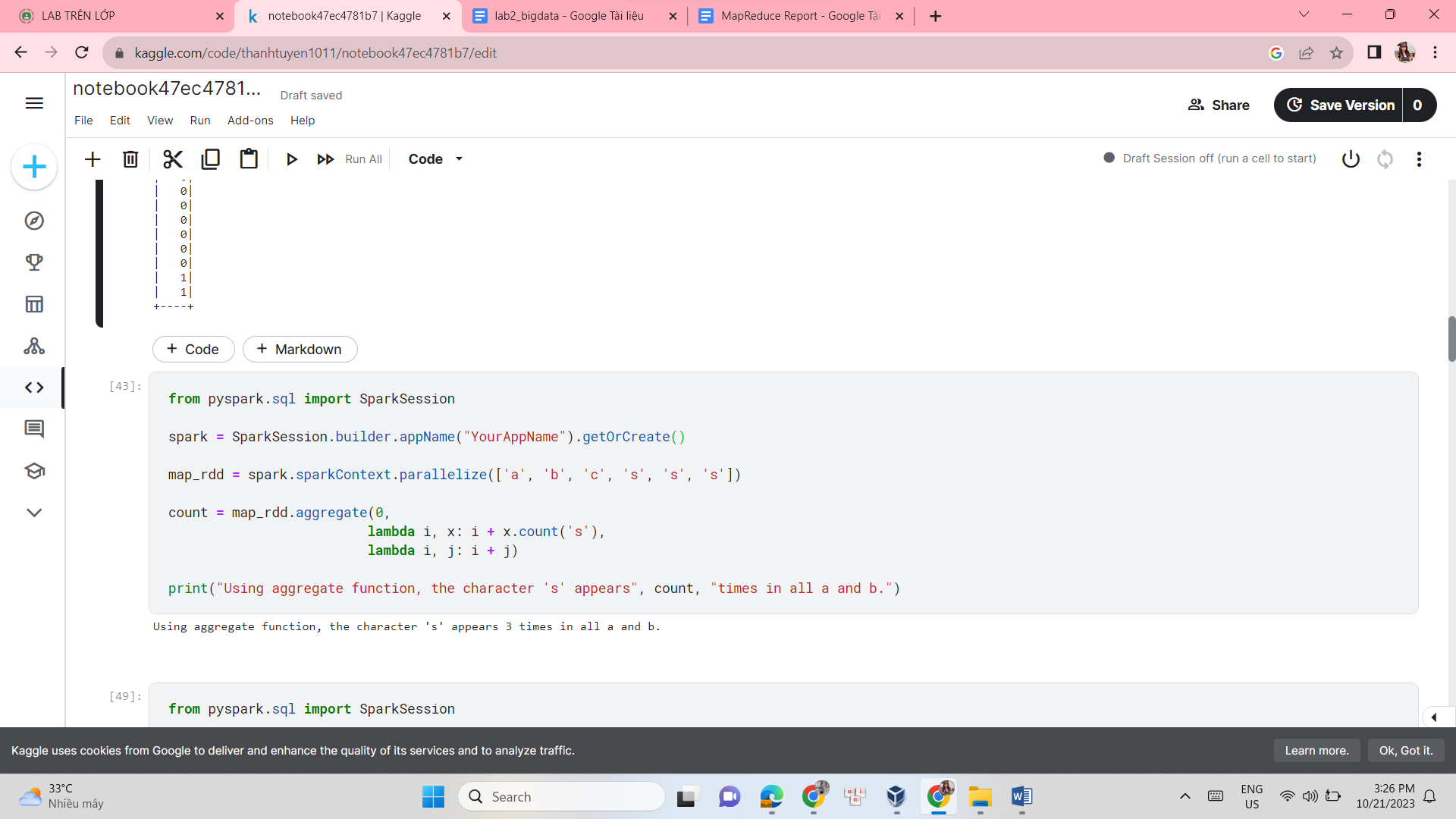
map\_rdd = spark.sparkContext.parallelize(['a', 'b', 'c', 's', 's', 's'])

count = map\_rdd.aggregate(0,

lambda i, x: i + x.count('s'),

lambda i, j: i + j)

print("Using aggregate function, the character 's' appears", count, "times in all a and b.")



