**Screen shorts for each step.**

1. Use the initialdata.csv file
2. Have a look at the data type of each column

df=spark.read.csv("C:/Users/shubhamagrawal.NAGARRO/Downloads/BigData and PowerBI/InitialData.csv",header=True)

for field in df.schema.fields:

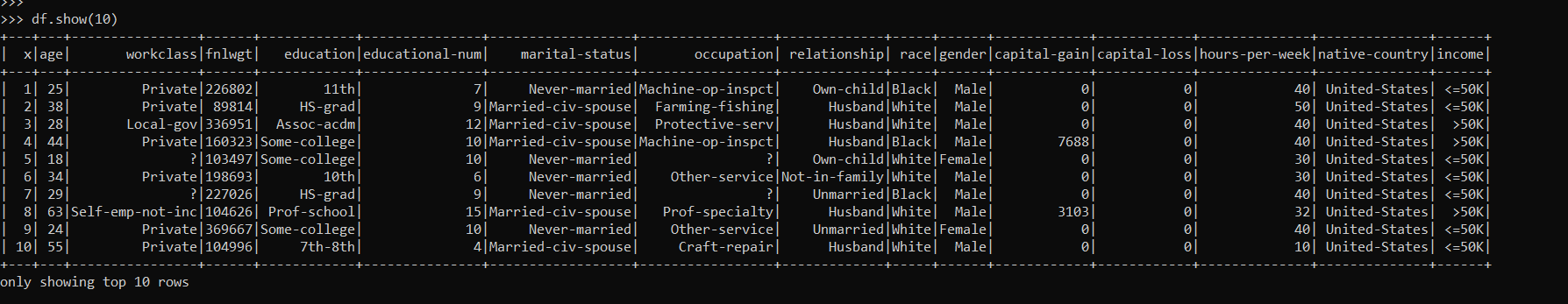
print(field.name +" , "+str(field.dataType))

Text

Description automatically generated

1. Show first 10 rows

df.show(10)



1. Make sure you set recast the column like age-float, workclass-string, fnlwgt-float, education-string, education\_num-float, marital-string, relationship-string, race-string, sex-string, capital\_gain-float , capital\_loss-float, hours\_week-float, native\_country-string, label-string

df2 = df \

.withColumn("x",df.x.cast("int")) \

.withColumn("age",df.age.cast("float")) \

.withColumn("fnlwgt",df.fnlwgt.cast("float")) \

.withColumn("educational-num",df["educational-num"].cast("float")) \

.withColumn("capital-gain",df["capital-gain"].cast("float")) \

.withColumn("capital-loss",df["capital-loss"].cast("float")) \

.withColumn("hours-per-week",df["hours-per-week"].cast("float")) \

Text

Description automatically generated

1. Show first 100 rows with column age and education\_num-float

df2.select("age","educational-num").show(100)

A picture containing rectangle

Description automatically generated

1. show the row count by the education level (hint: groupby and count)

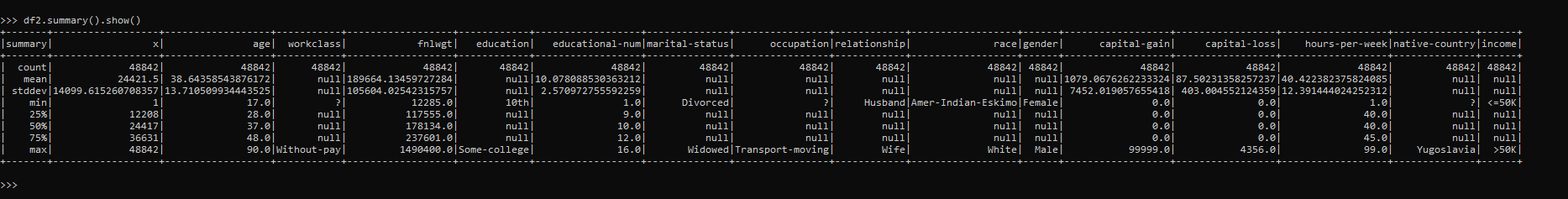
df3=df2.groupBy("educational-num").count()

