DATA FILTRATION USING PYSPARK

Q.1) There is a csv file titled "staff_record". There are 6 columns. In some column either the value is null or it is marked with 'unavailable'.

TASKS

i) Replace string "unavailable" with actual null value.

ii) If there are null values more than 75% in a column, then delete that column.

iii) If there are missing numeric values such as salary, then fill that value with mean value.

iv) Filter the records keeping ony the data of staff of age more than 40.

v) Export the dataframe in csv file format in specific directory.

#note: I had used jupyter notebook to run pyspark.

SOLUTIONS

i. Import spark session

→ from pyspark.sql import SparkSession

# import col & when

→ from pyspark.sql.functions import col, when

# create spark session

→ spark=SparkSession.builder.appName('csvfilter').getOrCreate()

# start spark session

→ spark

# read csv file with header

→ df=spark.read.csv('C:/spark practice/csv/staff_records.csv',header=True, inferSchema=True)

#note: both types of slash / and \ can be used

# List all column in dataframe

```
→ columns=df.columns
```

# Replace'unavailable' with actual null value with loop on each column

```
→ for column in df.columns:
   df=df.withColumn(column, when(col(column)== "unavailable", None).otherwise(col(column)))
```

ii. Find null percentage

```
→ def get_null_percentage(column):
   row_numb = df.count()
   null_count = df.filter(col(column).isNull()).count()
   return null_count / row_numb
```

# Assign threshold value and list the column if it exceed the threshold.

```
→ threshold = 0.75
   col_drop = [column for column in df.columns if get_null_percentage(column) > threshold]
```

# Delete the listed column

```
→  df = df.drop(*col_drop)
```

iii. Fill the missing value of 'salary' column with mean value

```
→ from pyspark.ml.feature import Imputer
   imputer=Imputer(
   inputCols=['salary'],
   outputCols=["{}_imputed".format(c) for c in['salary']]
   ).setStrategy("mean")
→ df=imputer.fit(df).transform(df)
```

iv. Only select the data of employee having age more than 40.

→ df1=df.filter(col("age")>40)

v. Gather the dataframe in single partition

→ df1=df1.coalesce(1)

```
df1.show()
```

# Export the dataframe in sngle csv file inside members folder.

→ df1.write.csv('H:/spark out/members',header=True)

##NOTE: We can replace the string "unavailable" with null value using regex in the following way.

→ for column in df.columns:

```
df=df.withColumn(column, regexp_replace(col(column), "null", ""))
```

##NOTE: sample csv file below

Create staff_record.csv file using the given data sample

```
name,age,address,email,salary,post
Polly Shaw,57,163 Habi Lane,lawicul@edkefhoj.mk,35000,unavailable
Leona Cummings,25,unavailable,ho@hug.sj,39000,unavailable
Mittie Crawford,53,unavailable,gis@zu.dz,,unavailable
Ricardo Flores,39,unavailable,ruk@wituz.br,35000,unavailable
Evelyn Wolfe,21,1660 Ropi Loop,niwaice@veteka.bz,34000,unavailable
Frederick Watkins,41,1213 Kiniz Manor,hem@pedgo.sy,34000,unavailable
Celia Davis,34,1493 Sefki Trail,lak@hon.pl,36000,unavailable
Pearl Harvey,53,unavailable,cikifi@toj.vu,,unavailable
Dylan Woods,64,566 Vahek Plaza,amoviipa@alabuz.kg,39000,unavailable
Ann Robinson,43,1255 Jokib Glen,wemmozug@tudtevi.us,39000,unavailable
Joshua Lucas,36,1480 Nigepa Extension,rathozo@zapozvu.sy,,unavailable
Henry Mendoza,48,380 Dalses Mill,emje@ru.tp,36000,unavailable
Myra Norris,40,1051 Iklat Pike,lakahza@toba.pw,39000,unavailable
Albert Ball,29,unavailable,segbo@ejo.hn,,unavailable
Todd Sims,36,468 Tedu Parkway,vala@jedoog.mw,36000,unavailable
Max Kelley,19,1416 Dazva Boulevard,itje@bin.ht,,unavailable
Elijah Holland,60,809 Gokoh Extension,za@ne.py,36000,unavailable
Lucinda Thompson,44,1987 Obaze Avenue,id@ebbu.bf,36000,unavailable
Isabel Diaz,36,97 Cojip Boulevard,otozufer@ofavove.sm,359000,IT
Eleanor Warren,32,695 Sojjub Road,kansolede@uwikit.vn,,unavailable
Caroline Bennett,42,1354 Wegfa Parkway,segofowu@votela.ca,39000,unavailable
Chase Stevens,63,1636 Lossil Extension,dusod@humebi.kn,,unavailable
Sophia Dean,24,100 Gojaw Drive,afobir@ip.sm,38000,unavailable
Alta Thomas,47,1760 Dobog Highway,sinvu@ofojidu.bh,35000,account
Brent Rice,23,1183 Waka Plaza,metulsab@iz.cu,38000,unavailable
Jonathan Richardson,46,942 Buosi Court,wapokdo@jub.ie,,unavailable
Alexander Lyons,27,221 Idrur Square,oj@de.eh,37000,unavailable
```

Donald Nash,56,894 Gejtoj River,folge@ur.as,37000,unavailable

Jesus Santiago,31,1243 Gapzam Circle,tawagozez@ug.tr,35000,unavailable

Micheal Cain,46,425 Jeke Loop,vip@ceewo.so,36000,unavailable

Mitchell Watkins,53,366 Rihubu Court,perik@ro.sn,,unavailable

Adam Cohen,58,1965 Epiko Drive,zad@ruceup.tn,36000,unavailable

Juan Francis,36,1139 Nonez Glen,dekhuju@civha.pl,35000,unavailable

Millie Caldwell,39,597 Viwo Pike,wipseffa@zorkeb.tp,35000,unavailable

Rachel Weber,45,unavailable,setocuk@fizutfa.ch,,unavailable

Austin Brady,23,380 Sutveh Junction,ruluidi@oluf.mx,35000,unavailable

Marie Carpenter,32,unavailable,nom@wus.bg,34000,unavailable

Carlos Woods,64,433 Owza Parkway,nabmebam@kelig.kn,32000,unavailable

Dollie Sims,29,147 Zunci Terrace,jodbadfad@nof.io,31000,unavailable

Maurice Keller,41,527 Nizuk Ridge,bejug@amosebu.mh,30000,unavailable

Shane Cohen,64,unavailable,lesaan@ziz.bm,39000,unavailable

Jesse Mendez,27,1051 Halu Loop,go@subonoew.gn,38000,unavailable

Viola Morris,61,unavailable,bucnovocu@kire.us,,unavailable

Calvin Joseph,29,unavailable,mizi@ri.mc,37000,unavailable

Olga Goodman,49,1884 Larav Street,popi@velimna.ck,,unavailable

Dominic Hansen,39,688 Ivasu Path,da@haabivu.pe,35000,unavailable

Isaac Curry,34,1337 Imcu View,jebde@gati.sy,,unavailable

Clyde Parsons,18,unavailable,mi@diab.pf,31000,unavailable

Cameron Malone,19,398 Ufsez Pass,zo@nep.mr,32000,unavailable

Craig Martin,30,1648 Mofor Parkway,tu@nomuk.ms,35000,unavailable