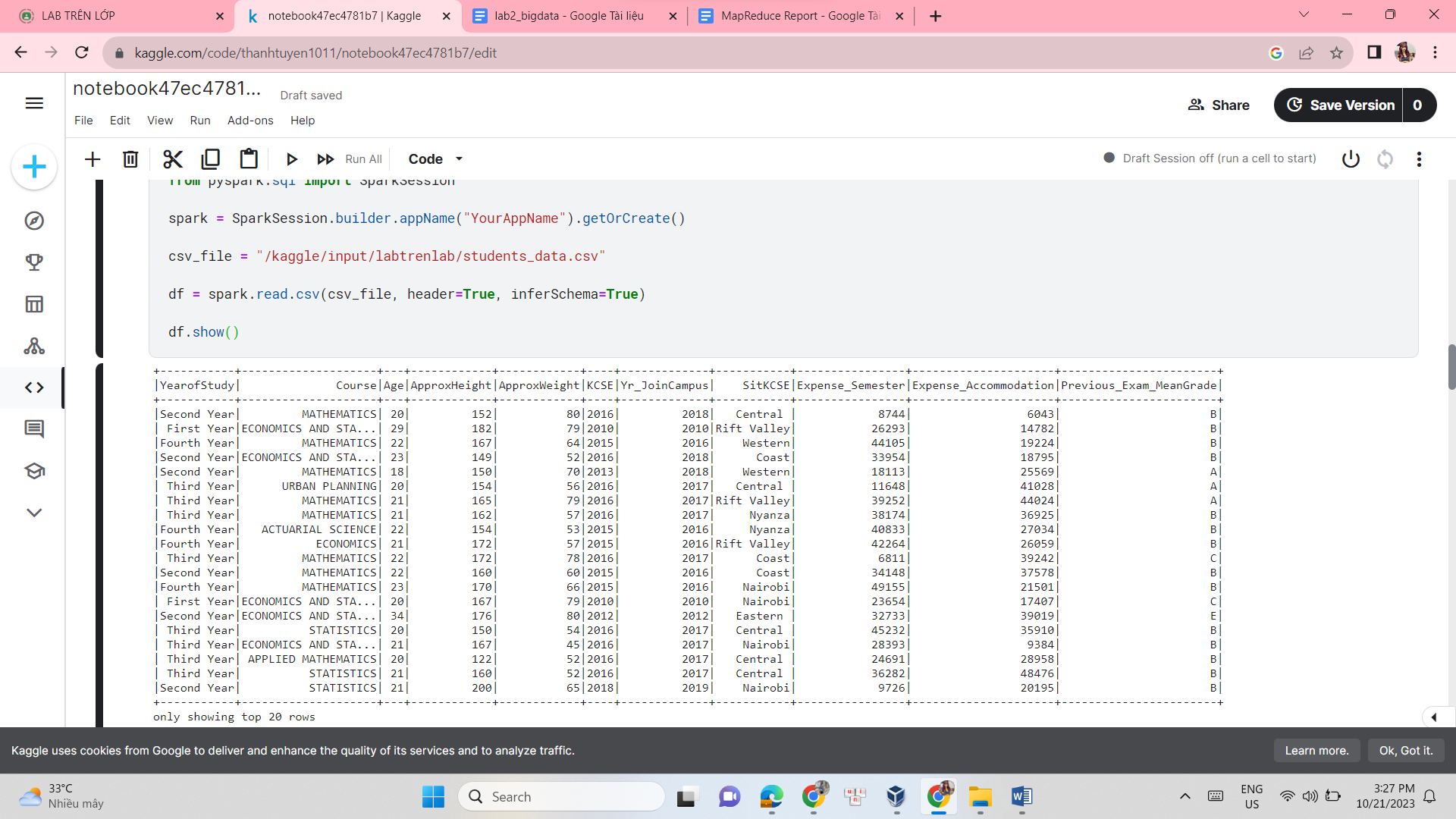
from pyspark.sql import SparkSession

spark = SparkSession.builder.appName("YourAppName").getOrCreate()

csv\_file = "/kaggle/input/labtrenlab/students\_data.csv"

df = spark.read.csv(csv\_file, header=True, inferSchema=True)

df.show()



from pyspark.sql import SparkSession

from pyspark.sql.functions import mean, col

spark = SparkSession.builder.appName("YourAppName").getOrCreate()

