

```
In [1]: import findspark
findspark.init()
findspark.find()
import pyspark
findspark.find()
```

```
Out[1]: 'C:\\Users\\vinitha\\Desktop\\spark-3.0.0-bin-hadoop2.7'
```

```
In [3]: pyspark --packages io.delta:delta-core_2.12:1.2.1 --conf "spark.sql.extensions=io.d
```

```
Input In [3]
pyspark --packages io.delta:delta-core_2.12:1.2.1 --conf "spark.sql.extensions
=io.delta.sql.DeltaSparkSessionExtension" --conf "spark.sql.catalog.spark_catalog=
org.apache.spark.sql.delta.catalog.DeltaCatalog"
^
SyntaxError: invalid syntax
```

```
In [ ]:
```

```
In [4]: import pyspark
from delta import *

builder = pyspark.sql.Session.builder.appName("MyApp") \
    .config("spark.sql.extensions", "io.delta.sql.DeltaSparkSessionExtension") \
    .config("spark.sql.catalog.spark_catalog", "org.apache.spark.sql.delta.catalog")

spark = configure_spark_with_delta_pip(builder).getOrCreate()
```

```
In [5]: import pyspark
from delta import *
```

```
In [2]: #Initiate spark context
from pyspark import SparkContext, SparkConf

from pyspark.sql import SparkSession

session = SparkSession.builder.getOrCreate()
```

```
In [14]: pip install pyspark==3.2.0
```

```
Requirement already satisfied: pyspark==3.2.0 in c:\users\vinitha\anaconda3\lib\s
ite-packages (3.2.0)
Requirement already satisfied: py4j==0.10.9.2 in c:\users\vinitha\anaconda3\lib\s
ite-packages (from pyspark==3.2.0) (0.10.9.2)
Note: you may need to restart the kernel to use updated packages.
```

```
In [5]: df = spark.read.csv("C:\\Users\\vinitha\\Desktop\\seleniumDemo\\people.csv", inferS
```

```
In [6]: df.show()
```

```

+-----+-----+-----+-----+-----+-----+-----+
|person_ID|          name|   first|   last| middle|          email|
|phone|          fax|          title|
+-----+-----+-----+-----+-----+-----+-----+
|    3130|    Burks, Rosella | Rosella|  Burks|   null|    BurksR@univ.edu|
963.555.1253|963.777.4065|    Professor |
|    3297|    Avila, Damien |  Damien|  Avila|   null|    AvilaD@univ.edu|
963.555.1352|963.777.7914|    Professor |
|    3547|    Olsen, Robin |  Robin|  Olsen|   null|    OlsenR@univ.edu|
963.555.1378|963.777.9262|Assistant Professor|
|    1538| Moises, Edgar Estes|  Edgar|  Moises|  Estes|    MoisesE@univ.edu|96
3.555.2731x3565|963.777.8264|    Professor|
|    2941| Brian, Heath Pruitt|  Heath|  Brian| Pruitt|    BrianH@univ.edu|
963.555.2800|963.777.7249| Associate Curator |
|    2401| Claude, Elvin Haney|  Elvin|  Claude|  Haney|    ClaudeE@univ.edu|
963.555.2902|963.777.9730|    Curator|
|    2070|    Mosley, Edmund | Edmund|  Mosley|   null|    MosleyE@univ.edu|
963.555.2945|963.777.9285|Assistant Professor|
|    2561|Derek, Antoine Mccoy| Antoine|  Derek| Mccoy|    DerekA@univ.edu|
963.555.2992|963.777.5454|    Curator|
|    1625|    Hawkins, Callie | Callie|  Hawkins|   null|    HawkinsC@univ.edu|96
3.555.3350x6480|963.777.4949|    Professor|
|    1307|    Pate, Andrea |  Andrea|  Pate|   null|    PateA@univ.edu|
963.555.3723|963.777.5049|    Professor|
|    2342|    Austin, Liz |  Liz|  Austin|   null|    AustinL@univ.edu|
963.555.4305|963.777.6143|Assistant Professor|
|    2755|Kendrick, Reba Al...|  Reba|Kendrick| Alford|KendrickR@univ.edu|96
