-- ATIVIDADE 17: RAYSSA, MONNICY, DANIELE

"""

Dados do portal da transparência: Viagens a serviço (2020, 2021)

Grupo: Daniele, Monnicy, Rayssa

"""

from pyspark import SparkContext, SparkConf

from pyspark.sql import SparkSession

from pyspark.sql import functions as B

from pyspark.sql.types import FloatType

spark = SparkSession.builder.appName("OTR").config("spark.sql.caseSensitive", "True").getOrCreate()

# IMPORTANDO OS ARQUIVOS CSV

viagem2020 = spark.read.format("csv").option("header", "true").option("delimiter", ";").option("inferSchema", "true").option("encoding", "ISO-8859-1").load(r"C:\Users\monni\Documents\Documentos\_SoulCode\Atividades\Python\2020\_Viagem.csv")

viagem2021 = spark.read.format("csv").option("header", "true").option("delimiter", ";").option("inferSchema", "true").option("encoding", "ISO-8859-1").load(r"C:\Users\monni\Documents\Documentos\_SoulCode\Atividades\Python\2021\_Viagem.csv")

# UNINDO

viagens = viagem2020.union(viagem2021)

# MUDANÇA NOME DAS COLUNAS

viagens = viagens.select(viagens["Identificador do processo de viagem"].alias("id\_processo"),viagens["Situação"].alias("Situacao"),viagens["Viagem Urgente"].alias("Viagem\_urgente"),viagens["Justificativa Urgência Viagem"].alias("Justificativa"),viagens["Nome órgão solicitante"].alias("Orgao\_solicitante"),viagens["Nome"].alias("Solicitante"),viagens["Cargo"].alias("Cargo"),viagens["Período - Data de início"].alias("Data\_inicio"),viagens["Período - Data de fim"].alias("Data\_fim"),viagens["Destinos"].alias("Destinos"),viagens["Motivo"].alias("Motivo"),viagens["Valor diárias"].alias("Valor\_diarias"),viagens["Valor passagens"].alias("Valor\_passagens"),viagens["Valor outros gastos"].alias("Outros\_gastos"))

#viagens.printSchema()

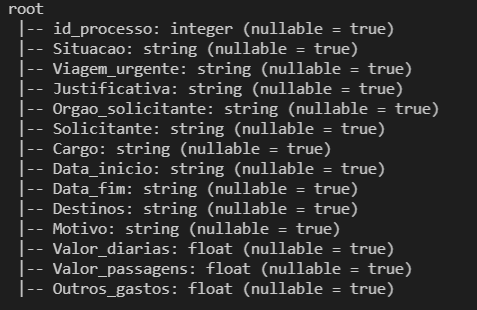
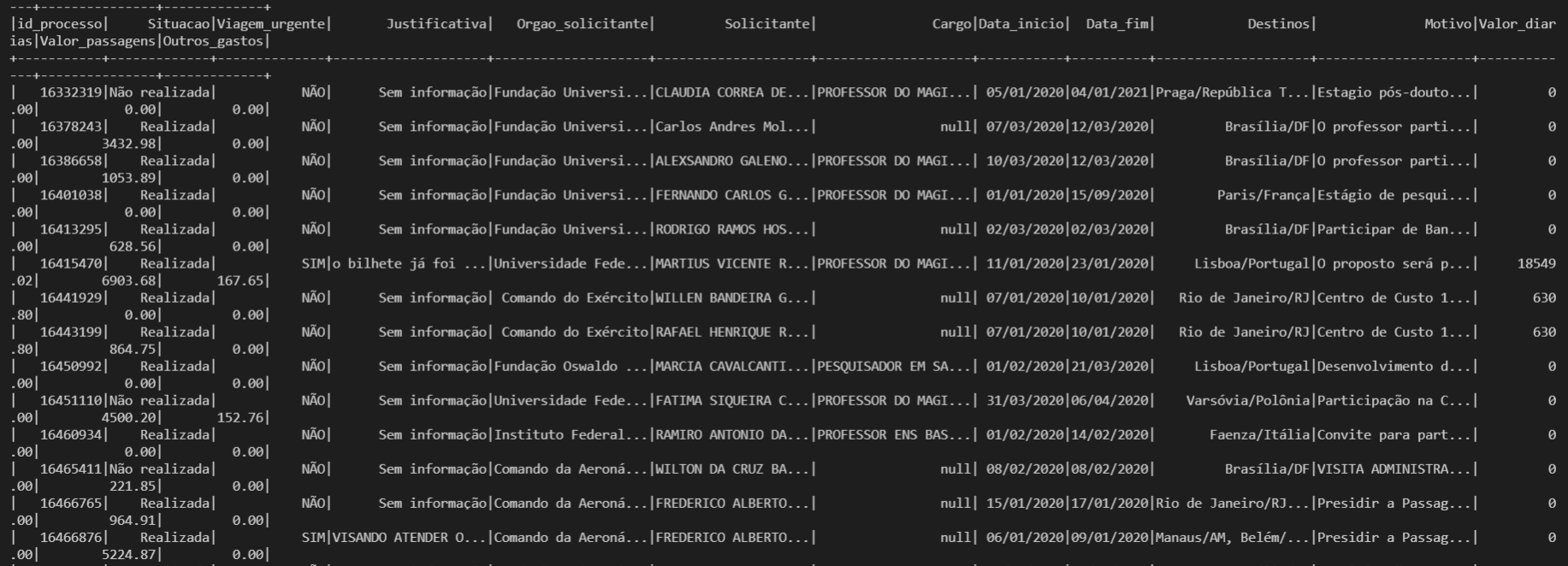
# CONVERSÃO DA VIRGULA PARA PONTO