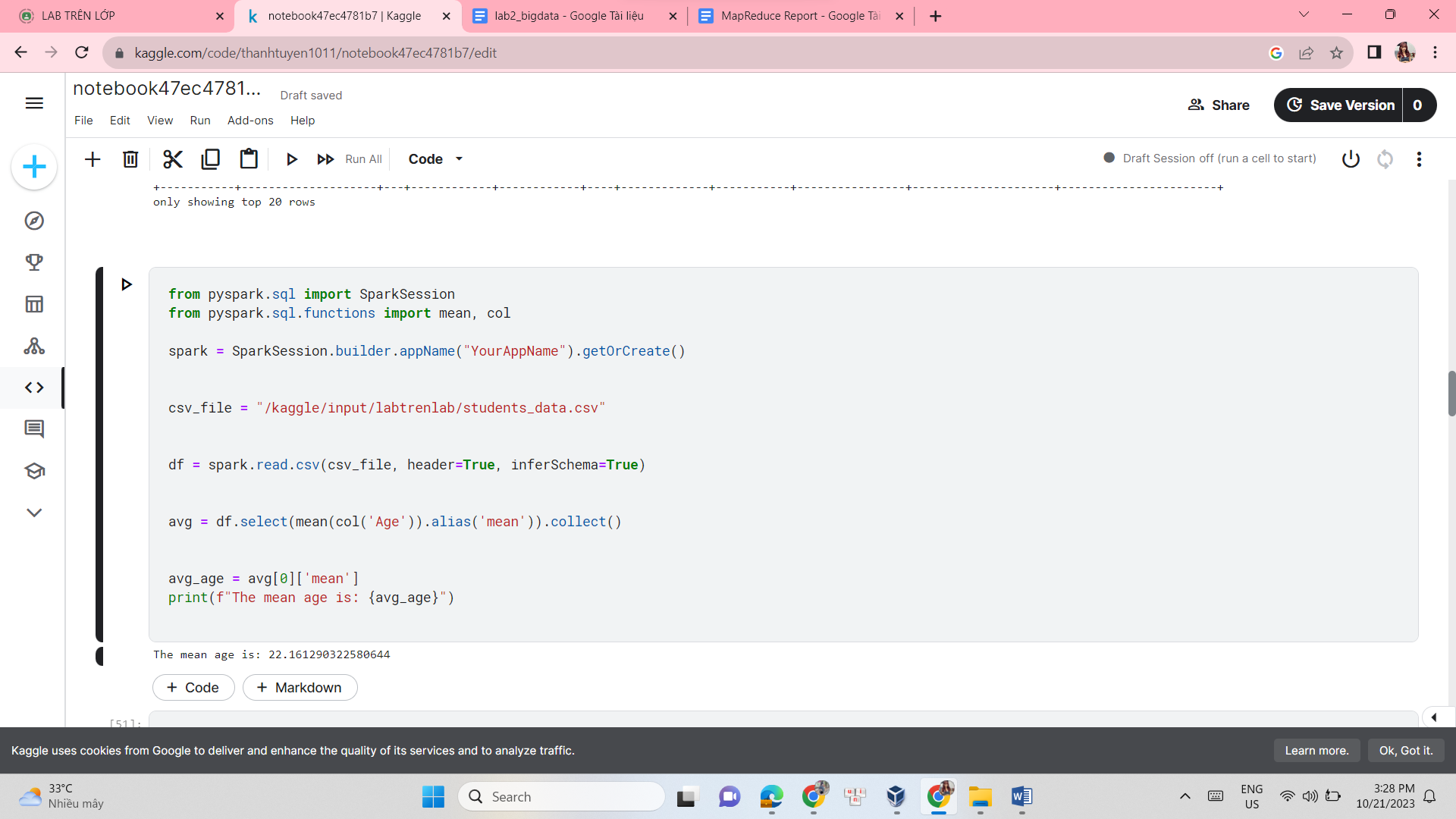
csv\_file = "/kaggle/input/labtrenlab/students\_data.csv"

df = spark.read.csv(csv\_file, header=True, inferSchema=True)

avg = df.select(mean(col('Age')).alias('mean')).collect()

avg\_age = avg[0]['mean']

print(f"The mean age is: {avg\_age}")



from pyspark.sql import SparkSession

from pyspark.sql.functions import mean, col

spark = SparkSession.builder.appName("YourAppName").getOrCreate()

csv\_file = "/kaggle/input/labtrenlab/students\_data.csv"

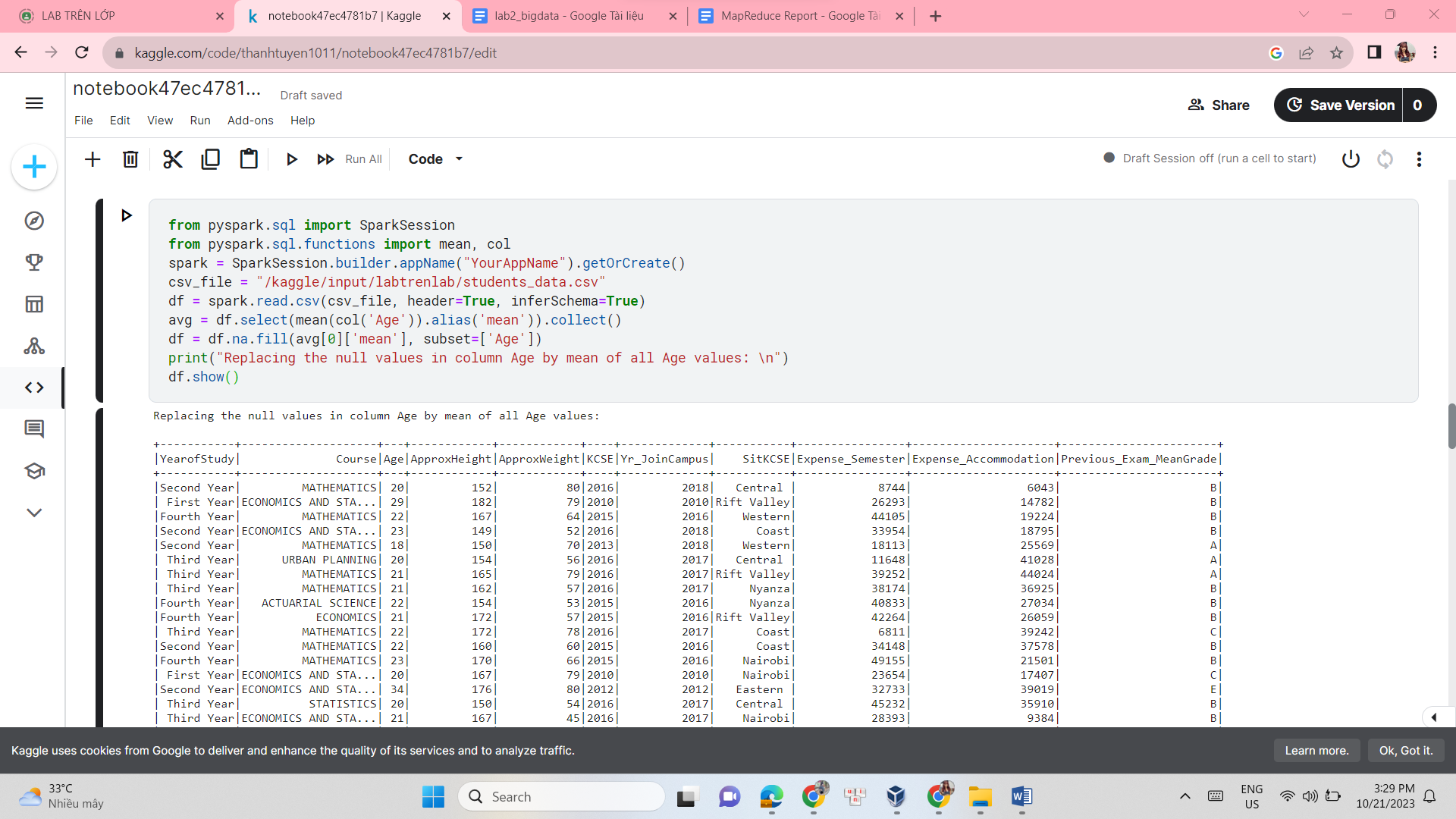
df = spark.read.csv(csv\_file, header=True, inferSchema=True)

