

# BDA - Lab 1

**Student1:** Akshay Gurudath(Aksgu350)

**Student2:** Keshav Padiyar Manuru(Kespa139)

In [ ]:

```
from pyspark import SparkContext
import pyspark.sql.functions as F
from operator import add
import sys

# Set up Spark Context
sc = SparkContext(appName = "BDA Lab1")

# Reading Temperature data
rdd_tempReadings = sc.textFile("file:///home/x_kesma/Lab1/input_data/temperature-readin
gs.csv") \
    .map(lambda line: line.split(";"))

# Reading Precipitation data
rdd_precReadings = sc.textFile("file:///home/x_kesma/Lab1/input_data/precipitation-read
ings.csv") \
    .map(lambda line: line.split(";"))

# Reading Ostergotland Stations data
rdd_OstStations = sc.textFile("file:///home/x_kesma/Lab1/input_data/stations-Ostergotla
nd.csv")\
    .map(lambda line: line.split(";"))\
    .map(lambda line: int(line[0]))
```

In [ ]:

```
print("Executing Q1")
rdd_filtered_1 = rdd_tempReadings.filter(lambda line: (int(line[1][0:4]))>=1950 and int
(line[1][0:4])<=2014) \
    .map(lambda line: (line[1][0:4],(float(line[3]))))

# 1_1
# Maximum Temp for each year
max_Temp = (rdd_filtered_1.reduceByKey(max)\
    .sortBy(keyfunc=lambda k: k[0],ascending = False))

# Save the output
max_Temp.repartition(1).saveAsTextFile('file:///home/x_kesma/Lab1/input_data/results/BD
A_LAB1/Q1_1/')
```

## Result:

```
1 (u'2014', 34.4)
2 (u'2013', 31.6)
3 (u'2012', 31.3)
4 (u'2011', 32.5)
5 (u'2010', 34.4)
6 (u'2009', 31.5)
7 (u'2008', 32.2)
8 (u'2007', 32.2)
9 (u'2006', 32.7)
10 (u'2005', 32.1)
11 (u'2004', 30.2)
12 (u'2003', 32.2)
13 (u'2002', 33.3)
14 (u'2001', 31.9)
15 (u'2000', 33.0)
16 (u'1999', 32.4)
17 (u'1998', 29.2)
18 (u'1997', 31.8)
19 (u'1996', 30.8)
20 (u'1995', 30.8)
```

In [ ]:

```
# 1_2
# Minimum Temp for each year
min_Temp = (rdd_filtered_1.reduceByKey(min)\
             .sortBy(keyfunc=lambda k: k[0],ascending = False))

min_Temp.repartition(1).saveAsTextFile('file:///home/x_kesma/Lab1/input_data/results/BD
A_LAB1/Q1_2/')

```

## Result:

```
1 (u'2014', -42.5)
2 (u'2013', -40.7)
3 (u'2012', -42.7)
4 (u'2011', -42.0)
5 (u'2010', -41.7)
6 (u'2009', -38.5)
7 (u'2008', -39.3)
8 (u'2007', -40.7)
9 (u'2006', -40.6)
10 (u'2005', -39.4)
11 (u'2004', -39.7)
12 (u'2003', -41.5)
13 (u'2002', -42.2)
14 (u'2001', -44.0)
15 (u'2000', -37.6)
16 (u'1999', -49.0)
17 (u'1998', -42.7)
18 (u'1997', -40.2)
19 (u'1996', -41.7)
20 (u'1995', -37.6)
```

In [ ]:

```
# 2_1 Count the number of readings for each month in the period of 1950-2014 which are
higher than 10 degrees
print("Executing Q2_1")
rdd_filtered_2_1 = rdd_tempReadings.filter(lambda line: ((int(line[1][0:4]))>=1950\
                                                         and int(line[1][0:4])<=2014)\
                                                         and float(line[3]) >10 )\
                                         .map(lambda line: ((line[1][0:4], line[1][5:7]),(line[0]
],float(line[3]))))\
                                         .countByKey()

out_2_1 = sc.parallelize(sorted(rdd_filtered_2_1.items(), key = lambda v:v[1], reverse
= True))

out_2_1.repartition(1).saveAsTextFile('file:///home/x_kesma/Lab1/input_data/results/BDA
_LAB1/Q2_1/')