viagens = viagens.withColumn("Valor\_diarias",B.regexp\_replace("Valor\_diarias",",","."))

viagens = viagens.withColumn("Valor\_passagens",B.regexp\_replace("Valor\_passagens",",","."))

viagens = viagens.withColumn("Outros\_gastos",B.regexp\_replace("Outros\_gastos",",","."))

viagens.show()



#CONVERSÃO PARA PARQUET

viagens.write.parquet(r"C:\Users\monni\Documents\Documentos\_SoulCode\Atividades\Parquet\_Ativ\_17")

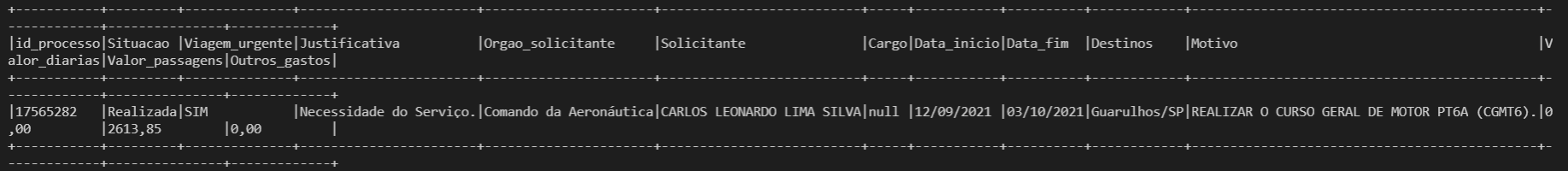
viagens = spark.read.parquet(r"C:\Users\monni\Documents\Documentos\_SoulCode\Atividades\Parquet\_Ativ\_17")

#viagens.printSchema()

# PESQUISA POR SOLICITANTE

viagens.groupBy("Solicitante").count().orderBy("count").show(truncate=False)

