Introducing Spark SQL

Reading and Writing Files

```
In: geoip_df
```

```
Out: DataFrame[ip: string, code: string, country: string]
```

```
In: geoip_df.show(3)
```

```
+---------------+----+------------------+
|             ip|code|           country|
+---------------+----+------------------+
|194.120.126.123|  NL|       Netherlands|
| 94.126.119.173|  FR|            France|
|  193.46.74.166|  RU|Russian Federation|
+---------------+----+------------------+
only showing top 3 rows
```

```
In: geoip_df.createTempView("geoip")
```

```
In: geoip_df.createTempView("geoip")
```

```
In: spark_session.sql("""
        create table web.geoip as
        select *
        from geoip
    """)
```

```
Out: DataFrame[]
```

```
In: spark_session.sql("""
        show tables in web
    """).show()
```

```
Out: +--------+----------+-----------+
     |database| tableName|isTemporary|
     +--------+----------+-----------+
     |     web|access_log|      false|
     |     web|     geoip|      false|
     |        |     geoip|       true|
     +--------+----------+-----------+
In:  geoip_df.write

Out: <pyspark.sql.readwriter.DataFrameWriter at
     0x7f60ef4b3810>
```

```
In: geoip_df.write
```

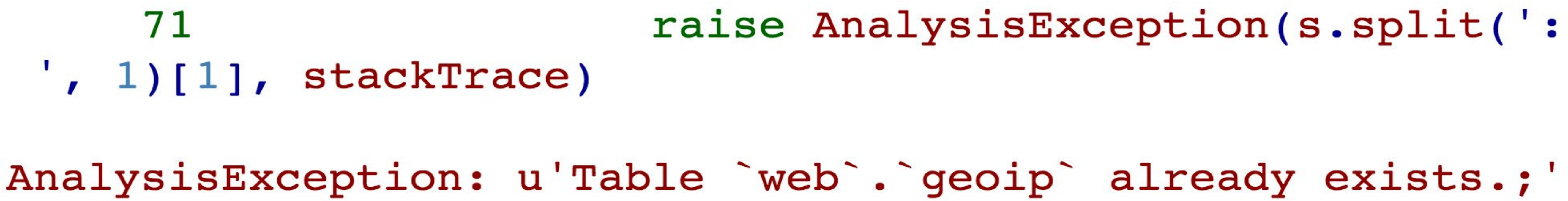
```
Out: <pyspark.sql.readwriter.DataFrameWriter at
0x7f60ef4b3810>
```

```
In: geoip_df.write.saveAsTable("web.geoip")
```

```
In: geoip_df.write

Out: <pyspark.sql.readwriter.DataFrameWriter at
     0x7f60ef4b3810>
```

```
In: geoip_df.write.saveAsTable("web.geoip")
```



```
     71                           raise AnalysisException(s.split(':
 ', 1)[1], stackTrace)

AnalysisException: u'Table `web`.`geoip` already exists.;'
```

```
In: geoip_df.write.saveAsTable("web.geoip_write")
```

```
write.saveAsTable(name, mode=None, ...)
write.saveAsTable(name, mode=None, ...)
```

"error"

```
write.saveAsTable(name, mode=None, ...)
```

"error"

"overwrite"

```
write.saveAsTable(name, mode=None, ...)
```

"error"
"overwrite"
"append"

```
write.saveAsTable(name, mode=None, ...)
```

"error"
"overwrite"
"append"

```
In: geoip_df.write.saveAsTable("web.geoip",
                                mode='overwrite')
```

```
write.saveAsTable(name, mode=None, ...)
```

"error"

"overwrite"

"append"

```
In: geoip_df.write.saveAsTable("web.geoip",
                                            mode='overwrite')
```