Text

Description automatically generated

1. show the summary statistic

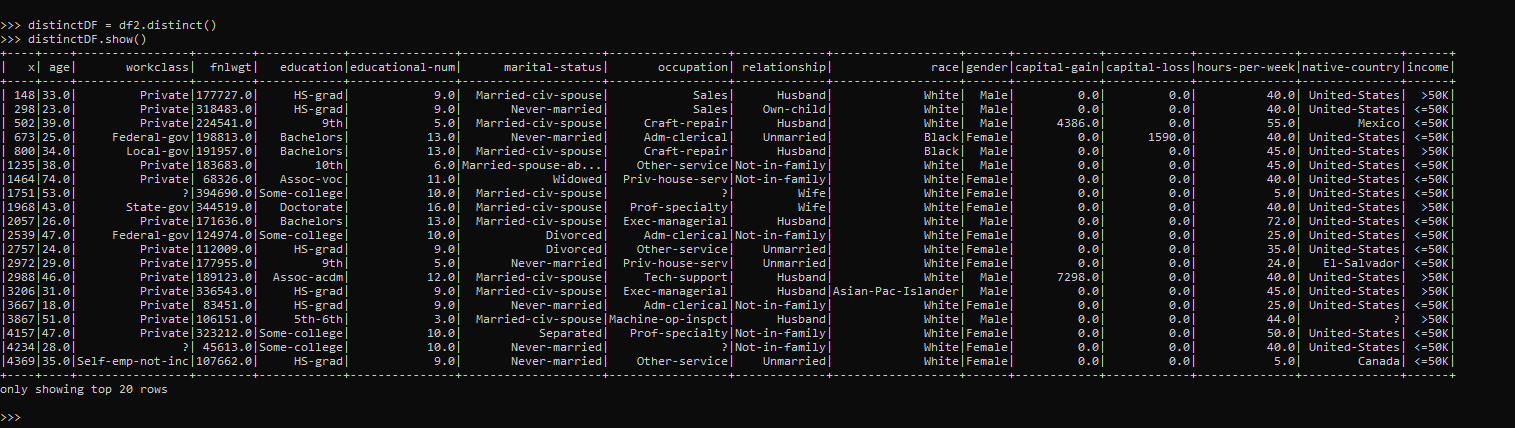
df2.summary().show()



1. count the number of people with income below or above 50k by education level (hint: cross tab)
2. remove duplicate records

distinctDF = df2.dropDuplicates()

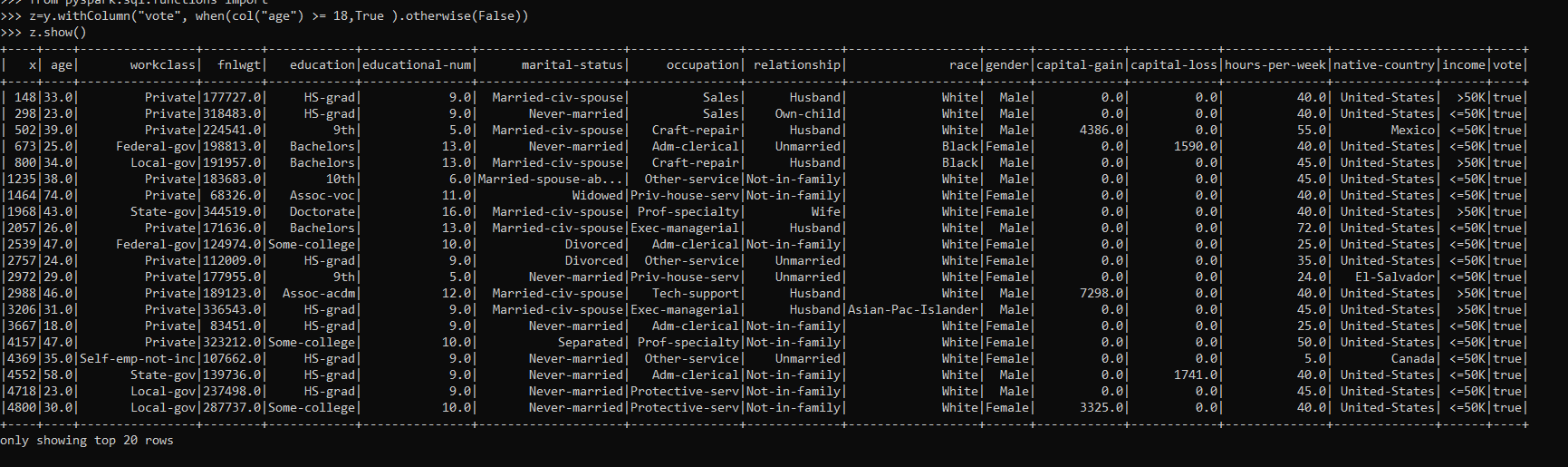
distinctDF.show()



