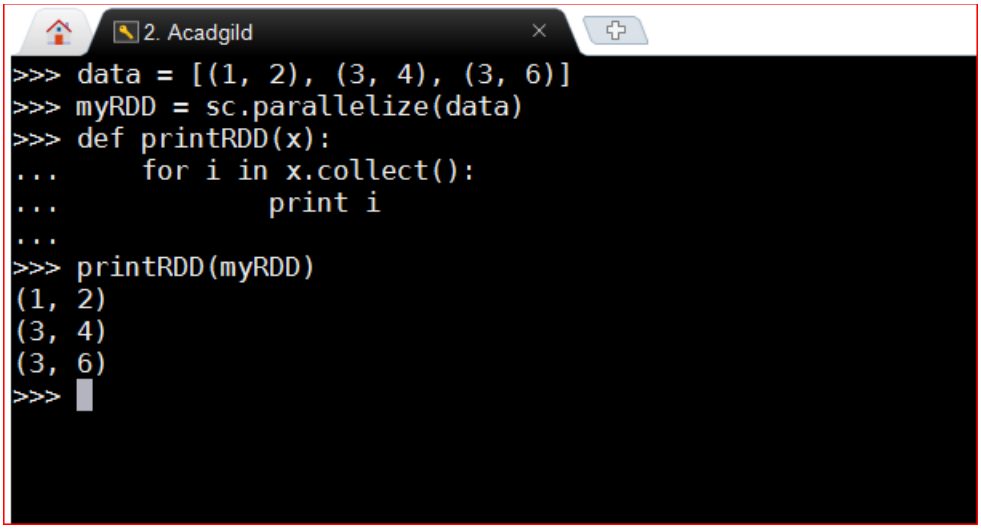
Assignment 13.2

1. Create a simple pairRDD of (1, 2), (3, 4), (3, 6).

```
data = [(1, 2), (3, 4), (3, 6)]          # create list of pairs (tuples)
myRDD = sc.parallelize(data)             # create RDD
def printRDD(x):                         # function to print RDD
        for i in x.collect():
                print i

printRDD(myRDD)                          # RDD printed
```



2. Transform an RDD of ("a","b","c","d","e") to PairRDD (a,0), (b,1), (c,2), (d,3), (e,4)

```
data = ["a","b","c","d","e"]             # create a list of elements as mentioned
myRDD = sc.parallelize(data)             #creating RDD with list of values
def printRDD(x):                         # function to print RDD
        for i in x.collect():
                print i

count = -1                               # counter set to -1
```

```
def evaluate(x):                          # evaluate function to prepare paired element or value for the keys
    global count                          # global variable declared for the computation on the RDD
    count = count+1
    return (x,count)                      # returns the ('a',0) for first element, subsequently ('b',1) .....


printRDD(myRDD)                           # print myRDD
mappedRDD = myRDD.map(evaluate)           # evaluating pair of the keys
printRDD(mappedRDD)                       # printing final RDD.
```