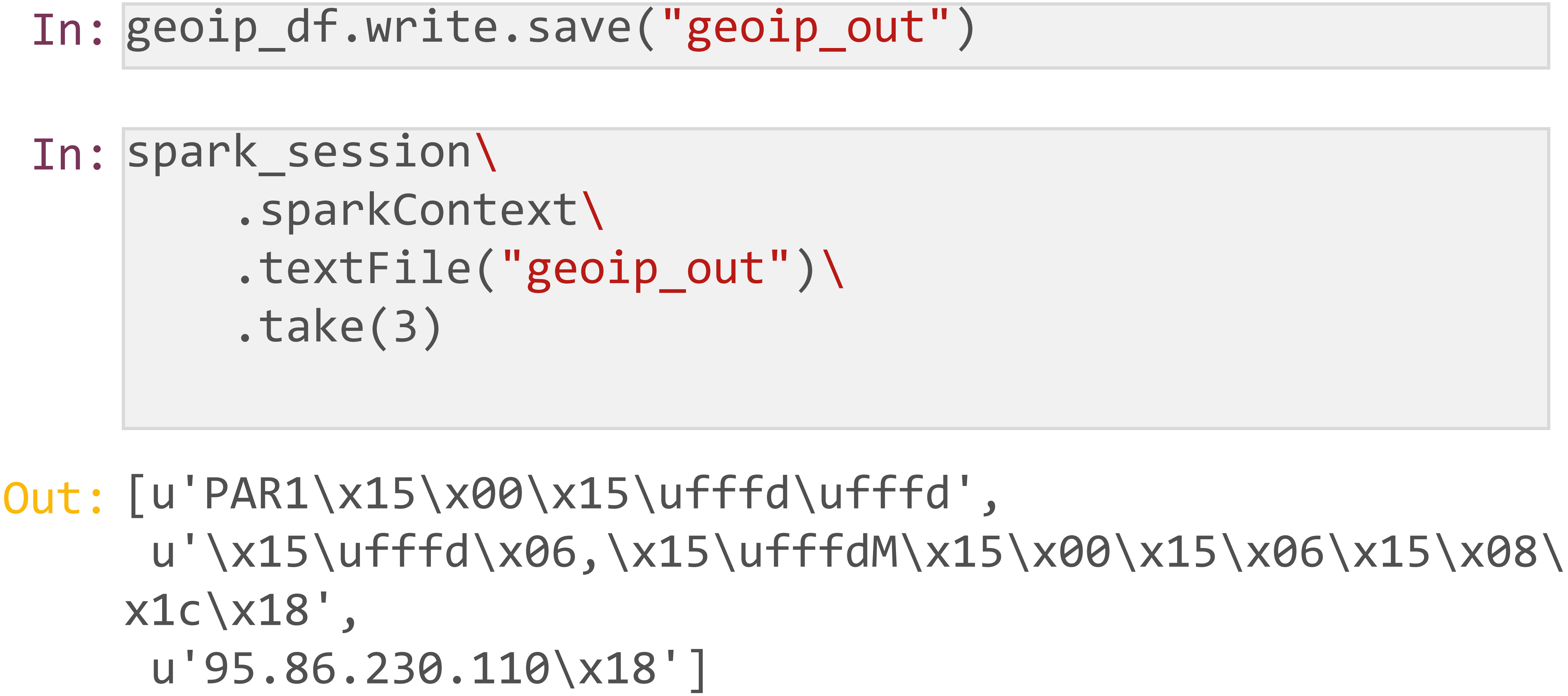
```
In: spark_session.sql("""
        select count(*)
        from web.geoip
    """).show()
```

```
+--------+
|count(1)|
+--------+
|    9910|
+--------+
In: geoip_df.write.saveAsTable("web.geoip",
                               mode='append')
```

In:
```
geoip_df.write.saveAsTable("web.geoip",
                                             mode='append')
```

In:
```
spark_session.sql("""
    select count(*)
    from web.geoip
""").show()
```

```
+--------+
|count(1)|
+--------+
|   19820|
+--------+
In: geoip_df.write.save("geoip_out")
```



```
In: geoip_df.write.save("geoip_out")
```

```
In: spark_session\
        .sparkContext\
        .textFile("geoip_out")\
        .take(3)
```

```
Out: [u'PAR1\x15\x00\x15\ufffd\ufffd',
      u'\x15\ufffd\x06,\x15\ufffdM\x15\x00\x15\x06\x15\x08\
     x1c\x18',
      u'95.86.230.110\x18']
```

In:
```
geoip_df.write.save("geoip_out")
```

In:
```
spark_session\
    .sparkContext\
    .textFile("geoip_out")\
    .take(3)
```