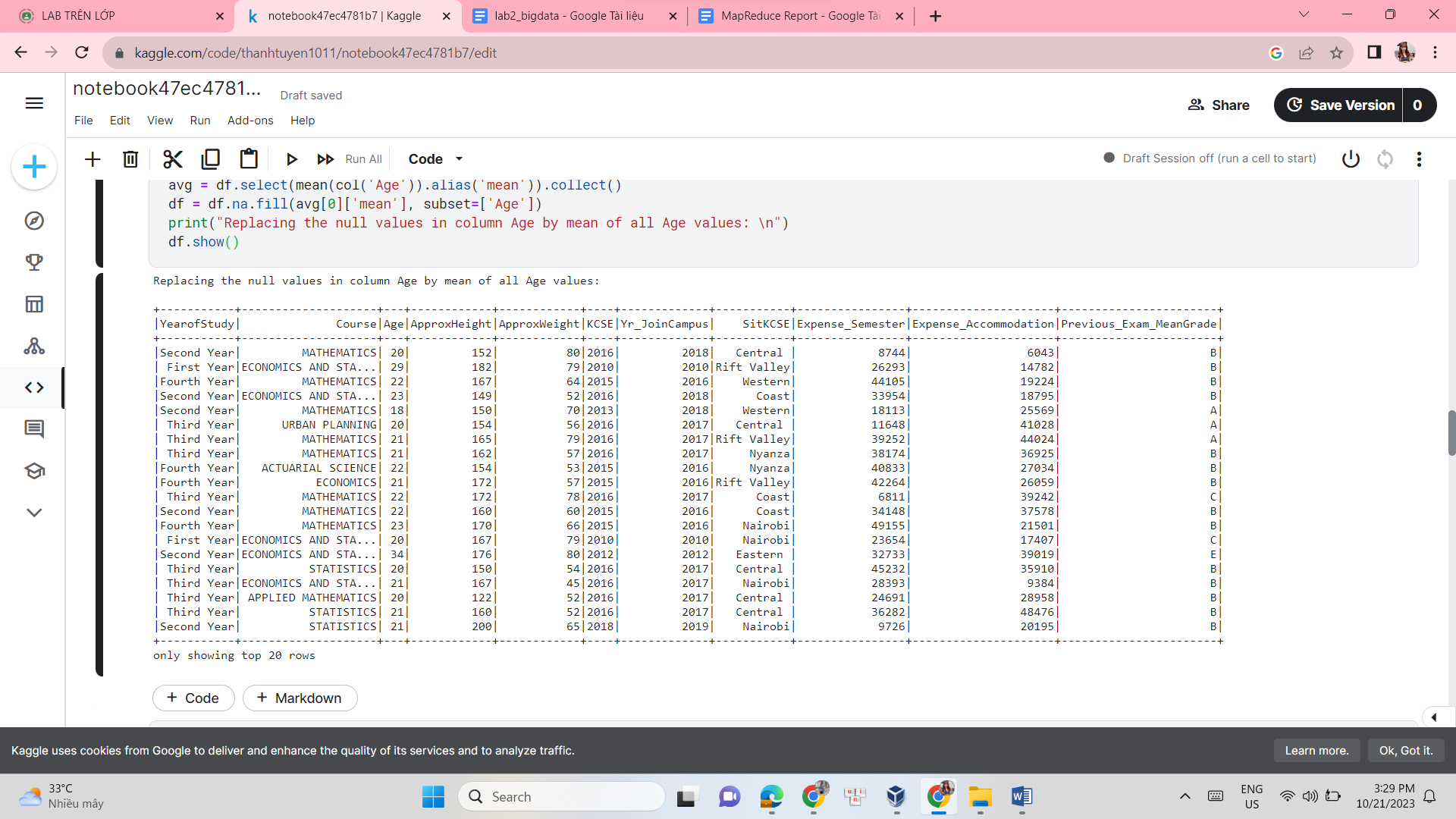
avg = df.select(mean(col('Age')).alias('mean')).collect()

df = df.na.fill(avg[0]['mean'], subset=['Age'])

print("Replacing the null values in column Age by mean of all Age values: \n")

