```
%spark2.pyspark

rdd1 = sc.parallelize([1,2,3,4])
rdd1.collect()
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=1) FINISHED

```
[1, 2, 3, 4]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 2:46:36 PM.

```
%spark2.pyspark
rdd_double = rdd1.map( lambda v: v*2 )
rdd_double.collect()
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=2) FINISHED

```
[2, 4, 6, 8]
```

Took 1 sec. Last updated by a.jourdan-dsti at March 16 2020, 2:48:24 PM.

FINISHED

ReduceByKey

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:47:27 PM.

```
%spark2.pyspark
rdd_kv = sc.parallelize([("alice",1),("alice",1),("rabbit",1),("alice",2)])

rdd_sum = rdd_kv.reduceByKey( lambda a,b: a+b)
rdd sum.collect()
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=4) FINISHED

```
[('rabbit', 1), ('alice', 4)]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 2:56:32 PM.

FINISHED

Lab: Shop revenues analysis

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:08:23 PM.

```
%spark2.pyspark
rdd = sc.wholeTextFiles('/learning/data/city_revenue')
rdd.take(2)
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=5) FINISHED

```
[(u'hdfs://hdfs-nn-1.au.adaltas.cloud:8020/learning/data/city_revenue/anger.txt', u'JAN 13\r\nFEB 1
2\r\nMAR 14\r\nAPR 15\r\nMAY 12\r\nJUN 15\r\nJUL 19\r\nAUG 15\r\nSEP 13\r\nOCT 8\r\nNOV 14\r\nDEC 1
6'), (u'hdfs://hdfs-nn-1.au.adaltas.cloud:8020/learning/data/city_revenue/lyon.txt', u'JAN 13\r\nFE
B 12\r\nMAR 14\r\nAPR 15\r\nMAY 12\r\nJUN 15\r\nJUL 19\r\nAUG 25\r\nSEP 13\r\nOCT 11\r\nNOV 22\r\nD
EC 22')]
```

Took 1 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:07:12 PM.

dsti-a19/ajourdan/lab1

FINISHED

1) Transform this rdd to get a rdd with format (city, store, month, revenue)

To begin with, we have a key value RDD with (key=HDFS path, value=file content)
-> transform this rdd to get a rdd with format (city,store,month,revenue)
example:
("Paris","Paris_2","JAN",43)
("Paris","Paris_2","FEB",42)
and so on

Useful functions: `map` , `flatMap` , `flatMapValues`

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:35:06 PM.

```
%spark2.pyspark
rdd_split = rdd.map(lambda x: (x[0].split('/')[-1],x[1].split('\r\n')))
rdd_split.take(5)
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=46) FINISHED

```
[(u'anger.txt', [u'JAN 13', u'FEB 12', u'MAR 14', u'APR 15', u'MAY 12', u'JUN 15', u'JUL 19', u'AUG
15', u'SEP 13', u'OCT 8', u'NOV 14', u'DEC 16']), (u'lyon.txt', [u'JAN 13', u'FEB 12', u'MAR 14',
u'APR 15', u'MAY 12', u'JUN 15', u'JUL 19', u'AUG 25', u'SEP 13', u'OCT 11', u'NOV 22', u'DEC 2
2']), (u'marseilles_1.txt', [u'JAN 21', u'FEB 21', u'MAR 21', u'APR 27', u'MAY 25', u'JUN 25', u'JU
L 21', u'AUG 22', u'SEP 23', u'OCT 28', u'NOV 24', u'DEC 26']), (u'nantes.txt', [u'JAN 16', u'FEB 1
5', u'MAR 20', u'APR 12', u'MAY 21', u'JUN 28', u'JUL 19', u'AUG 11', u'SEP 13', u'OCT 14', u'NOV 1
4', u'DEC 24']), (u'nice.txt', [u'JAN 16', u'FEB 15', u'MAR 20', u'APR 9', u'MAY 11', u'JUN 18',
u'JUL 19', u'AUG 11', u'SEP 23', u'OCT 18', u'NOV 14', u'DEC 29'])]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:45:49 PM.

```
%spark2.pyspark
rdd_flat = rdd_split.flatMapValues(lambda x: x)
rdd_flat.take(5)
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=47) FINISHED