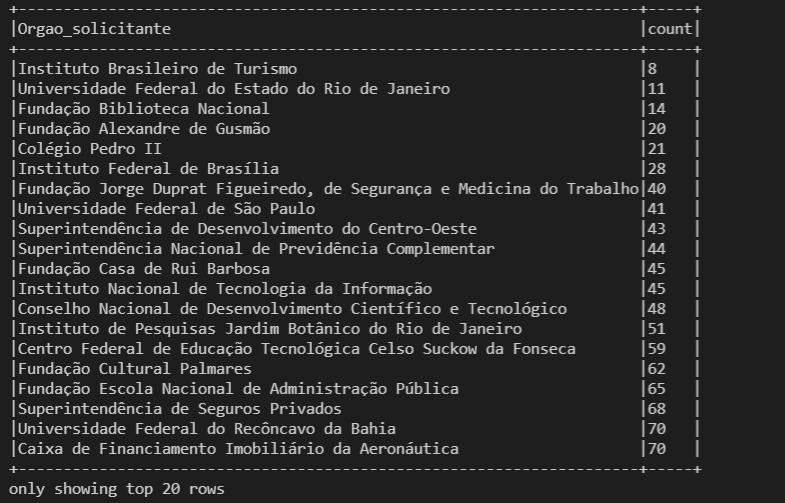
viagens.filter(viagens["Solicitante"] == "CARLOS LEONARDO LIMA SILVA").show(truncate=False)

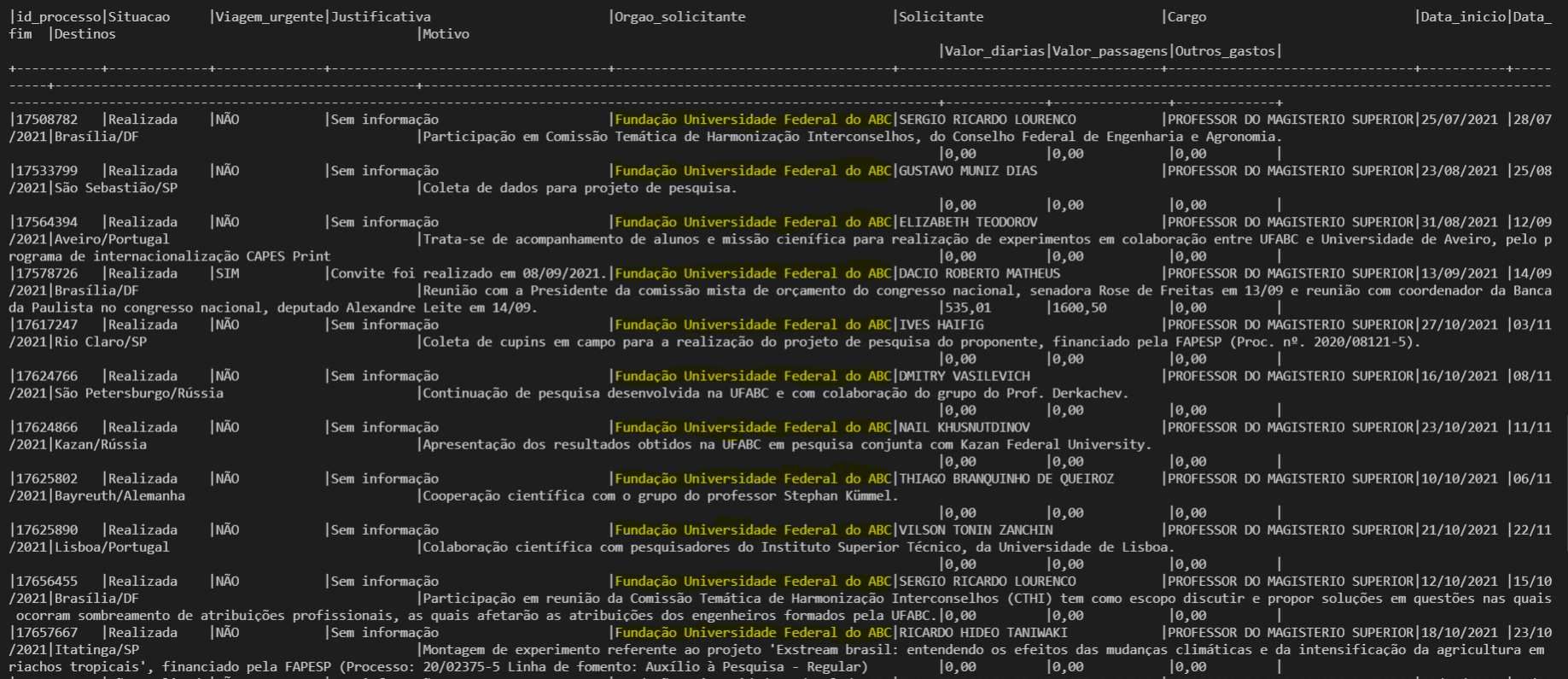


# PESQUISA POR ORGÃO SOLICITANTE - GERAL E ESPECIFICO

viagens.groupBy("Orgao\_solicitante").count().orderBy("count").show(truncate=False)