Out:
```
[u'PAR1\x15\x00\x15\ufffd\ufffd',
 u'\x15\ufffd\x06,\x15\ufffdM\x15\x00\x15\x06\x15\x08\
x1c\x18',
 u'95.86.230.110\x18']
```

```
In: geoip_df.write.save("geoip_out")
```

```
In: spark_session\
        .sparkContext\
        .textFile("geoip_out")\
        .take(3)
```

```
Out: [u'PAR1\x15\x00\x15\ufffd\ufffd',
      u'\x15\ufffd\x06,\x15\ufffdM\x15\x00\x15\x06\x15\x08\
     x1c\x18',
      u'95.86.230.110\x18']
```

Parquet

```
In: geoip_df.write.save("geoip_csv",
                        format='csv')
```

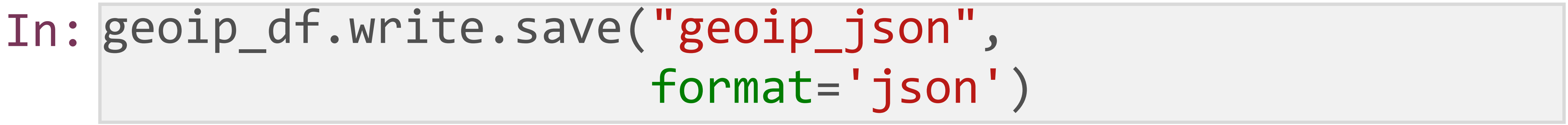
```
In: geoip_df.write.save("geoip_csv",
                        format='csv')
```

```
In: spark_session\
        .sparkContext\
        .textFile("geoip_csv")\
        .take(3)
```

```
Out: [u'194.120.126.123,NL,Netherlands',
      u'94.126.119.173,FR,France',
      u'193.46.74.166,RU,Russian Federation']
```

{JSON}

{JSON}



```
In: geoip_df.write.save("geoip_json",
                        format='json')
```

{JSON}

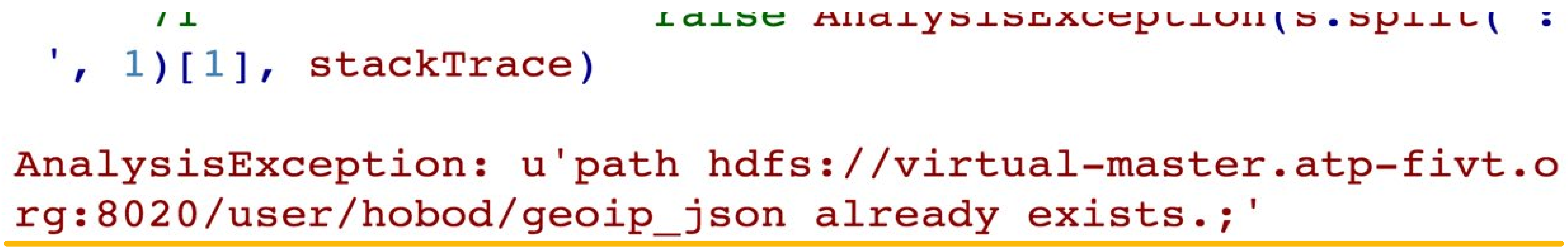
```
In: geoip_df.write.save("geoip_json",
                         format='json')
```

```
In: spark_session\
        .sparkContext\
        .textFile("geoip_json")\
        .take(3)
```

```
Out: [u'{"ip":"194.120.126.123","code":"NL","country":"Net
     herlands"}',
      u'{"ip":"94.126.119.173","code":"FR","country":"Fran
     ce"}',
      u'{"ip":"193.46.74.166","code":"RU","country":"Russi
     an Federation"}']
```

