7) Submit a copy of this modified program and a screen shot of the results of the program's execution as the output of your assignment.

Answer:

The modified code is as follows:

```
from mrjob.job import MRJob
import re
class MRWordCount (MRJob):
    def mapper(self, _, line):
        words = re.findall(r'\b[aA-zZ]\w*\b', line)  # Match words starting with a-n or A-N
        for word in words:
            if word[0].lower() <= 'n':
                yield "a-n", 1
            else:
                yield "other", 1
    def combiner(self, category, counts):
        yield category, sum(counts)
    def reducer(self, category, counts):
        yield category, sum(counts)
if __name__ == '__main__':
    MRWordCount.run()
```

```
job output is in hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20230929.213422.822223/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20230929.213422.822223/output...
"a-n"   49
"other" 46
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20230929.213422.822223...
Removing temp directory /tmp/WordCount2.hadoop.20230929.213422.822223...
[hadoop@ip-172-31-2-100 ~]$
```

11) Submit a copy of this modified program and a screen shot of the results of the program's execution as the output of your assignment.

Answer:
The modified code is as follows:

```
from mrjob.job import MRJob

class MRSalaries(MRJob):

    def mapper(self, _, line):

        (_, _, _, _, _, annual_salary, _) = line.split('\t')

        annual_salary = float(annual_salary)

        if annual_salary >= 100000.00:

            salary_category = "High"

        elif 50000.00 <= annual_salary <= 99999.99:

            salary_category = "Medium"

        else:
            salary_category = "Low"

        yield salary_category, 1

    def combiner(self, salary_category, counts):

        yield salary_category, sum(counts)


    def reducer(self, salary_category, counts):

        yield salary_category, sum(counts)


if __name__ == '__main__':

    MRSalaries.run()
```

```
job output is in hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20230929.212056.540124/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20230929.212056.540124/output...
"High"  442
"Low"   7064
"Medium"        6312
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20230929.212056.540124...
Removing temp directory /tmp/Salaries2.hadoop.20230929.212056.540124...
[hadoop@ip-172-31-2-100 ~]$
```

13) Write a program to perform the task of outputting a count of the number of movies each user (identified via their user id) reviewed.

Answer:
The modified code is as follows:

```
from mrjob.job import MRJob

from mrjob.step import MRStep

class MovieReviewCount(MRJob):

    def configure_args(self):

        super(MovieReviewCount, self).configure_args()

    def mapper(self, _, line):

        user_id, _, _, _ = line.split(',')  # Assuming the user_id is in the first column

        yield user_id, 1

    def reducer(self, user_id, counts):

        total_reviews = sum(counts)

        # Modify the output format to include a colon (":") after the user ID

        yield f"{user_id}:", total_reviews

    def steps(self):

        return [

            MRStep(mapper=self.mapper, reducer=self.reducer)

        ]

if __name__ == '__main__':

    MovieReviewCount.run()
```

```
job output is in hdfs:///user/hadoop/tmp/mrjob/user.hadoop.20230929.213718.048312/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/user.hadoop.20230929.213718.048312/output...
"102"   678
"105"   525
"108"   31
"111"   341
"114"   25
"117"   55
"12"    61
"120"   138
"123"   33
"126"   64
"129"   26
"132"   94
"135"   22
```