testcase27_csv_csv_bigdata_match
content comparison

1. Run Summary

Application Name	:sample
Protocol Name	:DBTablesDemo
Protocol File Path	:/app/test/testprotocol/testprotocol.xlsx
Testcase Name	:testcase27_csv_csv_bigdata_match
Testcase Type	:content
Test Environment	:dev
Start Time	:28-Jul-2023 14:12:24 UTC
End Time	:28-Jul-2023 14:17:41 UTC
Run Time	:0:05:16
Test Result	:Passed
Reason	:Content matched

2. Configuration Details

Compare Type	:s2tcompare
testquerygenerationmode	:Auto
Testcase Type	:content
Source Connection Type	:aws-s3
Source Format	:delimitedfile
Source Path	:/app/test/data/source/ar_properties.csv
Source Exclude Columns	:start_date, end_date,operation_type,description
Target Connection Type	:aws-s3
Target Format	:delimitedfile
Target Path	:/app/test/data/target/ar_properties.csv
Target Exclude Columns	:start_date, end_date,operation_type,description
S2T Path	:/app/test/s2t/s2t_27_csv_csv_bigdata_match.xlsx
Primary Keys	:id,created_on

3. Content Summary

Test Result	:Passed
No. of rows in Source	:3,380,253
No. of distinct rows in Source	:1,799,578
No. of duplicate rows in Source	:1,580,675
No. of rows in Target	:3,380,253
No. of distinct rows in Target	:1,799,578
No. of duplicate rows in Target	:1,580,675
No. of matched rows	:1,799,578
No. of mismatched rows	:0
No. of rows in Source but not in Target	:0
No. of rows in Target but not in Source	:0

4. SQL Queries

4.1 Source Query

```
readdatadf=spark.read.format('delimitedfile').option('delimiter',,).option('header','true').load('/app/test/data/source/ar_properties.csv')

readdatadf.createOrReplaceTempView('dataview')

spark.sql("SELECT src.ad_type as ad_type, src.bathrooms as bathrooms, src.bedrooms as bedrooms, src.created_on as created_on, src.currency as currency, src.end_date as end_date, src.id as id, src.l1 as l1, src.l2 as l2, src.l3 as l3, src.l4 as l4, src.l5 as l5, src.l6 as l6, src.lat as lat, src.lon as lon, src.price as price, src.price_period as price_period, src.property_type as property_type, src.rooms as rooms, src.surface_covered as surface_covered, src.surface_total as surface_total, src.title as title FROM dataview src  ")
```

4.2 Target Query

```
readdatadf=spark.read.format('delimitedfile').option('delimiter',,).option('header','true').load('/app/test/data/target/ar_properties.csv')

readdatadf.createOrReplaceTempView('dataview')

spark.sql("SELECT ad_type, bathrooms, bedrooms, created_on, currency, end_date, id, l1, l2, l3, l4, l5, l6, lat, lon, price, price_period, property_type, rooms, surface_covered, surface_total, title FROM dataview tgt ")
```

5. Sample Mismatches 8 rows

5.1 Keys in source but not in target

None

5.2 Keys in target but not in source

None

5.3 Keys having one or more unequal column values

None

6. Columnwise Mismatch Summary

None

7. Columnwise Mismatch Details

None