df.show()





from pyspark.sql import SparkSession

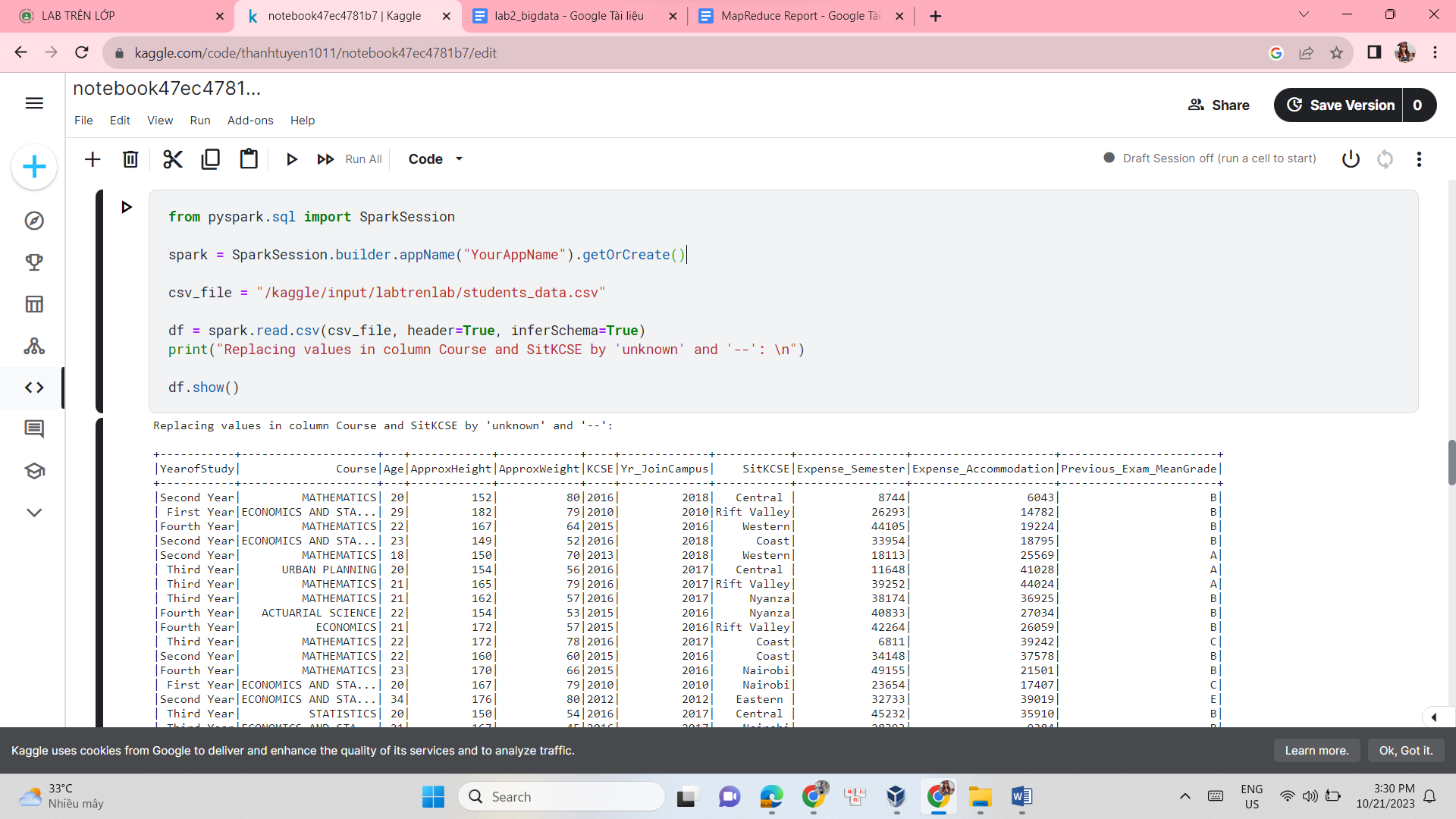
spark = SparkSession.builder.appName("YourAppName").getOrCreate()