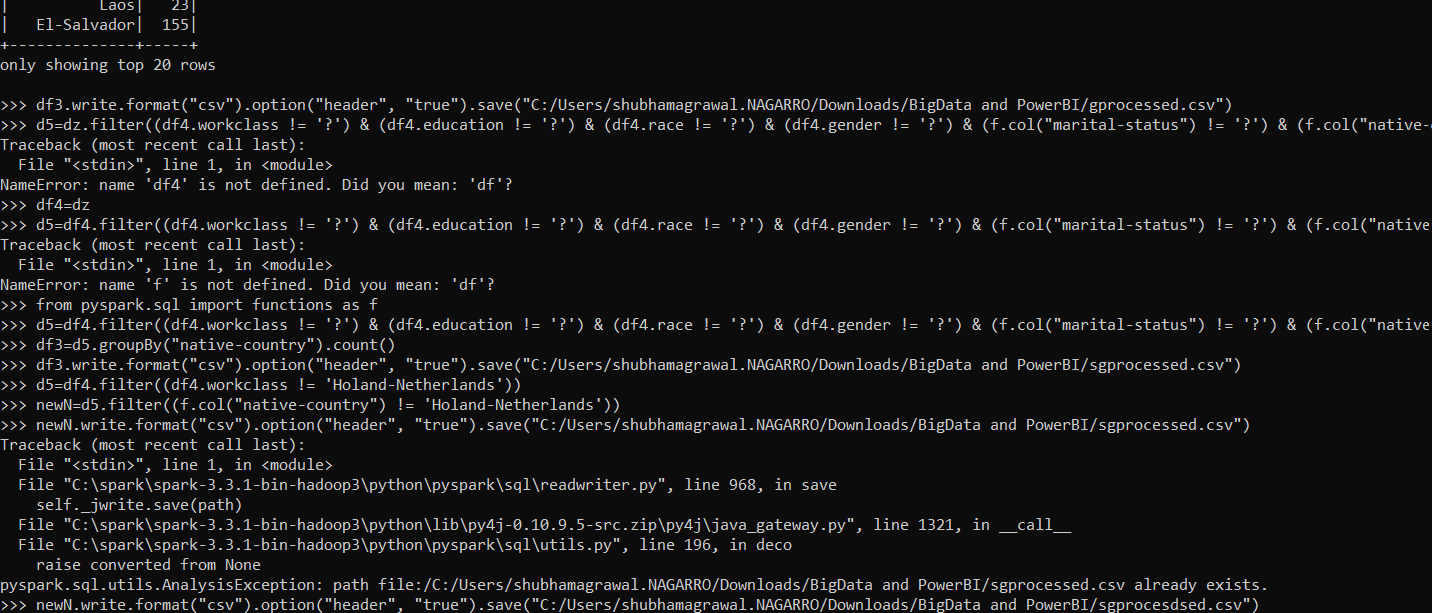
1. add a new column which store the Boolean describing if person is eligible to vote or not (>=18: True, OW: False)

from pyspark.sql.functions import \*

df3=df2.withColumn("vote", when(col("age") >= 18, True).otherwise(false))



1. remove the records which belongs to the native\_country who has less then 10 records



1. remove the records which has null value in any of the column

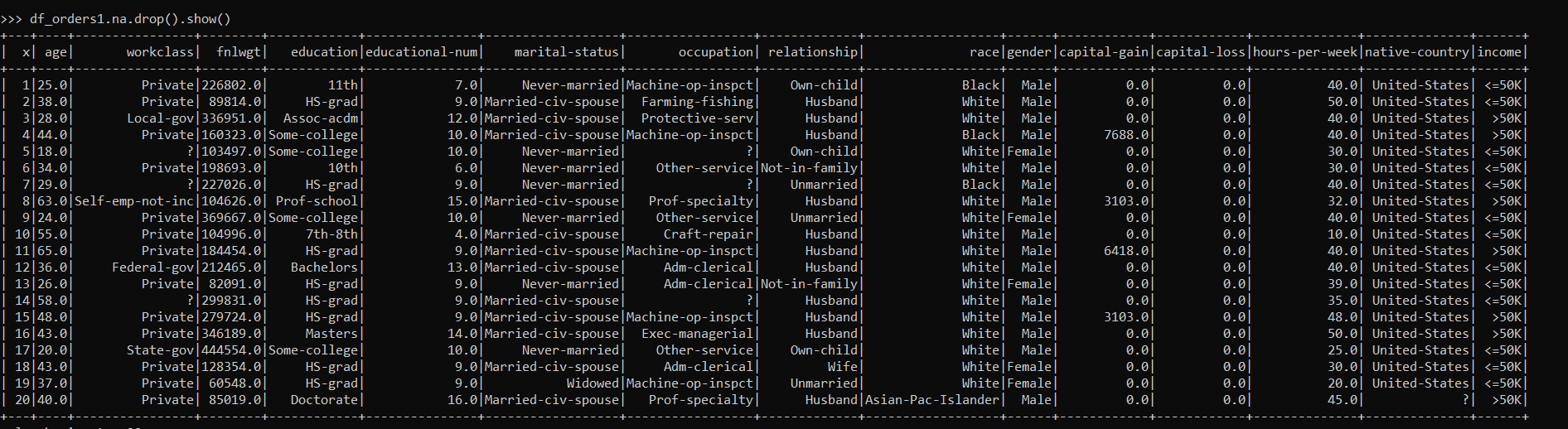
df.na.drop().show()

//filter Noice data

*from pyspark.sql import functions as f*

d5=df4.filter((df4.workclass != '?') & (df4.education != '?') & (df4.race != '?') & (df4.gender != '?') & (f.col("marital-status") != '?') & (f.col("native-country") != '?'))

>>> d5.show()



1. Convert the above selected column data frame into an RDD and save it into a csv file(processed.csv).

X= Df5.rdd

df5. coalesce(1).write.format("csv").option("header", "true").save("C:/Users/shubhamagrawal.NAGARRO/Downloads/BigData and PowerBI/proces.csv")

To create a sigle csv - coalesce(1)