

Spark shell

- `pip install pyspark`

- pyspark

```
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Welcome to

      /_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_\
     /  \  /  \  /  \  /  \  /  \  /  \
    /_/_/  /_/_/  /_/_/  /_/_/  /_/_/
   /_/_/  /_/_/  /_/_/  /_/_/  /_/_/
  /_/_/  /_/_/  /_/_/  /_/_/  /_/_/
 /_/_/  /_/_/  /_/_/  /_/_/  /_/_/
/_/_/  /_/_/  /_/_/  /_/_/  /_/_/

version 3.1.2

Using Python version 3.8.10 (default, Jun  2 2021 10:49:15)
Spark context Web UI available at http://192.168.1.33:4040
Spark context available as 'sc' (master = local[*], app id = local-1633970851451).
SparkSession available as 'spark'.
>>> lines = spark.read.text('/home/hadoop_log.log')
>>> lines.count()
377
>>>
```

Primjer

- Fajl `clients_ids.log` sadrži ID klijenata koji su pokrenuli neku transakciju tokom dana, jedan red u fajlu odnosi se na jedan dan
- Potrebno je naći ID klijenata koji su realizovali bar jednu transakciju tokom posljednje sedmice

```
15, 16, 20, 20  
77, 80, 94  
94, 98, 16, 31  
31, 15, 20
```

Rješenje

```
/clients_ids.log  
>>> clients = spark.read.text('clients_ids.log')  
>>> clients.show()
```

```
+-----+  
|      value|  
+-----+  
|15, 16, 20, 20|  
|   77, 80, 94|  
|94, 98, 16, 31|  
|   31, 15, 20|  
+-----+
```

```
>>> █
```

```
>>> df1 = clients.rdd.flatMap(lambda line: line)  
>>> df1.collect()  
['15, 16, 20, 20', '77, 80, 94', '94, 98, 16, 31', '31, 15, 20']  
>>> df2 = df1.flatMap(lambda item: item.split(", "))  
>>> df2.collect()  
['15', '16', '20', '20', '77', '80', '94', '94', '98', '16', '31', '31', '15', '20']  
>>> unique = df2.distinct()  
>>> unique.count()  
8  
>>> unique.collect()  
['15', '16', '20', '77', '80', '94', '98', '31']  
>>>
```

Kreiranje uzorka

```
[ 0.0]
>>> sample = unique.sample(False, 0.3)
>>> sample.count()
4
>>> sample.collect()
['80', '94', '98', '31']
>>> sample = unique.sample(False, 0.3)
>>> sample.collect()
['20', '94']
>>>
```

```
>>> sample3 = unique.takeSample(False, 5)
>>> sample3
['20', '98', '94', '16', '77']
>>> sample4 = unique.takeSample(True, 5)
>>> sample4
['16', '98', '98', '94', '31']
>>>
```

Osnovne statističke funkcije

```
>>> idsInt.collect()
[15, 16, 20, 77, 80, 94, 98, 31]
>>> intIds = unique.map(lambda el: int(el))
>>> intIds.max()
98
>>> intIds.mean()
53.875
>>> intIds.sum()
431
>>>
```

Histogram

```
>>> intIds.histogram([10, 20, 40, 100])  
([10, 20, 40, 100], [2, 2, 4])  
>>> intIds.histogram(3)  
([15.0, 42.66666666666667, 70.33333333333334, 98], [4, 0, 4])  
>>>
```