viagens.filter(viagens["Orgao\_solicitante"] == "Fundação Universidade Federal do ABC").show(truncate=False)





# VALOR DAS DIÁRIAS:

valor\_diaria\_soma = viagens.groupBy("Orgao\_solicitante").agg(B.sum("Valor\_diarias").alias("Total valor gasto em diárias"))

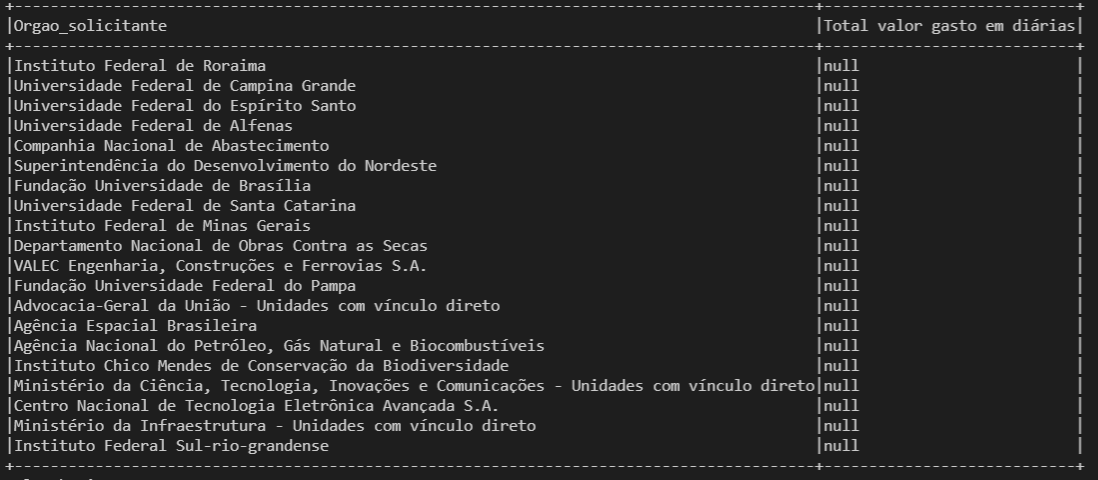
max\_valor\_diaria = valor\_diaria\_soma.orderBy(B.desc("Total valor gasto em diárias")).show(truncate=False)

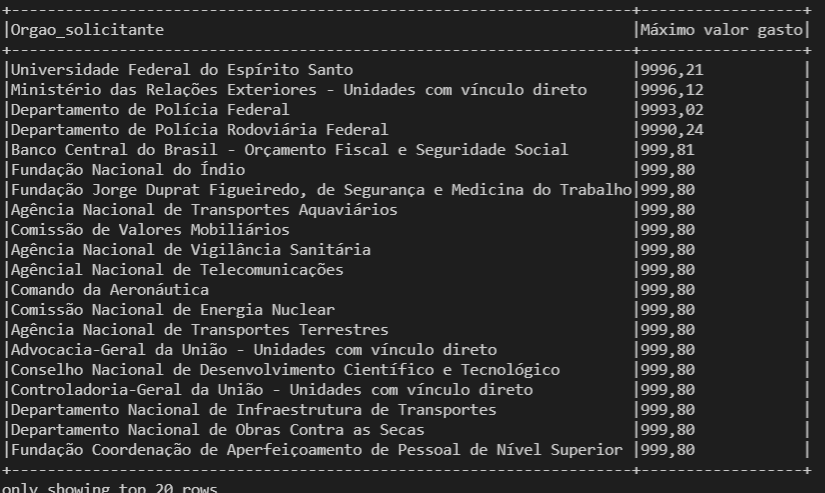
valor\_diaria\_max = viagens.groupBy("Orgao\_solicitante").agg(B.max("Valor\_diarias").alias("Máximo valor gasto"))