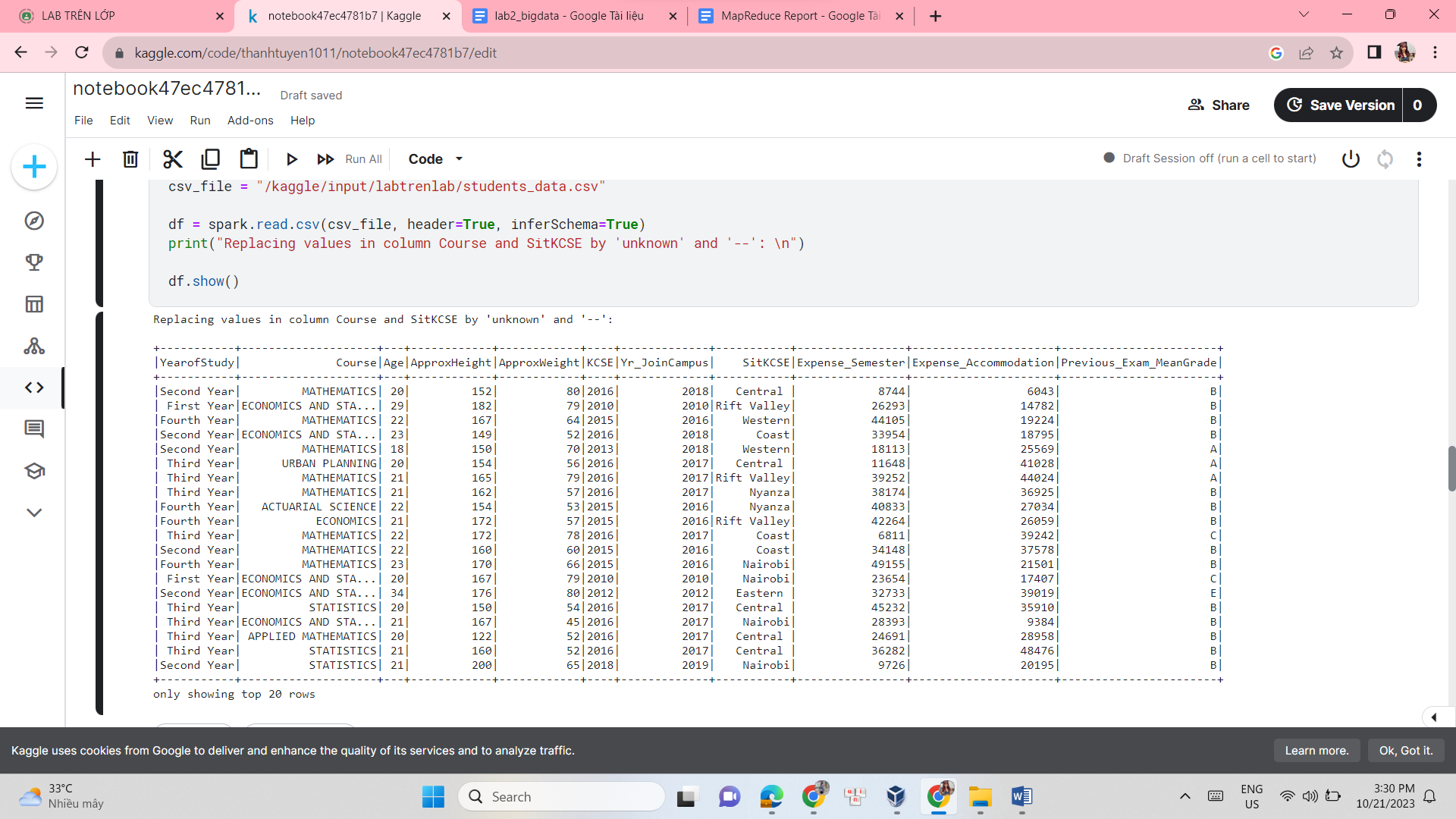
csv\_file = "/kaggle/input/labtrenlab/students\_data.csv"

df = spark.read.csv(csv\_file, header=True, inferSchema=True)

print("Replacing values in column Course and SitKCSE by 'unknown' and '--': \n")

df.show()





from pyspark.sql import SparkSession

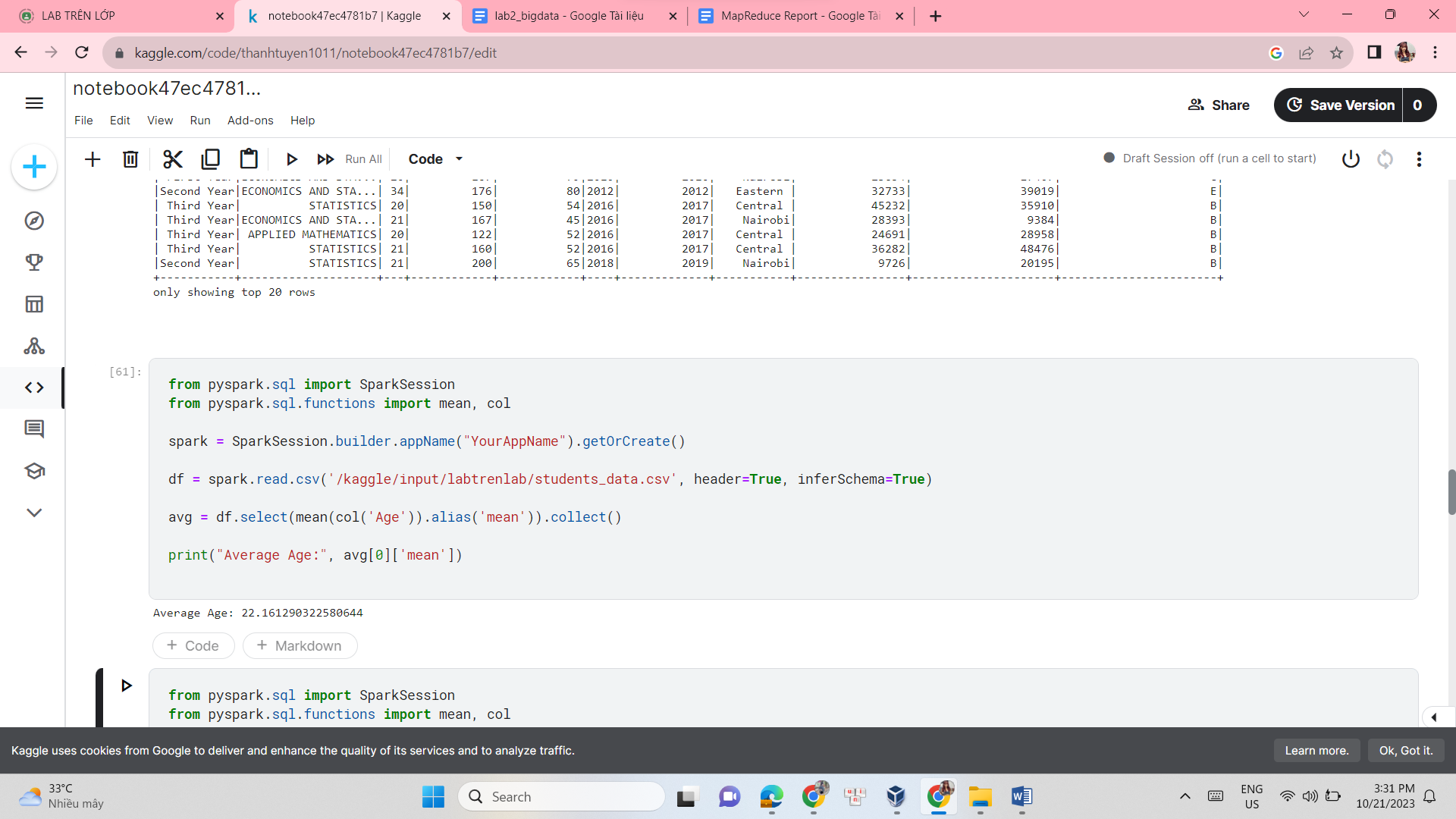
from pyspark.sql.functions import mean, col