3.555.4618x7744|963.777.4371| Associate Curator |
|    4150|    Sims, Angelina | Angelina|  Sims|   null|    SimsA@univ.edu|
963.555.5278|963.777.4400|    Professor|
|    3544| Mullins, Kimberly | Kimberly| Mullins|   null|    MullinsK@univ.edu|96
3.555.5471x1017|963.777.9783|Assistant Professor|
|    2096| Chuck, Lloyd Haney|  Lloyd|  Chuck|  Haney|    ChuckL@univ.edu|96
3.555.5568x2652|963.777.9290|Assistant Professor|
|    1089|    Payne, Ladonna | Ladonna|  Payne|   null|    PayneL@univ.edu|
963.555.5639|963.777.6469|    Professor|
|    2948|Baxter, Johnathan...|Johnathan|  Baxter|Browning|    BaxterJ@univ.edu|
963.555.5902|963.777.8336| Research Professor|
|    4539|    Weiss, Gilbert | Gilbert|  Weiss|   null|    WeissG@univ.edu|
963.555.5969|963.777.4975|    Professor|
|    2811|Deirdre, Florence...| Florence| Deirdre| Barrera| DeirdreF@univ.edu|
963.555.6319|963.777.9654| Associate Curator |
|    4580|Fernando, Toby Ca...|  Toby|Fernando|Calderon|FernandoT@univ.edu|
963.555.6469|963.777.9864| Research Professor|
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+
only showing top 20 rows

```

```
In [7]: df.printSchema()
```

```

root
|-- person_ID: integer (nullable = true)
|-- name: string (nullable = true)
|-- first: string (nullable = true)
|-- last: string (nullable = true)
|-- middle: string (nullable = true)
|-- email: string (nullable = true)
|-- phone: string (nullable = true)
|-- fax: string (nullable = true)
|-- title: string (nullable = true)

```

```
In [8]: #Write the data in the above dataframe to local filesystem in delta format  
df.write.format("delta").saveAsTable("default.parent_demo1")
```

```

-----
Py4JJavaError                                Traceback (most recent call last)
Input In [8], in <cell line: 1>()
----> 1 df.write.format("delta").saveAsTable("default.parent_demo1")

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\pyspark\sql\readwriter.py:868, in
DataFrameWriter.saveAsTable(self, name, format, mode, partitionBy, **options)
    866 if format is not None:
    867     self.format(format)
--> 868 self._jwrite.saveAsTable(name)

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\lib\py4j-0.10.9-src.zip\py4j\java_
gateway.py:1304, in JavaMember.__call__(self, *args)
    1298 command = proto.CALL_COMMAND_NAME + \
    1299     self.command_header + \
    1300     args_command + \
    1301     proto.END_COMMAND_PART
    1303 answer = self.gateway_client.send_command(command)
-> 1304 return value = get_return_value(
    1305     answer, self.gateway_client, self.target_id, self.name)
    1307 for temp_arg in temp_args:
    1308     temp_arg._detach()

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\pyspark\sql\utils.py:131, in captu
re_sql_exception.<locals>.deco(*a, **kw)
    129 def deco(*a, **kw):
    130     try:
--> 131         return f(*a, **kw)
    132     except py4j.protocol.Py4JJavaError as e:
    133         converted = convert_exception(e.java_exception)

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\lib\py4j-0.10.9-src.zip\py4j\proto
col.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))

Py4JJavaError: An error occurred while calling o39.saveAsTable.