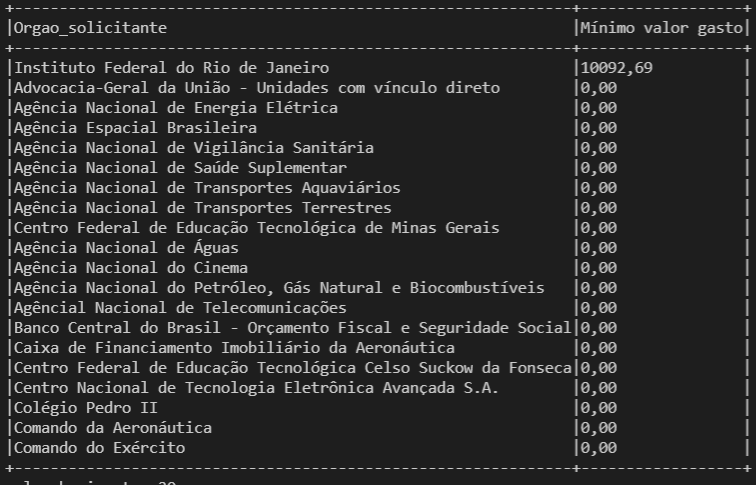
max\_valor\_diaria = valor\_diaria\_max.orderBy(B.desc("Máximo valor gasto")).show(truncate=False)

valor\_diaria\_min = viagens.groupBy("Orgao\_solicitante").agg(B.min("Valor\_diarias").alias("Mínimo valor gasto"))

min\_valor\_diaria = valor\_diaria\_min.orderBy(B.desc("Mínimo valor gasto")).show(truncate=False)







# VALOR DAS PASSAGENS

valor\_passagem\_soma= viagens.groupBy("Orgao\_solicitante").agg(B.sum("Valor\_passagens").alias("Total valor gasto em passagens"))

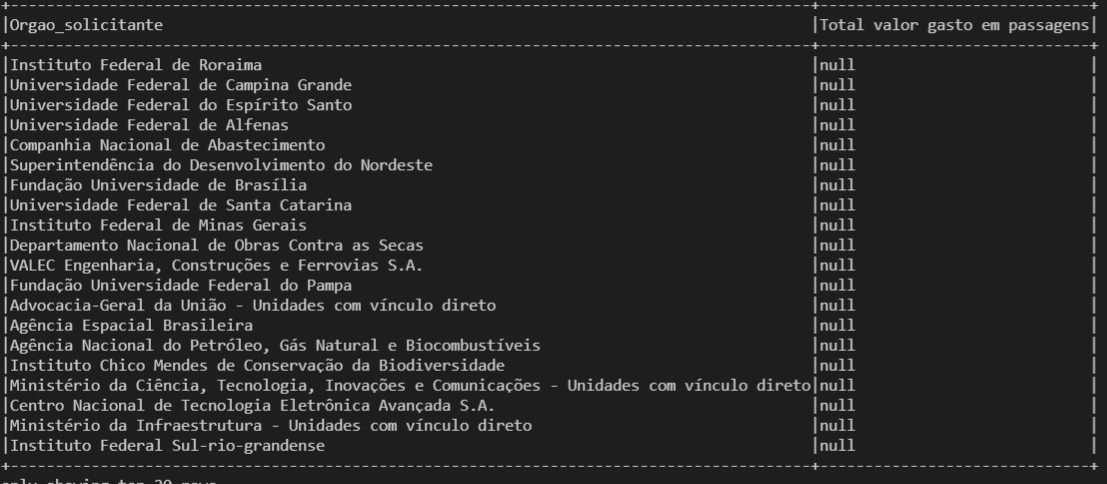
soma\_passagem= valor\_passagem\_soma.orderBy(B.desc("Total valor gasto em passagens")).show(truncate=False)

