

Step 1: Read a file in spark dataframe

```
Activities Terminal Thu 00:15
stephen@stephen-VirtualBox: ~
File Edit View Search Terminal Help
scala> val zipCodesdf = spark.read.format("csv").option("header",true).load("hdfs://localhost:9000/data-space/zipcodes.csv")
zipCodesdf: org.apache.spark.sql.DataFrame = [RecordNumber: string, Zipcode: string ... 18 more fields]
scala> zipCodesdf.printSchema
root
|-- RecordNumber: string (nullable = true)
|-- Zipcode: string (nullable = true)
|-- ZipcodeType: string (nullable = true)
|-- City: string (nullable = true)
|-- State: string (nullable = true)
|-- LocationType: string (nullable = true)
|-- Lat: string (nullable = true)
|-- Long: string (nullable = true)
|-- Xaxis: string (nullable = true)
|-- Yaxis: string (nullable = true)
|-- Zaxis: string (nullable = true)
|-- WorldRegion: string (nullable = true)
|-- Country: string (nullable = true)
|-- LocationText: string (nullable = true)
|-- Location: string (nullable = true)
|-- Decommissioned: string (nullable = true)
|-- TaxReturnsFiled: string (nullable = true)
|-- EstimatedPopulation: string (nullable = true)
|-- TotalWages: string (nullable = true)
|-- Notes: string (nullable = true)
```

Step 2: Filter it on a column

```
Activities Terminal Thu 00:01
stephen@stephen-VirtualBox: ~
File Edit View Search Terminal Help
scala> val selctColumnndf = zipCodesdf.select("RecordNumber","ZipCode","City","State","Lat","Long")
selctColumnndf: org.apache.spark.sql.DataFrame = [RecordNumber: string, ZipCode: string ... 4 more fields]
scala> selctColumnndf.show()
+-----+-----+-----+-----+-----+
|RecordNumber|ZipCode|City|State|Lat|Long|
+-----+-----+-----+-----+-----+
|1|704|PARC PARQUE|PR|17.96|-66.22|
|2|704|PASEO COSTA DEL SUR|PR|17.96|-66.22|
|10|709|BDA SAN LUIS|PR|18.14|-66.26|
|61391|76166|CINGULAR WIRELESS|TX|32.72|-97.31|
|61392|76177|FORT WORTH|TX|32.75|-97.33|
|61393|76177|FT WORTH|TX|32.75|-97.33|
|4|704|URB EUGENE RICE|PR|17.96|-66.22|
|39827|85209|MESA|AZ|33.37|-111.64|
|39828|85210|MESA|AZ|33.38|-111.84|
|49345|32046|HILLIARD|FL|30.69|-81.92|
|49346|34445|HOLDER|FL|28.96|-82.41|
|49347|32564|HOLT|FL|30.72|-86.67|
|49348|34487|HOMOSASSA|FL|28.78|-82.61|
|10|708|BDA SAN LUIS|PR|18.14|-66.26|
|3|704|SECT LANAUSSSE|PR|17.96|-66.22|
|54354|36275|SPRING GARDEN|AL|33.97|-85.55|
|54355|35146|SPRINGVILLE|AL|33.77|-86.47|
|54356|35585|SPRUCE PINE|AL|34.37|-87.69|
|76511|27007|ASH HILL|NC|36.4|-80.56|
|76512|27203|ASHEBORO|NC|35.71|-79.81|
+-----+-----+-----+-----+-----+
```

Step 3 & 4: create a dataframe using that filter and print out final dataframe

Activities Terminal Wed 23:38

stephen@stephen-VirtualBox: ~/Workspace/Metanauts_BD

stephen@stephen-VirtualBox: ~

File Edit View Search Terminal Help

only showing top 20 rows

```
scala> val filterSelctDF = selctColumnndf.filter(selctColumnndf("Lat") >= 19)
filterSelctDF: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [RecordN
umber: string, ZipCode: string ... 4 more fields]
```

```
scala> filterSelctDF.show()
```

RecordNumber	ZipCode	City	State	Lat	Long
61391	76166	CINGULAR WIRELESS	TX	32.72	-97.31
61392	76177	FORT WORTH	TX	32.75	-97.33
61393	76177	FT WORTH	TX	32.75	-97.33
39827	85209	MESA	AZ	33.37	-111.64
39828	85210	MESA	AZ	33.38	-111.84
49345	32046	HILLIARD	FL	30.69	-81.92
49346	34445	HOLDER	FL	28.96	-82.41
49347	32564	HOLT	FL	30.72	-86.67
49348	34487	HOMOSASSA	FL	28.78	-82.61
54354	36275	SPRING GARDEN	AL	33.97	-85.55
54355	35146	SPRINGVILLE	AL	33.77	-86.47
54356	35585	SPRUCE PINE	AL	34.37	-87.69
76511	27007	ASH HILL	NC	36.4	-80.56