import sys

from awsglue.transforms import \*

from awsglue.utils import getResolvedOptions

from pyspark.context import SparkContext

from awsglue.context import GlueContext

from awsglue.job import Job

## @params: [JOB\_NAME]

args = getResolvedOptions(sys.argv, ['JOB\_NAME'])

sc = SparkContext()

glueContext = GlueContext(sc)

spark = glueContext.spark\_session

job = Job(glueContext)

job.init(args['JOB\_NAME'], args)

CustomersCurated\_node = glueContext.create\_dynamic\_frame.from\_options(

format\_options= {"multiline":True},

connection\_type="s3",

format="json",

connection\_options={

"paths":["s3://stedi-supraja/customer/curated/"],

"recurse":True,

},

transformation\_ctx = "CustomersCurated\_node",

)

StepTrainerLanding\_node1 = glueContext.create\_dynamic\_frame.from\_options(

format\_options={"multiline":True},

connection\_type="s3",

format="json",

connection\_options={

"paths":["s3://stedi-supraja/step\_trainer/landing"],

"recurse":True,

},

transformation\_ctx="StepTrainerLanding\_node1",

)

RenamedKeysforJoi\_node = ApplyMapping.apply(

frame=CustomersCurated\_node,

mappings=[

("serialNumber","string","right\_serialNumber","string"),

("birthDay","string","right\_birthDay","string"),

("shareWithResearchAsOfDate","bigint","shareWithResearchAsOfDate","long"),

("registrationDate","bigint","registrationDate","long"),

("customerName","string","right\_customerName","string"),

("email","string","right\_email","string"),

("lastUpdateAsOfDate","bigint","lastUpdateAsOfDate","long"),

("phone","string","right\_phone","string"),

("shareWithFriendsAsOfDate","bigint","shareWithFriendsAsOfDate","long"),

("shareWithPublicAsOfDate","bigint","shareWithPublicAsOfDate","long")

],

transformation\_ctx="RenamedKeysforJoi\_node" ,

)

Join\_node1 = Join.apply(

frame1=StepTrainerLanding\_node1,

frame2=RenamedKeysforJoi\_node,

keys1=["serialNumber"],

keys2=["right\_serialNumber"],

transformation\_ctx="Join\_node1",

)

DropFields\_node1 = DropFields.apply(

frame=Join\_node1,

paths=[

"right\_serialNumber",

"right\_birthDay",

"right\_customerName",

"right\_email",

"right\_phone",

],

transformation\_ctx="DropFields\_node1",

)

StepTrainerTrusted\_node2 = glueContext.write\_dynamic\_frame.from\_options(

frame=DropFields\_node1,

connection\_type="s3",

format="json",

connection\_options={

"path": "s3://stedi-supraja/step\_trainer/trusted/",

"partitionKeys": [],

},

transformation\_ctx="StepTrainerTrusted\_node2",

)

job.commit()