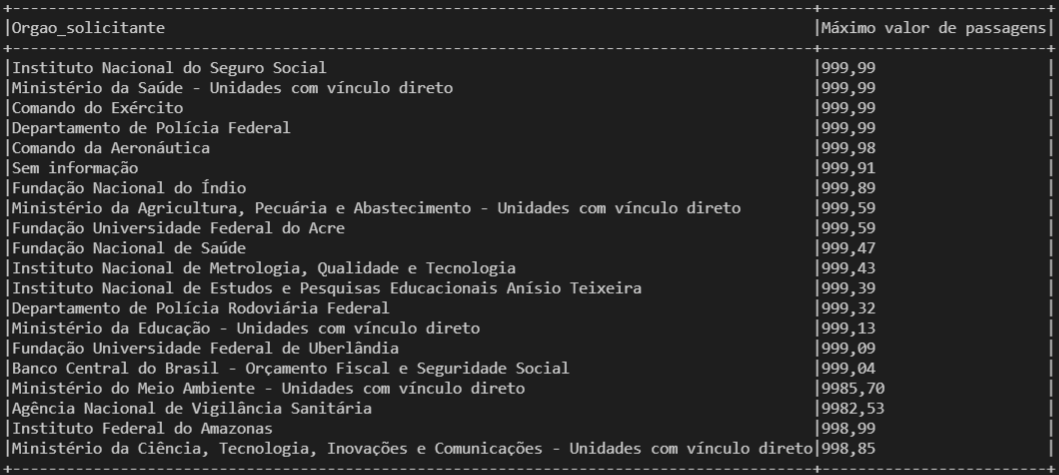
valor\_passagem\_max = viagens.groupBy("Orgao\_solicitante").agg(B.max("Valor\_passagens").alias("Máximo valor de passagens"))

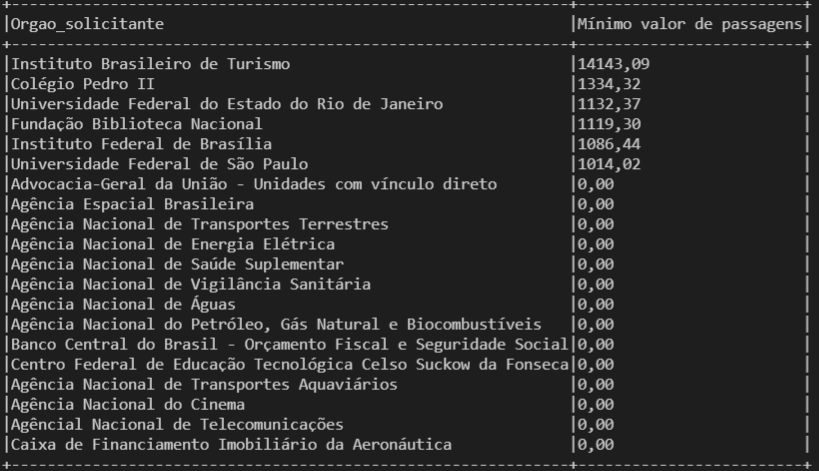
max\_passagem = valor\_passagem\_max.orderBy(B.desc("Máximo valor de passagens")).show(truncate=False)

valor\_passagem\_min = viagens.groupBy("Orgao\_solicitante").agg(B.min("Valor\_passagens").alias("Mínimo valor de passagens"))

min\_passagem = valor\_passagem\_min.orderBy(B.desc("Mínimo valor de passagens")).show(truncate=False)







# OUTROS GASTOS

outros\_gastos\_soma= viagens.groupBy("Orgao\_solicitante").agg(B.sum("Outros\_gastos").alias("Total valor gasto em outros gastos"))