spark = SparkSession.builder.appName("YourAppName").getOrCreate()

df = spark.read.csv('/kaggle/input/labtrenlab/students\_data.csv', header=True, inferSchema=True)

avg = df.select(mean(col('Age')).alias('mean')).collect()

print("Average Age:", avg[0]['mean'])



from pyspark.sql import SparkSession

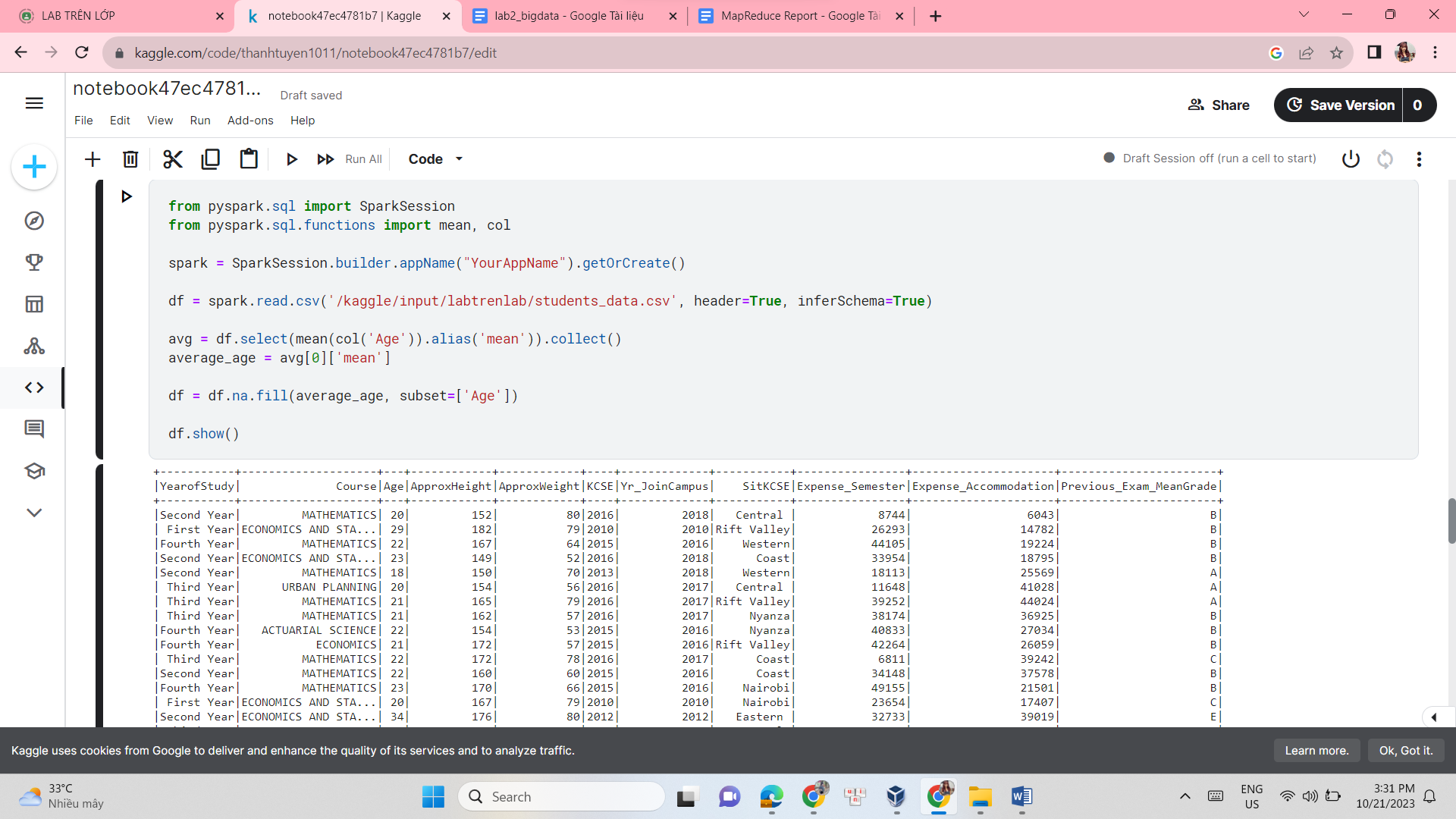
from pyspark.sql.functions import mean, col