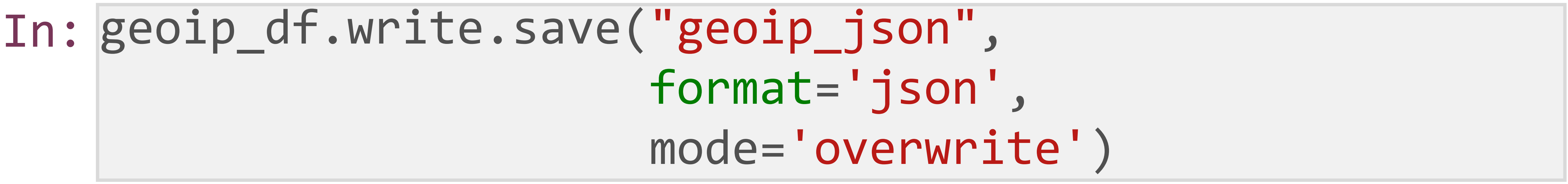
```
In: geoip_df.write.save("geoip_json",
                         format='json')
```

```
In: geoip_df.write.save("geoip_json",
                        format='json')
```



```
     71                    raise AnalysisException(s.split(':
 ', 1)[1], stackTrace)

AnalysisException: u'path hdfs://virtual-master.atp-fivt.o
rg:8020/user/hobod/geoip_json already exists.;'
```



```
In: geoip_df.write.save("geoip_json",
                        format='json',
                        mode='overwrite')
```

In:

```
geoip_df.write.parquet("geoip_parquete",
                    mode='overwrite')
```

```
In: geoip_df.write.parquet("geoip_parquete",
                        mode='overwrite')
```

```
In: geoip_df.write.csv("geoip_csv",
                        mode='overwrite')
```

```
In: geoip_df.write.parquet("geoip_parquete",
                        mode='overwrite')
```

```
In: geoip_df.write.csv("geoip_csv",
                        mode='overwrite')
```

```
In: geoip_df.write.save("geoip_json",
                        mode='overwrite')
```

```
In: spark_session.read
```

```
Out: <pyspark.sql.readwriter.DataFrameReader at
     0x7f60ef208190>
```

In:
```
spark_session.read
```

Out:
```
<pyspark.sql.readwriter.DataFrameReader at
0x7f60ef208190>
```

In:
```
geoip_from_table = spark_session\
    .read.table("web.geoip")
```

```
In: spark_session.read

Out: <pyspark.sql.readwriter.DataFrameReader at
     0x7f60ef208190>


In: geoip_from_table = spark_session\
        .read.table("web.geoip")


In: geoip_from_table.show(3)

    +---------------+----+------------------+
    |             ip|code|           country|
    +---------------+----+------------------+
    |194.120.126.123|  NL|       Netherlands|
    | 94.126.119.173|  FR|            France|
    |  193.46.74.166|  RU|Russian Federation|
    +---------------+----+------------------+
    only showing top 3 rows
```

```
In: geoip_from_json = spark_session\
         .read.json("geoip_json")
```

In:

```
geoip_from_json = spark_session\
    .read.json("geoip_json")
```

In:

```
geoip_from_json.show(3)
```

```
+----+------------------+---------------+
|code|           country|             ip|
+----+------------------+---------------+
|  NL|       Netherlands|194.120.126.123|
|  FR|            France| 94.126.119.173|
|  RU|Russian Federation|  193.46.74.166|
+----+------------------+---------------+
only showing top 3 rows
```

```
In: geoip_from_csv = spark_session\
        .read.csv("geoip_csv")
```

```
In: geoip_from_csv = spark_session\
        .read.csv("geoip_csv")
```

```
In: geoip_from_csv.show(3)
```

```
+--------------+---+------------------+
|           _c0|_c1|               _c2|
+--------------+---+------------------+
|   217.8.92.38| RU|Russian Federation|
|185.102.10.199| RU|Russian Federation|
|  217.73.57.80| RU|Russian Federation|
+--------------+---+------------------+
only showing top 3 rows
```

```
In: schema = StructType().add("ip",      StringType())\
                         .add("code",    StringType())\
                         .add("country", StringType())
```

```
In: schema = StructType().add("ip",      StringType())\
                         .add("code",    StringType())\
                         .add("country", StringType())
```

```
In: geoip_from_csv = spark_session\
        .read.csv("geoip_csv")
```

```
In: schema = StructType().add("ip",      StringType())\
                         .add("code",    StringType())\
                         .add("country", StringType())
```

```
In: geoip_from_csv = spark_session\
        .read.csv("geoip_csv")
```

```
In: geoip_from_csv.show(3)
```

```
+---------------+----+--------------------+
|             ip|code|             country|
+---------------+----+--------------------+
|    217.8.92.38|  RU|Russian Federation|
|185.102.10.199|  RU|Russian Federation|
|   217.73.57.80|  RU|Russian Federation|
+---------------+----+--------------------+
only showing top 3 rows
```