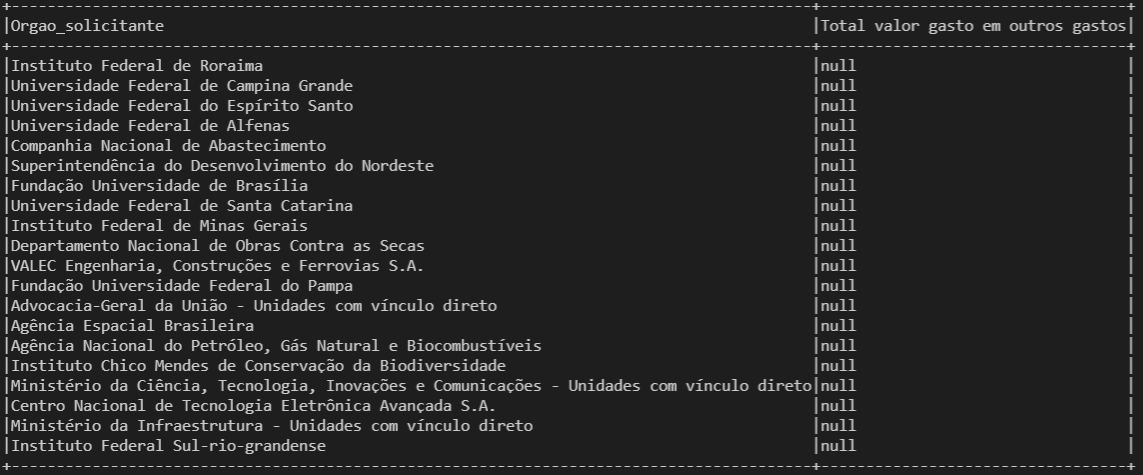
soma\_outros\_gastos = outros\_gastos\_soma.orderBy(B.desc("Total valor gasto em outros gastos")).show(truncate=False)

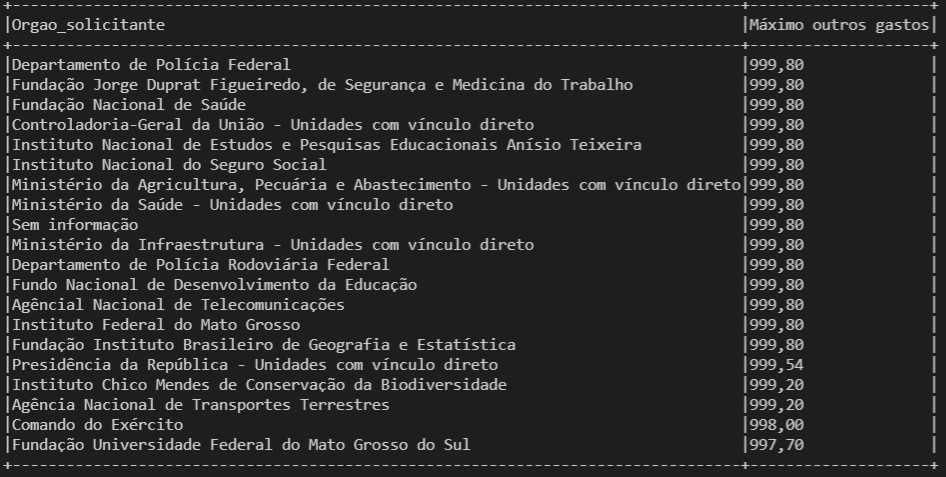
outros\_gastos\_max = viagens.groupBy("Orgao\_solicitante").agg(B.max("Outros\_gastos").alias("Máximo outros gastos"))

