


\_\_\_\_\_

```
$ hadoop fs -mkdir SPRK
$ hadoop fs -mkdir SPRK/Ex2
$ hadoop fs -copyFromLocal d311.csv SPRK/Ex2
$ hadoop fs -ls SPRK/Ex2

adminingroup 188659283 2024-05-01 04:23 SPRK/Ex2/d311.csv
$ █
```



```

Welcome to
version 3.5.0-amzn-0

Using Scala version 2.12.17 (OpenJDK 64-Bit Server VM, Java 17.0.10)
Type in expressions to have them evaluated.
Type :help for more information.

scala> val d311DF=spark.read.format("csv").option("header", "true").load("SPRK/Ex2/d311.csv")
d311DF: org.apache.spark.sql.DataFrame = [Unique Key: string, Created Date: string ... 36 more fields]

scala> d311DF.printSchema()
root
|-- Unique Key: string (nullable = true)
|-- Created Date: string (nullable = true)
|-- Closed Date: string (nullable = true)
|-- Agency: string (nullable = true)
|-- Agency Name: string (nullable = true)
|-- Complaint Type: string (nullable = true)
|-- Descriptor: string (nullable = true)
|-- Location Type: string (nullable = true)
|-- Incident Zip: string (nullable = true)
|-- Incident Address: string (nullable = true)
|-- Street Name: string (nullable = true)
|-- Cross Street 1: string (nullable = true)
|-- Cross Street 2: string (nullable = true)
|-- Intersection Street 1: string (nullable = true)
|-- Intersection Street 2: string (nullable = true)
|-- Address Type: string (nullable = true)
|-- City: string (nullable = true)
|-- Landmark: string (nullable = true)
|-- Facility Type: string (nullable = true)
|-- Status: string (nullable = true)
|-- Due Date: string (nullable = true)
|-- Resolution Action Updated Date: string (nullable = true)
|-- Community Board: string (nullable = true)
|-- Borough: string (nullable = true)
|-- X Coordinate (State Plane): string (nullable = true)
|-- Y Coordinate (State Plane): string (nullable = true)
|-- Park Facility Name: string (nullable = true)
|-- Park Borough: string (nullable = true)
|-- Vehicle Type: string (nullable = true)
|-- Taxi Company Borough: string (nullable = true)
|-- Taxi Pick Up Location: string (nullable = true)
|-- Bridge Highway Name: string (nullable = true)
|-- Bridge Highway Direction: string (nullable = true)
|-- Road Ramp: string (nullable = true)
|-- Bridge Highway Segment: string (nullable = true)
|-- Latitude: string (nullable = true)
|-- Longitude: string (nullable = true)
|-- Location: string (nullable = true)

```

[illegible]

3. (part 2) Display the first 5 records of the DataFrame. Provide a screenshot of the result.

```
scala> d311DF.show(5)
```

[Unique Key]	[Created Date]	[Closed Date]	[Agency]	[Agency Name]	[Complaint Type]	[Descriptor]	[Location Type]	[Incident Zip]
28163399	6/1/2014 0:00	6/1/2014 0:15	DOT	Department of Transportation	Traffic Signal Co...	LED Pedestrian Unit	4 ST E	INTERSECTION
28157590	6/1/2014 0:00	6/6/2014 16:03	DOHMH	Department of Health	Rodent	Mouse Sighting	1-2 Family Dwelling	1
137270-04	ROOSEVELT A...	ROOSEVELT AVENUE	70 STREET	BOE WESTBOUND ENT...	6/6/2014 16:03	02 QUEENS	QUEENS	ADDRESS
1013248	211238	Unspecified	QUEENS	Unspecified	6/9/2014 0:00	07 QUEENS	QUEENS	N
28157974	6/1/2014 0:00	6/10/2014 0:00	HPD	Department of Housing	UNSANITARY CONDITION	GARBAGE/RECYCLING...	RESIDENTIAL BUILDING	1

only showing top 5 rows

4. Use the count action to return the number of items in the DataFrame. Provide a screenshot of the result.

```
scala> d311DF.count
res4: Long = 518909
scala>
```