```
[(u'anger.txt', u'JAN 13'), (u'anger.txt', u'FEB 12'), (u'anger.txt', u'MAR 14'), (u'anger.txt',
u'APR 15'), (u'anger.txt', u'MAY 12')]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:45:52 PM.

```
%spark2.pyspark
rdd_split2 = rdd_flat.map(lambda x: (x[0].split('.')[0],x[1].split(' ')))
rdd_split2.take(5)
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=48) FINISHED

```
[(u'anger', [u'JAN', u'13']), (u'anger', [u'FEB', u'12']), (u'anger', [u'MAR', u'14']), (u'anger',
[u'APR', u'15']), (u'anger', [u'MAY', u'12'])]
```

Took 1 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:45:55 PM.

```
%spark2.pyspark
rdd_final = rdd_split2.map(lambda x: (x[0],x[1][0],x[1][1]))
rdd_final.take(10)
```

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=49) FINISHED

```
[(u'anger', u'JAN', u'13'), (u'anger', u'FEB', u'12'), (u'anger', u'MAR', u'14'), (u'anger', u'AP
R', u'15'), (u'anger', u'MAY', u'12'), (u'anger', u'JUN', u'15'), (u'anger', u'JUL', u'19'), (u'ang
er', u'AUG', u'15'), (u'anger', u'SEP', u'13'), (u'anger', u'OCT', u'8')]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:45:56 PM.

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=50) FINISHED

```
%spark2.pyspark

def f(x):
    if '_' in x[0]:
        a = x[0].split('_')[0]
        b = x[0]
    else:
        a = x[0]
        b = x[0]
    return (a,b,x[1],x[2])

rdd_final2 = rdd_final.map(f)
rdd_final2.collect()
```

```
u'27'), (u'marseilles', u'marseilles_1', u'MAY', u'25'), (u'marseilles', u'marseilles_1', u'JU
N', u'25'), (u'marseilles', u'marseilles_1', u'JUL', u'21'), (u'marseilles', u'marseilles_1',
u'AUG', u'22'), (u'marseilles', u'marseilles_1', u'SEP', u'23'), (u'marseilles', u'marseilles_
1', u'OCT', u'28'), (u'marseilles', u'marseilles_1', u'NOV', u'24'), (u'marseilles', u'marseille
s_1', u'DEC', u'26'), (u'nantes', u'nantes', u'JAN', u'16'), (u'nantes', u'nantes', u'FEB', u'1
5'), (u'nantes', u'nantes', u'MAR', u'20'), (u'nantes', u'nantes', u'APR', u'12'), (u'nantes',
u'nantes', u'MAY', u'21'), (u'nantes', u'nantes', u'JUN', u'28'), (u'nantes', u'nantes', u'JUL',
u'19'), (u'nantes', u'nantes', u'AUG', u'11'), (u'nantes', u'nantes', u'SEP', u'13'), (u'nante
s', u'nantes', u'OCT', u'14'), (u'nantes', u'nantes', u'NOV', u'14'), (u'nantes', u'nantes', u'D
EC', u'24'), (u'nice', u'nice', u'JAN', u'16'), (u'nice', u'nice', u'FEB', u'15'), (u'nice', u'n
ice', u'MAR', u'20'), (u'nice', u'nice', u'APR', u'9'), (u'nice', u'nice', u'MAY', u'11'), (u'ni
ce', u'nice', u'JUN', u'18'), (u'nice', u'nice', u'JUL', u'19'), (u'nice', u'nice', u'AUG', u'1
1'), (u'nice', u'nice', u'SEP', u'23'), (u'nice', u'nice', u'OCT', u'18'), (u'nice', u'nice',
u'NOV', u'14'), (u'nice', u'nice', u'DEC', u'29'), (u'orlean', u'orlean', u'JAN', u'13'), (u'orl
ean', u'orlean', u'FEB', u'12'), (u'orlean', u'orlean', u'MAR', u'14'), (u'orlean', u'orlean',
u'APR', u'15'), (u'orlean', u'orlean', u'MAY', u'12'), (u'orlean', u'orlean', u'JUN', u'15'),
(u'orlean', u'orlean', u'JUL', u'19'), (u'orlean', u'orlean', u'AUG', u'25'), (u'orlean', u'orle
an', u'SEP', u'13'), (u'orlean', u'orlean', u'OCT', u'8'), (u'orlean', u'orlean', u'NOV', u'2
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:45:57 PM.

FINISHED

2) Average per month of the the shop (all stores combined)

Sum of all the stores revenue divided by 12.

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:34:45 PM.

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=51) FINISHED