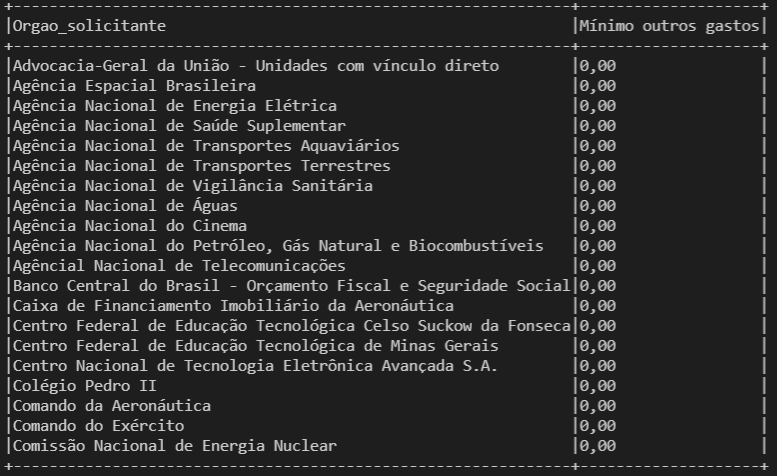
max\_outros\_gastos = outros\_gastos\_max.orderBy(B.desc("Máximo outros gastos")).show(truncate=False)

outros\_gastos\_min = viagens.groupBy("Orgao\_solicitante").agg(B.min("Outros\_gastos").alias("Mínimo outros gastos"))

min\_outros\_gastos = outros\_gastos\_min .orderBy(B.desc("Mínimo outros gastos")).show(truncate=False)

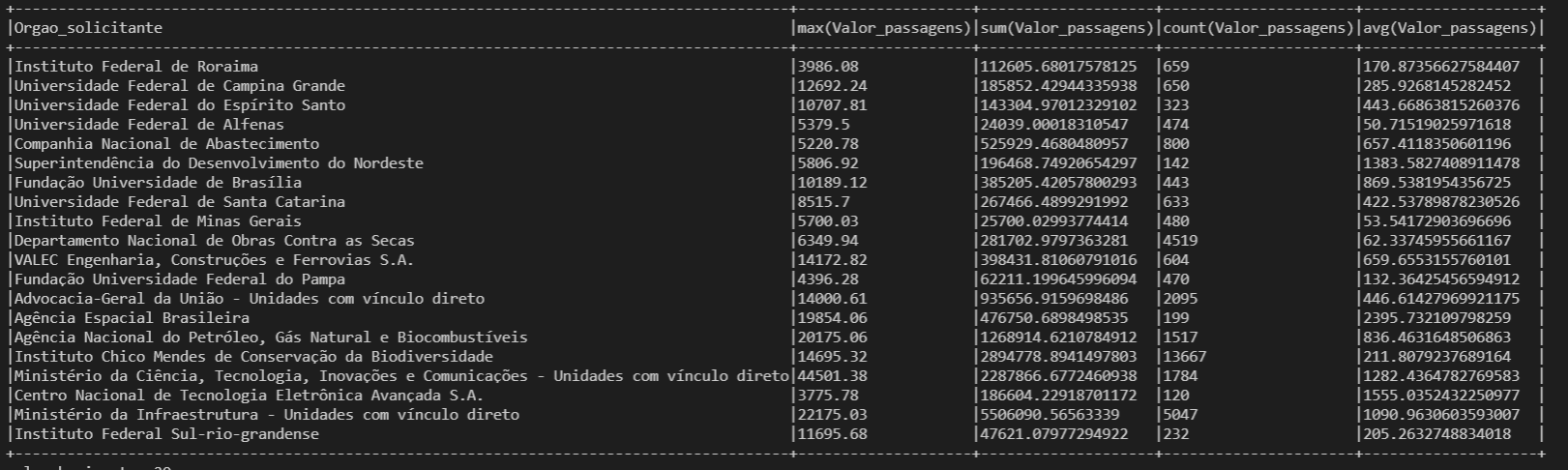






# ESTATISTICA DOS GASTOS DOS ORGAOS SOLICITANTES

viagens.groupBy("Orgao\_solicitante").agg(B.max("Valor\_passagens"),B.sum("Valor\_passagens"), B.count("Valor\_passagens"), B.avg("Valor\_passagens")).show(truncate=False)



# FILTRANDO DADOS ESPECIFICO = COMANDO DO EXERCITO

dados = viagens.filter(viagens["Orgao\_solicitante"] == "Comando do Exército")

dados.select("Situacao","Motivo","Destinos","Valor\_diarias","Valor\_passagens","Outros\_gastos").show(19,truncate=False)

dados.printSchema()