Parquet

Parquet

```
In: geoip_from_parquet = spark_session\
        .read.parquet("geoip_parquet")
```

Parquet

```
In: geoip_from_parquet = spark_session\
        .read.parquet("geoip_parquet")
```

```
In: geoip_from_parquet.show(3)
```

```
+---------------+----+------------------+
|             ip|code|           country|
+---------------+----+------------------+
|194.120.126.123|  NL|       Netherlands|
| 94.126.119.173|  FR|            France|
|  193.46.74.166|  RU|Russian Federation|
+---------------+----+------------------+
only showing top 3 rows
```

Microsoft SQL Server
PostgreSQL
ORACLE
MySQL

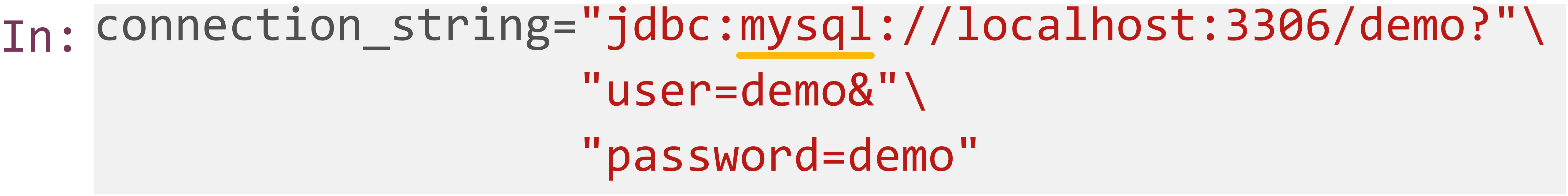
Microsoft SQL Server
PostgreSQL
ORACLE
MySQL
JDBC
Java Database Connectivity

JDBC - Java DataBase Connectivity

JDBC - Java DataBase Connectivity

```
In: connection_string="jdbc:mysql://localhost:3306/demo?"\
                      "user=demo&"\
                      "password=demo"
```

JDBC - Java DataBase Connectivity



```
In: connection_string="jdbc:mysql://localhost:3306/demo?"\
                      "user=demo&"\
                      "password=demo"
```

JDBC - Java DataBase Connectivity

```
In: connection_string="jdbc:mysql://localhost:3306/demo?"\
                      "user=demo&"\
                      "password=demo"
```

JDBC - Java DataBase Connectivity

```
In: connection_string="jdbc:mysql://localhost:3306/demo?"\
                      "user=demo&"\
                      "password=demo"
```

JDBC - Java DataBase Connectivity

```
In: connection_string="jdbc:mysql://localhost:3306/demo?"\
                      "user=demo&"\
                      "password=demo"
```

JDBC - Java DataBase Connectivity

```
In: connection_string="jdbc:mysql://localhost:3306/demo?"\
                      "user=demo&"\
                      "password=demo"
```

In:
```
geoip_from_jdbc = spark_session\
    .read.jdbc(connection_string, "geoip")
```

In:
```
geoip_from_jdbc.show(3)
```

```
+---------------+----+------------------+
|             ip|code|           country|
+---------------+----+------------------+
|194.120.126.123|  NL|       Netherlands|
| 94.126.119.173|  FR|            France|
|  193.46.74.166|  RU|Russian Federation|
+---------------+----+------------------+
only showing top 3 rows
```

```
In: geoip_df.write.jdbc(connection_string, "geoip")
```

```
In: geoip_from_jdbc = spark_session\
        .read.jdbc(connection_string, "geoip")
```

```
In: geoip_from_jdbc.show(3)
```

```
+---------------+----+------------------+
|             ip|code|           country|
+---------------+----+------------------+
|194.120.126.123|  NL|       Netherlands|
| 94.126.119.173|  FR|            France|
|  193.46.74.166|  RU|Russian Federation|
+---------------+----+------------------+
only showing top 3 rows
```

What have we learned:

- how to read/write tables by spark api methods
- how to read/write data from directories
- import and export data to any rdbms