```

## Result:

```
1 ((u'2014', u'07'), 147681)
2 ((u'2011', u'07'), 146656)
3 ((u'2010', u'07'), 143419)
4 ((u'2012', u'07'), 137477)
5 ((u'2013', u'07'), 133657)
6 ((u'2009', u'07'), 133008)
7 ((u'2011', u'08'), 132734)
8 ((u'2009', u'08'), 128349)
9 ((u'2013', u'08'), 128235)
10 ((u'2003', u'07'), 128133)
11 ((u'2002', u'07'), 127956)
12 ((u'2006', u'08'), 127622)
13 ((u'2008', u'07'), 126973)
14 ((u'2002', u'08'), 126073)
15 ((u'2005', u'07'), 125294)
16 ((u'2011', u'06'), 125193)
17 ((u'2012', u'08'), 125037)
18 ((u'2006', u'07'), 124794)
19 ((u'2010', u'08'), 124417)
20 ((u'2014', u'08'), 124045)

```

In [ ]:

```
# 2_2 Repeat the exercise, this time taking only distinct readings from each station.
# That is, if a station reported a reading above 10 degrees in some month, then itappea
rs only
# once in the count for that month

print("Executing Q2_2")
rdd_filtered_2_2 = rdd_tempReadings.filter(lambda line: ((int(line[1][0:4]))>=1950\
                                                         and int(line[1][0:4])<=2014)\
                                                         and float(line[3]) >10 )\
                                         .map(lambda line: (line[1][0:4], line[1][5:7],line[0]))
\
                                         .distinct()\
                                         .map(lambda line: ((line[0],line[1]),(line[2])))\
                                         .countByKey()

out_2_2 = sc.parallelize(sorted(rdd_filtered_2_2.items(), key = lambda v:v[1], reverse
= True))

out_2_2.repartition(1).saveAsTextFile('file:///home/x_kesma/Lab1/input_data/results/BDA
_LAB1/Q2_2/')

```

## Result:

```
1 ((u'1972', u'10'), 378)
2 ((u'1973', u'05'), 377)
3 ((u'1973', u'06'), 377)
4 ((u'1973', u'09'), 376)
5 ((u'1972', u'08'), 376)
6 ((u'1972', u'09'), 375)
7 ((u'1972', u'06'), 375)
8 ((u'1971', u'08'), 375)
9 ((u'1972', u'05'), 375)
10 ((u'1971', u'06'), 374)
11 ((u'1972', u'07'), 374)
12 ((u'1971', u'09'), 374)
13 ((u'1973', u'08'), 373)
14 ((u'1971', u'05'), 373)
15 ((u'1974', u'06'), 372)
16 ((u'1974', u'08'), 372)
17 ((u'1974', u'05'), 370)
18 ((u'1974', u'09'), 370)
19 ((u'1973', u'07'), 370)
20 ((u'1971', u'07'), 370)
```

In [ ]:

```
# 3 Find the average monthly temperature for each available station in Sweden. Your result
#should include average temperature for each station for each month in the period of 19
60-
#2014. Bear in mind that not every station has the readings for each month in this time
frame.
```

```
print("Executing Q3")
rdd_filtered_3 = rdd_tempReadings.filter(lambda line: (int(line[1][0:4]))>=1950 and int
(line[1][0:4])<=2014) \
    .map(lambda line: ((line[1][0:4], line[1][5:7], line[0]),(float(line[3]))))\
    .groupByKey()\
    .mapValues(lambda val: sum(val)/len(val))
out_3 = (rdd_filtered_3\
    .sortBy(keyfunc=lambda k: (k[0][2],k[0][0],k[0][1]),ascending = False))
out_3.saveAsTextFile('file:///home/x_kesma/Lab1/input_data/results/BDA_LAB1/Q3/')
```

