a. What you learned in the ICP

One of the main things I learned from this ICP was setting up the Apache Spark environment in Google Co-lab. Reading the text file and analyzing it based on the data provided. Learned how to transform a data and why it is important do so when a large dataset is given. For this assignment, the file was not too large, and okay to work with. Which allowed me to check for mistakes to make sure the outputs were being performed correctly. Also getting familiar with the most important concept, which is RDD. RDD stand for Resilient Distributed Dataset and is the main component of Apache Spark. It is an immutable collection of objects that can be computed on different nodes of cluster. All the dataset in Spark RDD is logically partition on various servers that allows computation to happen on different nodes of the cluster.

b. ICP description what was the task you were performing

The goal of ICP was to read a text file and output words that began with the same letter. The following are a brief description of what I was trying to do.

- First the file was read using the following code. (sc.textFile("`/content/icp2.txt`"))
- After reading the file I wanted to know my many lines or elements were in the text file.
- When the output of 7 was displayed typed a simple code to display the content of the file (rdd.take(7)). This would display all the 7 lines of data in array format.
- splitting the data. This was written in a function named (split_fun).
- The data was then mapped, and the function was passed into it. (map(split_fun)
- After the data was mapped it was flattened to make it look cleaner
- Finally, the words were grouped based on the letter they began with.

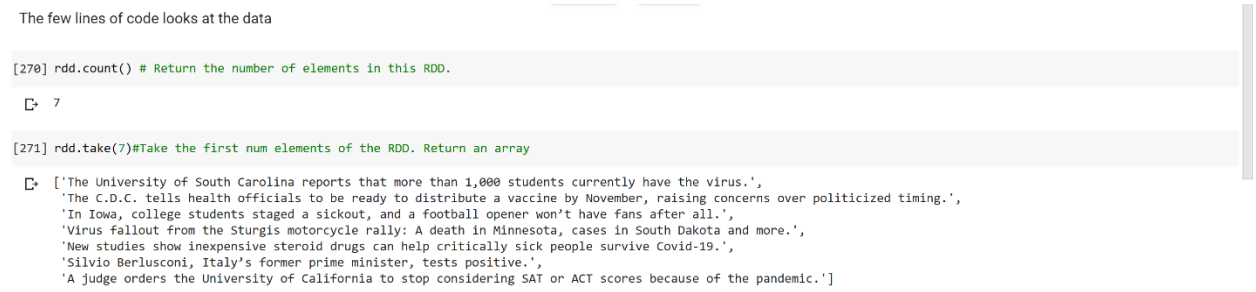
c. Challenges that you faced

The ICP was bit hard because I have never worked with Spark. It was difficult trying to figure out the syntax and how to approach the problem. I first ended up writing the code in simple python code to see what I would need to do. Then I tried to apply that to spark, but later realized that was unnecessary, because Spark implication would be different. One thing I was stuck on was how to split the data after it is read and how to extract it based on the first character then group it. It took lot of trial and error but slowly figured it out.

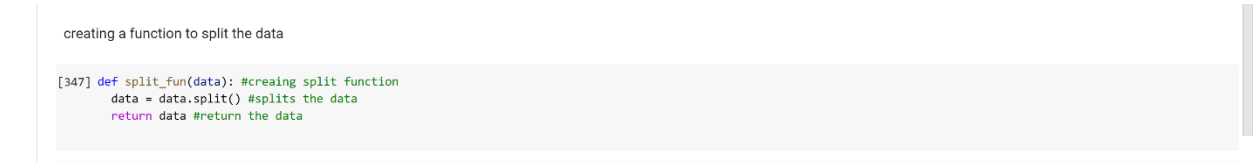
d. Screen shots that shows the successful execution of each required step of your code

The first image is just general import of important libraries to get the program up and running. Also showing the code the file is read and uploaded.

The second image just shows simple data analysis what the data looks like.



Creating a simple function to split the data so its easy to work with.



Passing the split_fun to map and looking at the data.

```
map_rdd = rdd.map(split_fun) #The rdd is mapped by passing in the function
```

+ Code + Text

```
[350] map_rdd.take(2) #takes the first five. Return an array
```

```
[['The',
  'University',
  'of',
  'South',
  'Carolina',
  'reports',
  'that',
  'more',
  'than',
  '1,000',
  'students',
  'currently',
  'have',
  'the',
  'virus.'],
 ['The',
  'C.D.C.',
  'tells',
  'health',
  'officials',
  'to',
  'be',
  'ready',
  'to',
  'distribute',
  'a',
  'vaccine',
  'by',
  'November,',
  'raising',
```

Flatting the maped data so it is easier to look at.

Return a new RDD by first applying a function to all elements of this RDD, and then flattening the results.

```
[351] flatMap_rdd = rdd.flatMap(lower_split_fun) #passing in the function
```

```
[352] flatMap_rdd.take(4)
```

```
['The', 'University', 'of', 'South']
```

Finally grouping the data and printing out the results.

The code below will out put the data based on the first letter of the word. It is grouped.

```
group_rdd = flatMap_rdd.groupBy(lambda w: w[0]) #taking the first letter
y = group_rdd.collect() #Return a list that contains all of the elements in this RDD.
print([(k, list(v)) for (k, v) in y])
print()
```

```
[('S', ['South', 'Sturgis', 'South', 'Silvio', 'SAT']), ('C', ['Carolina', 'C.D.C.', 'Covid-19.', 'California']), ('r', ['reports', 'ready', 'raising', 'rally:']), ('1
```

```
[330]
```

e. Output file link if applicable

No output link

f. Video link (YouTube or any other publicly available video platform)

Should not need access. If you do please let me know. I changed my settings to where it is accessible once the link is clicked.

https://umkc.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=eca38823-2f94-4d32-915e-ac2d0012f24f

g. Any inside about the data or the ICP in general

I think it would be helpful to have resources where I can go learn the topic that is going be tested on in the ICP. Majority of my time was spent researching the basics and how to implement them. But overall challenging and fun long as I figure it out. (lol).