Maitrey Prajapati - Programming Assignment 3

Input file : I changed the format of xlsx file to csv because pyspark doesn’t have a function to load excel file. And after loading the file in pandas and then creating spark dataframe gave error that there were multiple data types in a single column so it can’t be loaded into rdd.

Output :

```
[('Average Temperature (F)', 37.946348928727595)]

[('Average High Temperature (F)', 47.190852575488464)]

[('Average Low Temperature (F)', 28.799822301199466)]

[('Average Precipitation (in)', 34.4720404040404)]
```

Program :

```
from pyspark import SparkContext
from pyspark.sql import SQLContext
from pyspark.sql.types import IntegerType,FloatType

def calculate_total(row_dict):
   tmp_val = float(row_dict['ANNUAL\xa0'])
   cities = int(row_dict['# CITIES\xa0'])
   return (row_dict['Alberta'],(tmp_val*cities,cities))

def calculate_total_temp_cities(row1,row2):
   return(row1[0]+row2[0],row1[1]+row2[1] )

def temp_calc(row):
   return(row[0],row[1][0]/row[1][1])

sc = SparkContext.getOrCreate()
sqlContext = SQLContext(sc)
df = sqlContext.read.format('csv').options(header=True).load('data.csv')
data_df = df.dropna()

avg_tmp = data_df.filter(data_df['Alberta'] == 'Average Temperature (F)').rdd.map(lambda row:
row.asDict())
avg_high = data_df.filter(data_df['Alberta'] == 'Average High Temperature (F)').rdd.map(lambda
row: row.asDict())
avg_low = data_df.filter(data_df['Alberta'] == 'Average Low Temperature (F)').rdd.map(lambda
row: row.asDict())
avg_pcp = data_df.filter(data_df['Alberta'] == 'Average Precipitation (in)').rdd.map(lambda row:
row.asDict())

avg_tmp = avg_tmp.map(calculate_total)
avg_tmp =avg_tmp.reduceByKey(calculate_total_temp_cities)
avg_tmp = avg_tmp.map(temp_calc)
avg_tmp.collect()

avg_high = avg_high.map(calculate_total)
avg_high =avg_high.reduceByKey(calculate_total_temp_cities)
avg_high = avg_high.map(temp_calc)
avg_high.collect()
```

```
avg_low = avg_low.map(calculate_total)
avg_low =avg_low.reduceByKey(calculate_total_temp_cities)
avg_low = avg_low.map(temp_calc)
avg_low.collect()

avg_pcp = avg_pcp.map(calculate_total)
avg_pcp =avg_pcp.reduceByKey(calculate_total_temp_cities)
avg_pcp = avg_pcp.map(temp_calc)
avg_pcp.collect()
```

Answers:

```
'Average Temperature (F)' : 37.946348928727595
'Average High Temperature (F)' :  47.190852575488464
'Average Low Temperature (F)' : 28.799822301199466
'Average Precipitation (in)' :  34.4720404040404
```