## Result:

```
1 ((u'2014', u'12', u'99450'), 1.989784946236562)
2 ((u'2014', u'11', u'99450'), 5.9738888888888884)
3 ((u'2014', u'10', u'99450'), 9.300811907983755)
4 ((u'2014', u'09', u'99450'), 13.712222222222223)
5 ((u'2014', u'08', u'99450'), 16.915053763440866)
6 ((u'2014', u'07', u'99450'), 18.455510752688177)
7 ((u'2014', u'06', u'99450'), 11.006944444444443)
8 ((u'2014', u'05', u'99450'), 7.565456989247306)
9 ((u'2014', u'04', u'99450'), 4.4734722222222218)
10 ((u'2014', u'03', u'99450'), 2.7974462365591384)
11 ((u'2014', u'02', u'99450'), 1.8333333333333337)
12 ((u'2014', u'01', u'99450'), -0.9764784946236575)
13 ((u'2013', u'12', u'99450'), 3.6639077340569917)
14 ((u'2013', u'11', u'99450'), 5.5281944444444446)
15 ((u'2013', u'10', u'99450'), 9.186290322580643)
16 ((u'2013', u'09', u'99450'), 13.620972222222242)
17 ((u'2013', u'08', u'99450'), 17.183333333333332)
18 ((u'2013', u'07', u'99450'), 15.397849462365592)
19 ((u'2013', u'06', u'99450'), 13.851805555555556)
20 ((u'2013', u'05', u'99450'), 8.732795698924724)
```

In [ ]:

```
# 4 Provide a List of stations with their associated maximum measured temperatures and  
# maximum measured daily precipitation. Show only those stations where the maximum  
# temperature is between 25 and 30 degrees and maximum daily precipitation is between 1  
00mm and 200mm.  
  
print("Executing Q4")  
rdd_filter_4 = rdd_precReadings\  
                .filter(lambda line: float(line[3])>=100 and float(line[3])  
                <=200)\  
                .map(lambda line: (line[0],line[3]))\  
                .reduceByKey(max)  
  
rdd_tempReadings_4 = rdd_tempReadings.filter(lambda line: float(line[3])>=25 and float(  
line[3]<=30) )\  
                .map(lambda line: (line[0],line[3]))\  
                .reduceByKey(max)  
  
rdd_result = rdd_tempReadings_4.join(rdd_filter_4)  
  
rdd_result.repartition(1).saveAsTextFile('file:///home/x_kesma/Lab1/input_data/results/  
BDA_LAB1/Q4/')
```

## Result:

No Resultset obtained

In [ ]:

```
### 5 Calculate the average monthly precipitation for the Ostergotland region (list of
stations is provided in the separate file)
### for the period 1993-2016. In order to do this, you will first need to calculate the
totalmonthly precipitation for each
### station before calculating the monthly average (by averaging over stations).

print("Executing Q5")
list_OstStations = rdd_OstStations.collect()

broadcastVar = sc.broadcast(list_OstStations)

rdd_filter_5 = rdd_precReadings.filter(lambda line: (int(line[0]) in broadcastVar.value
) and\
                                         (int(line[1][0:4])>=1993 and int(li
ne[1][0:4])<=2016))\
                                         .map(lambda line: ((line[1][0:4], line[1][5:7], line[0]),(f
loat(line[3]))))\
                                         .reduceByKey(add)\
                                         .map(lambda line:((line[0][0], line[0][1]),(line[1])))\
                                         .groupByKey()\
                                         .mapValues(lambda val:sum(val)/len(val))

rdd_filter_5 = (rdd_filter_5\
               .sortBy(keyfunc=lambda k: (k[0][0],k[0][1]),ascending = False))

rdd_filter_5.repartition(1).saveAsTextFile('file:///home/x_kesma/Lab1/input_data/result
s/BDA_LAB1/Q5/')

sys.exit(0)
```

## Result:

```
1 ((u'2016', u'07'), 0.0)
2 ((u'2016', u'06'), 47.6625)
3 ((u'2016', u'05'), 29.250000000000007)
4 ((u'2016', u'04'), 26.900000000000001)
5 ((u'2016', u'03'), 19.962500000000002)
6 ((u'2016', u'02'), 21.5625)
7 ((u'2016', u'01'), 22.325000000000003)
8 ((u'2015', u'12'), 28.925000000000004)
9 ((u'2015', u'11'), 63.887500000000002)
10 ((u'2015', u'10'), 2.2625)
11 ((u'2015', u'09'), 101.3)
12 ((u'2015', u'08'), 26.987499999999997)
13 ((u'2015', u'07'), 119.09999999999997)
14 ((u'2015', u'06'), 78.662500000000001)
15 ((u'2015', u'05'), 93.225)
16 ((u'2015', u'04'), 15.337499999999999)
17 ((u'2015', u'03'), 42.612500000000004)
18 ((u'2015', u'02'), 24.825)
19 ((u'2015', u'01'), 59.112500000000003)
20 ((u'2014', u'12'), 35.462500000000001)
```