: java.lang.ClassNotFoundException: Failed to find data source: delta. Please find
packages at http://spark.apache.org/third-party-projects.html
    at org.apache.spark.sql.execution.datasources.DataSource$.lookupDataSource
(DataSource.scala:674)
    at org.apache.spark.sql.execution.datasources.DataSource$.lookupDataSource
V2(DataSource.scala:728)
    at org.apache.spark.sql.DataFrameWriter.lookupV2Provider(DataFrameWriter.s
cala:948)
    at org.apache.spark.sql.DataFrameWriter.saveAsTable(DataFrameWriter.scala:
573)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
    at java.lang.reflect.Method.invoke(Unknown Source)
    at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:244)
    at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:357)
    at py4j.Gateway.invoke(Gateway.java:282)
    at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:132)
    at py4j.commands.CallCommand.execute(CallCommand.java:79)
    at py4j.GatewayConnection.run(GatewayConnection.java:238)

```

```
    at java.lang.Thread.run(Unknown Source)
Caused by: java.lang.ClassNotFoundException: delta.DefaultSource
    at java.net.URLClassLoader.findClass(Unknown Source)
    at java.lang.ClassLoader.loadClass(Unknown Source)
    at java.lang.ClassLoader.loadClass(Unknown Source)
    at org.apache.spark.sql.execution.datasources.DataSource$.anonfun$lookupD
ataSource$5(DataSource.scala:648)
    at scala.util.Try$.apply(Try.scala:213)
    at org.apache.spark.sql.execution.datasources.DataSource$.anonfun$lookupD
ataSource$4(DataSource.scala:648)
    at scala.util.Failure.orElse(Try.scala:224)
    at org.apache.spark.sql.execution.datasources.DataSource$.lookupDataSource
(DataSource.scala:648)
    ... 14 more
```

```
In [9]: #Read the above data written in the delta format into dataframe
df = spark.read.format("delta").load("/tmp/delta-table")
df.show()
```

```

-----
Py4JJavaError                                Traceback (most recent call last)
Input In [9], in <cell line: 1>()
----> 1 df = spark.read.format("delta").load("/tmp/delta-table")
      2 df.show()

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\pyspark\sql\readwriter.py:178, in
DataFrameReader.load(self, path, format, schema, **options)
    176 self.options(**options)
    177 if isinstance(path, basestring):
--> 178     return self._df(self._jreader.load(path))
    179 elif path is not None:
    180     if type(path) != list:

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\lib\py4j-0.10.9-src.zip\py4j\java_
gateway.py:1304, in JavaMember.__call__(self, *args)
    1298 command = proto.CALL_COMMAND_NAME + \
    1299     self.command_header + \
    1300     args_command + \
    1301     proto.END_COMMAND_PART
    1303 answer = self.gateway_client.send_command(command)
-> 1304 return_value = get_return_value(
    1305     answer, self.gateway_client, self.target_id, self.name)
    1307 for temp_arg in temp_args:
    1308     temp_arg._detach()

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\pyspark\sql\utils.py:131, in captu
re_sql_exception.<locals>.deco(*a, **kw)
    129 def deco(*a, **kw):
    130     try:
--> 131         return f(*a, **kw)
    132     except py4j.protocol.Py4JJavaError as e:
    133         converted = convert_exception(e.java_exception)

File ~\Desktop\spark-3.0.0-bin-hadoop2.7\python\lib\py4j-0.10.9-src.zip\py4j\proto
col.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))

Py4JJavaError: An error occurred while calling o48.load.
: java.lang.ClassNotFoundException: Failed to find data source: delta. Please find
packages at http://spark.apache.org/third-party-projects.html
    at org.apache.spark.sql.execution.datasources.DataSource$.lookupDataSource
(DataSource.scala:674)
    at org.apache.spark.sql.execution.datasources.DataSource$.lookupDataSource
V2(DataSource.scala:728)
    at org.apache.spark.sql.DataFrameReader.load(DataFrameReader.scala:230)
    at org.apache.spark.sql.DataFrameReader.load(DataFrameReader.scala:214)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
    at java.lang.reflect.Method.invoke(Unknown Source)
    at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:244)
    at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:357)
    at py4j.Gateway.invoke(Gateway.java:282)
    at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:132)
    at py4j.commands.CallCommand.execute(CallCommand.java:79)

```

```
    at py4j.GatewayConnection.run(GatewayConnection.java:238)
    at java.lang.Thread.run(Unknown Source)
Caused by: java.lang.ClassNotFoundException: delta.DefaultSource
    at java.net.URLClassLoader.findClass(Unknown Source)
    at java.lang.ClassLoader.loadClass(Unknown Source)
    at java.lang.ClassLoader.loadClass(Unknown Source)
    at org.apache.spark.sql.execution.datasources.DataSource$.anonfun$lookupD
ataSource$5(DataSource.scala:648)
    at scala.util.Try$.apply(Try.scala:213)
    at org.apache.spark.sql.execution.datasources.DataSource$.anonfun$lookupD
ataSource$4(DataSource.scala:648)
    at scala.util.Failure.orElse(Try.scala:224)
    at org.apache.spark.sql.execution.datasources.DataSource$.lookupDataSource
(DataSource.scala:648)
    ... 14 more
```