

```
11:34 AM (<1s) 55 Python
sample = TEST.withColumn("decimal_value", col("string_value").cast(DecimalType(12,0)))

sample: pyspark.sql.dataframe.DataFrame
  string_value: string
  decimal_value: decimal(12,0)
```

```
Last execution failed 56 Python
1 sample.display()

Py4JJavaError: An error occurred while calling t.addCustomDisplayData.
: java.sql.SQLException: Status of query associated with resultSet is FAILED_WITH_ERROR. Numeric value 'WVL383' is not recognized Results not generated.

at net.snowflake.client.jdbc.SFAsyncResultSet.getRealResults(SFAsyncResultSet.java:138)
at net.snowflake.client.jdbc.SFAsyncResultSet.getResultSetSerializables(SFAsyncResultSet.java:375)
at net.snowflake.spark.snowflake.SnowflakeRelation.$anonfun$getSnowflakeResultSetRDD$2(SnowflakeRelation.scala:283)
at net.snowflake.spark.snowflake.AsyncExecutionThread.run(AsyncExecutionThread.scala:53)

File <command-234779613005257>, line 1
----> 1 sample.display()
^
^
File /databricks/python_shell/dbruntime/monkey_patches.py:42, in apply_dataframe_display_patch.<locals>.df_display(df, *args, **kwargs)
38 def df_display(df, *args, **kwargs):
39     """
40     df.display() is an alias for display(df). Run help(display) for more information.
41     """
--> 42     display(df, *args, **kwargs)
File /databricks/python_shell/dbruntime/display.py:194, in Display.display(self, input, *args, **kwargs)
191     if kwargs.get('trigger'):
192         raise Exception('Triggers can only be set for streaming queries.')
--> 194     self.add_custom_display_data("table", input._jdf)
195 # This version is for Serverless + Spark Connect dogfooding.
196 elif self.spark_connect_enabled and isinstance(input, ConnectDataFrame):
```

```
File /databricks/python_shell/dbruntime/display.py:48, in Display.add_custom_display_data(self, data_type, data)
46 def add_custom_display_data(self, data_type, data):
47     custom_display_key = str(uuid.uuid4())
--> 48     return_code = self.entry_point.addCustomDisplayData(custom_display_key, data_type, data)
49     ip_display({
50         "application/vnd.databricks.v1+display": custom_display_key,
51         "text/plain": "<Databricks Output (not supported in output widgets)>"
52     },
53         raw=True)
54     if return_code == 1:
File /databricks/spark/python/lib/py4j-0.10.9.7-src.zip/py4j/java_gateway.py:1355, in JavaMember.__call__(self, *args)
1349 command = proto.CALL_COMMAND_NAME + \
1350     self.command_header + \
1351     args_command + \
1352     proto.END_COMMAND_PART
1354 answer = self.gateway_client.send_command(command)
-> 1355 return_value = get_return_value(
1356     answer, self.gateway_client, self.target_id, self.name)
1358 for temp_arg in temp_args:
1359     if hasattr(temp_arg, "_detach"):
File /databricks/spark/python/pyspark/errors/exceptions/captured.py:224, in capture_sql_exception.<locals>.deco(*a, **kw)
222 def deco(*a: Any, **kw: Any) -> Any:
223     try:
--> 224         return f(*a, **kw)
225     except Py4JJavaError as e:
226         converted = convert_exception(e.java_exception)
```

```
File /databricks/spark/python/lib/py4j-0.10.9.7-src.zip/py4j/protocol.py:326, in get_return_value(answer, gateway_client, target_id, name)
324 value = OUTPUT_CONVERTER[type](answer[2:], gateway_client)
325 if answer[1] == REFERENCE_TYPE:
--> 326     raise Py4JJavaError(
327         "An error occurred while calling {0}{1}{2}.\n".
328         format(target_id, ".", name), value)
329 else:
330     raise Py4JError(
331         "An error occurred while calling {0}{1}{2}. Trace:\n{3}\n".
332         format(target_id, ".", name), value)

Diagnose error
```