

Checking Spark Table in new session

February 4, 2025

```
[1]: from pyspark.sql import SparkSession
```

```
# Start a new session (same application)
spark = SparkSession.builder \
    .appName("TableVerification") \
    .enableHiveSupport() \
    .getOrCreate()
```

25/02/02 03:27:11 WARN SparkSession: Using an existing Spark session; only runtime SQL configurations will take effect.

```
[2]: sparn_new = spark.newSession()
```

```
[3]: # Temporary Table should NOT exist
print("### Querying Temporary Table (Should Fail) ###")
try:
    sparn_new.sql("SELECT * FROM temp_customers").show()
except Exception as e:
    print("Temporary Table is not accessible in a new session!")
```

Querying Temporary Table (Should Fail)

ivysettings.xml file not found in HIVE_HOME or
HIVE_CONF_DIR,/etc/hive/conf.dist/ivysettings.xml will be used

Temporary Table is not accessible in a new session!

```
[4]: sparn_new
```

```
[4]: <pyspark.sql.session.SparkSession at 0x7f66779021d0>
```

```
[4]: # Global Temporary Table should exist
print("### Querying Global Temporary Table ###")
spark.sql("SELECT * FROM global_temp.global_customers").show()
```

Querying Global Temporary Table

```
-----
AnalysisException                                Traceback (most recent call last)
/tmp/ipykernel_12034/1652584772.py in <cell line: 3>()
```

```

1 # Global Temporary Table should exist
2 print("### Querying Global Temporary Table ###")
----> 3 spark.sql("SELECT * FROM global_temp.global_customers").show()
4
5

/usr/lib/spark/python/pyspark/sql/session.py in sql(self, sqlQuery, **kwargs)
1032         sqlQuery = formatter.format(sqlQuery, **kwargs)
1033         try:
-> 1034         return DataFrame(self._jsparkSession.sql(sqlQuery), self)
1035         finally:
1036             if len(kwargs) > 0:

/usr/lib/spark/python/lib/py4j-0.10.9.5-src.zip/py4j/java_gateway.py in
-> __call__(self, *args)
1319
1320         answer = self.gateway_client.send_command(command)
-> 1321         return_value = get_return_value(
1322             answer, self.gateway_client, self.target_id, self.name)
1323

/usr/lib/spark/python/pyspark/sql/utils.py in deco(*a, **kw)
194         # Hide where the exception came from that shows a
-> non-Pythonic
195         # JVM exception message.
--> 196         raise converted from None
197         else:
198             raise

AnalysisException: Table or view not found: global_temp.global_customers; line
-> pos 14;
'Project [*]
+- 'UnresolvedRelation [global_temp, global_customers], [], false

```

```

[5]: print("### Querying Global Temporary Table ###")
      spark_new.sql("SHOW TABLES IN global_temp").show()

```

```

### Querying Global Temporary Table ###
+-----+-----+-----+
|namespace|tableName|isTemporary|
+-----+-----+-----+
+-----+-----+-----+

```

```
[6]: # Persistent Table should exist
print("### Querying Persistent Table ###")
sparn_new.sql("SELECT * FROM customers_persistent").show()
```

Querying Persistent Table

[Stage 0:> (0 + 1) / 1]

```
+-----+-----+-----+-----+-----+-----+-----+
--+
|customer_id|      name|      city|
state|country|registration_date|is_active|
+-----+-----+-----+-----+-----+-----+-----+
--+
|          0| Customer_0|      Pune|Maharashtra|  India|      2023-06-29|
False|
|          1| Customer_1|Bangalore|  Tamil Nadu|  India|      2023-12-07|
True|
|          2| Customer_2|Hyderabad|    Gujarat|  India|      2023-10-27|
True|
|          3| Customer_3|Bangalore|  Karnataka|  India|      2023-10-17|
False|
|          4| Customer_4|Ahmedabad|  Karnataka|  India|      2023-03-14|
False|
|          5| Customer_5|Hyderabad|  Karnataka|  India|      2023-07-28|
False|
|          6| Customer_6|      Pune|      Delhi|  India|      2023-08-29|
False|
|          7| Customer_7|Ahmedabad|West Bengal|  India|      2023-12-28|
True|
|          8| Customer_8|      Pune|  Karnataka|  India|      2023-06-22|
True|
|          9| Customer_9|  Mumbai|  Telangana|  India|      2023-01-05|
True|
|         10|Customer_10|      Pune|    Gujarat|  India|      2023-08-05|
True|
|         11|Customer_11|      Delhi|West Bengal|  India|      2023-08-02|
False|
|         12|Customer_12|  Chennai|    Gujarat|  India|      2023-11-21|
False|
|         13|Customer_13|  Chennai|  Karnataka|  India|      2023-11-06|
True|
|         14|Customer_14|Hyderabad|  Tamil Nadu|  India|      2023-02-07|
False|
|         15|Customer_15|  Mumbai|    Gujarat|  India|      2023-03-02|
True|
|         16|Customer_16|  Chennai|  Karnataka|  India|      2023-04-05|
False|
|         17|Customer_17|Hyderabad|West Bengal|  India|      2023-08-21|
```

```
False|
|      18|Customer_18|      Pune|      Delhi|  India|      2023-10-04|
True|
|      19|Customer_19|  Kolkata|      Gujarat|  India|      2023-02-05|
True|
+-----+-----+-----+-----+-----+-----+-----+
--+
only showing top 20 rows
```

```
[8]: sparn_new.sql('show tables').show()
```

```
+-----+-----+-----+
|namespace|      tableName|isTemporary|
+-----+-----+-----+
| default|customers_persistent|      false|
+-----+-----+-----+
```

```
[9]: sparn_new.stop()
```