5. Use a select transformation to return a DataFrame with only the Created Date, Agency, Complaint Type and City. The select transformation should return all columns with an alias instead of the real name. Display the schema of the new DataFrame. Provide a screenshot of the result.

```
scala> val d311Lite=d311DF.select(
  $"Created Date".alias("Date of Creation"),
  $"Agency".alias("Agency Name"),
  $"Complaint Type".alias("Type of Complaint"),
  $"City".alias("City Name")
)
d311Lite: org.apache.spark.sql.DataFrame = [Date of Creation: string, Agency Name: string ... 2 more fields]

scala> d311Lite.show()
[Date of Creation|Agency Name|Type of Complaint|City Name]
[6/1/2014 0:00|DOT|Traffic Signal Co...|NULL]
[6/1/2014 0:00|DOHMH|Rodent|Jackson Heights]
[6/1/2014 0:00|HPD|UNSANITARY CONDITION|NEW YORK]
[6/1/2014 0:00|HPD|UNSANITARY CONDITION|Flushing]
[6/1/2014 0:00|HPD|UNSANITARY CONDITION|BRONX]
[6/1/2014 0:00|HPD|UNSANITARY CONDITION|NEW YORK]
[6/1/2014 0:00|HPD|WATER LEAK|Flushing]
[6/1/2014 0:00|HPD|WATER LEAK|Flushing]
[6/1/2014 0:00|HPD|UNSANITARY CONDITION|BRONX]
[6/1/2014 0:00|HPD|WATER LEAK|Flushing]
[6/1/2014 0:00|DOHMH|Rodent|BROOKLYN]
[6/1/2014 0:00|DOHMH|Rodent|Flushing]
[6/1/2014 0:00|HPD|HEAT/HOT WATER|BROOKLYN]
[6/1/2014 0:00|DOHMH|Rodent|BROOKLYN]
[6/1/2014 0:00|DOHMH|Rodent|BRONX]
[6/1/2014 0:00|DOHMH|Rodent|BRONX]
[6/1/2014 0:00|DOHMH|Rodent|BRONX]
[6/1/2014 0:00|DOHMH|Rodent|BRONX]
[6/1/2014 0:00|DOHMH|Rodent|BROOKLYN]
[6/1/2014 0:00|DOHMH|Rodent|BROOKLYN]

only showing top 20 rows

scala> d311Lite.printSchema()
root
|-- Date of Creation: string (nullable = true)
|-- Agency Name: string (nullable = true)
|-- Type of Complaint: string (nullable = true)
|-- City Name: string (nullable = true)

scala>
```

6. Write a query (a series of one or more transformations followed by an action) that displays the first 20 lines of Agency, City, Complaint Type, where City is not null. Provide a screenshot of the result.

```
scala> val simpleQuery20=d311DF.filter($"City".isNotNull).select($"Agency", $"City", $"Complaint Type")
simpleQuery20: org.apache.spark.sql.DataFrame = [Agency: string, City: string ... 1 more field]

scala> simpleQuery20.show()
+-----+-----+-----+
|Agency|City|Complaint Type|
+-----+-----+-----+
|DOHMH|Jackson Heights|Rodent|
|HPD|NEW YORK|UNSANITARY CONDITION|
|HPD|Flushing|UNSANITARY CONDITION|
|HPD|BRONX|UNSANITARY CONDITION|
|HPD|NEW YORK|UNSANITARY CONDITION|
|HPD|Flushing|WATER LEAK|
|HPD|Flushing|WATER LEAK|
|HPD|BRONX|UNSANITARY CONDITION|
|HPD|Flushing|WATER LEAK|
|DOHMH|BROOKLYN|Rodent|
|DOHMH|Flushing|Rodent|
|HPD|BROOKLYN|HEAT/HOT WATER|
|DOHMH|BROOKLYN|Rodent|
|DOHMH|BRONX|Rodent|
|DOHMH|BRONX|Rodent|
|DOHMH|BRONX|Rodent|
|DOHMH|BRONX|Rodent|
|DOHMH|BROOKLYN|Rodent|
|DOHMH|BROOKLYN|Rodent|
|DOHMH|NEW YORK|Rodent|
+-----+-----+-----+

only showing top 20 rows

scala> █
```