```
%spark2.pyspark
from operator import add
red_2 = rdd_final2.map(lambda x: int(x[3])).reduce(add)/12
red_2
```

```
301
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:46:00 PM.

FINISHED

3) Total revenue per city for the year

Sum of all the revenue for each city.

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:35:32 PM.

SPARK JOBS FINISHED

```
%spark2.pyspark
```

```
from operator import add
red_2 = rdd_final2.map(lambda x: (x[0],int(x[3]))).reduceByKey(add)
```

```
[(u'paris', 1568), (u'troyes', 214), (u'lyon', 193), (u'toulouse', 177), (u'anger', 166), (u'orlea
n', 196), (u'rennes', 180), (u'nice', 203), (u'nantes', 207), (u'marseilles', 515)]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:46:14 PM.

FINISHED

4) Average per month per city (on this 1 year data)

Like question 3 but devided by 12.

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:46:15 PM.

SPARK JOBS FINISHED

```
%spark2.pyspark
from operator import add
red_2 = rdd_final2.map(lambda x: (x[0],float(x[3])/12)).reduceByKey(add)
red_2.take(10)
```

```
[(u'paris', 130.66666666666669), (u'troyes', 17.833333333333336), (u'lyon', 16.083333333333336),
(u'toulouse', 14.75), (u'anger', 13.833333333333334), (u'orlean', 16.333333333333336), (u'rennes',
14.999999999999998), (u'nice', 16.916666666666664), (u'nantes', 17.25), (u'marseilles', 42.91666666
666667)]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:46:18 PM.

FINISHED

5) Total revenue per store on the year

Sum of all the revenue per store.

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:36:12 PM.

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=58) FINISHED

```
%spark2.pyspark
from operator import add
red_2 = rdd_final2.map(lambda x: (x[1],int(x[3]))).reduceByKey(add)
red_2.collect()
```

```
[(u'troyes', 214), (u'lyon', 193), (u'toulouse', 177), (u'marseilles_2', 231), (u'anger', 166),
(u'paris_3', 330), (u'paris_1', 596), (u'orlean', 196), (u'marseilles_1', 284), (u'rennes', 180),
(u'nice', 203), (u'paris_2', 642), (u'nantes', 207)]
```

Took 1 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:48:28 PM.

FINISHED

6) For each month, best store (most revenue)

Get the store with highest revenue each month. Hint: this can be done with a reduceByKey.

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 3:36:32 PM.

SPARK JOB (http://wrk-2.au.adaltas.cloud:40135/jobs/job?id=64) FINISHED

```
%spark2.pyspark

def f(x,y):
    print(x[1])
    if x[1] > y[1]:
        best = x
    else:
        best = y
    return best

red_2 = rdd_final2.map(lambda x: (x[2],[x[1],int(x[3])])).reduceByKey(f).map(lambda x: (x[0],x[1
red_2.collect()
```

```
[(u'FEB', u'paris_2'), (u'AUG', u'paris_2'), (u'APR', u'paris_1'), (u'JUN', u'paris_2'), (u'JUL',
u'paris_1'), (u'JAN', u'paris_1'), (u'MAY', u'paris_2'), (u'NOV', u'paris_2'), (u'MAR', u'paris_
2'), (u'DEC', u'paris_1'), (u'OCT', u'paris_1'), (u'SEP', u'paris_2')]
```

Took 0 sec. Last updated by a.jourdan-dsti at March 16 2020, 4:06:00 PM.

```
%spark2.pyspark
```

READY

```
%spark2.pyspark
```

READY