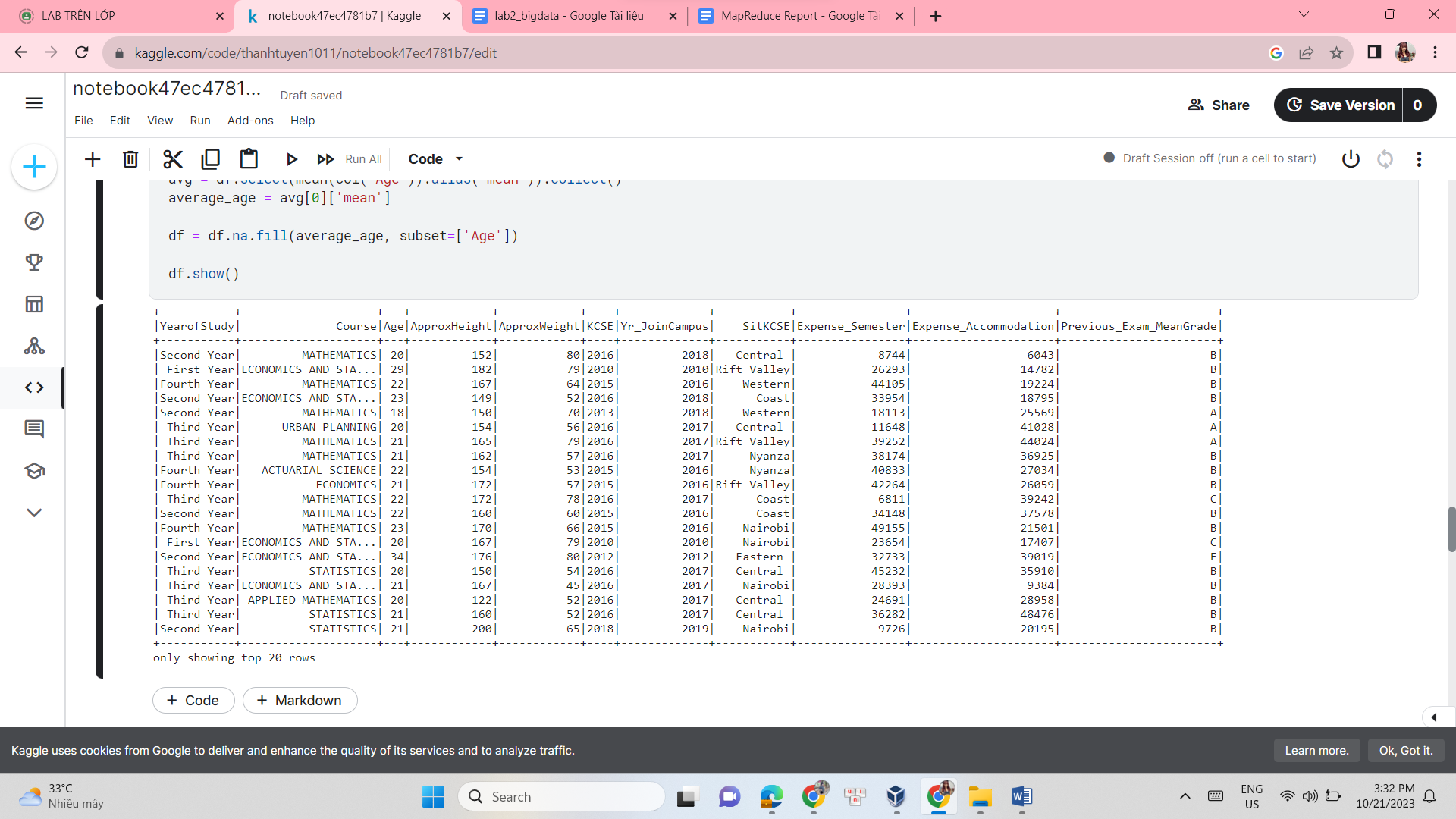
spark = SparkSession.builder.appName("YourAppName").getOrCreate()

df = spark.read.csv('/kaggle/input/labtrenlab/students\_data.csv', header=True, inferSchema=True)

avg = df.select(mean(col('Age')).alias('mean')).collect()

print("Average Age:", avg[0]['mean'])





from pyspark.sql import SparkSession

spark = SparkSession.builder.appName("YourAppName").getOrCreate()

csv\_file = "/kaggle/input/labtrenlab/students\_data.csv"

df = spark.read.csv(csv\_file, header=True, inferSchema=True)

df = df.na.fill({'Course': 'Big data', 'SitKCSE': '--'})

df.show()

