Batch Name: Infosys FP5.0 Summer 2018

Enrollment No: R171217044

SAPID: 500060722

Name: PRAJJAWAL BANATI

Sem: SEM-II

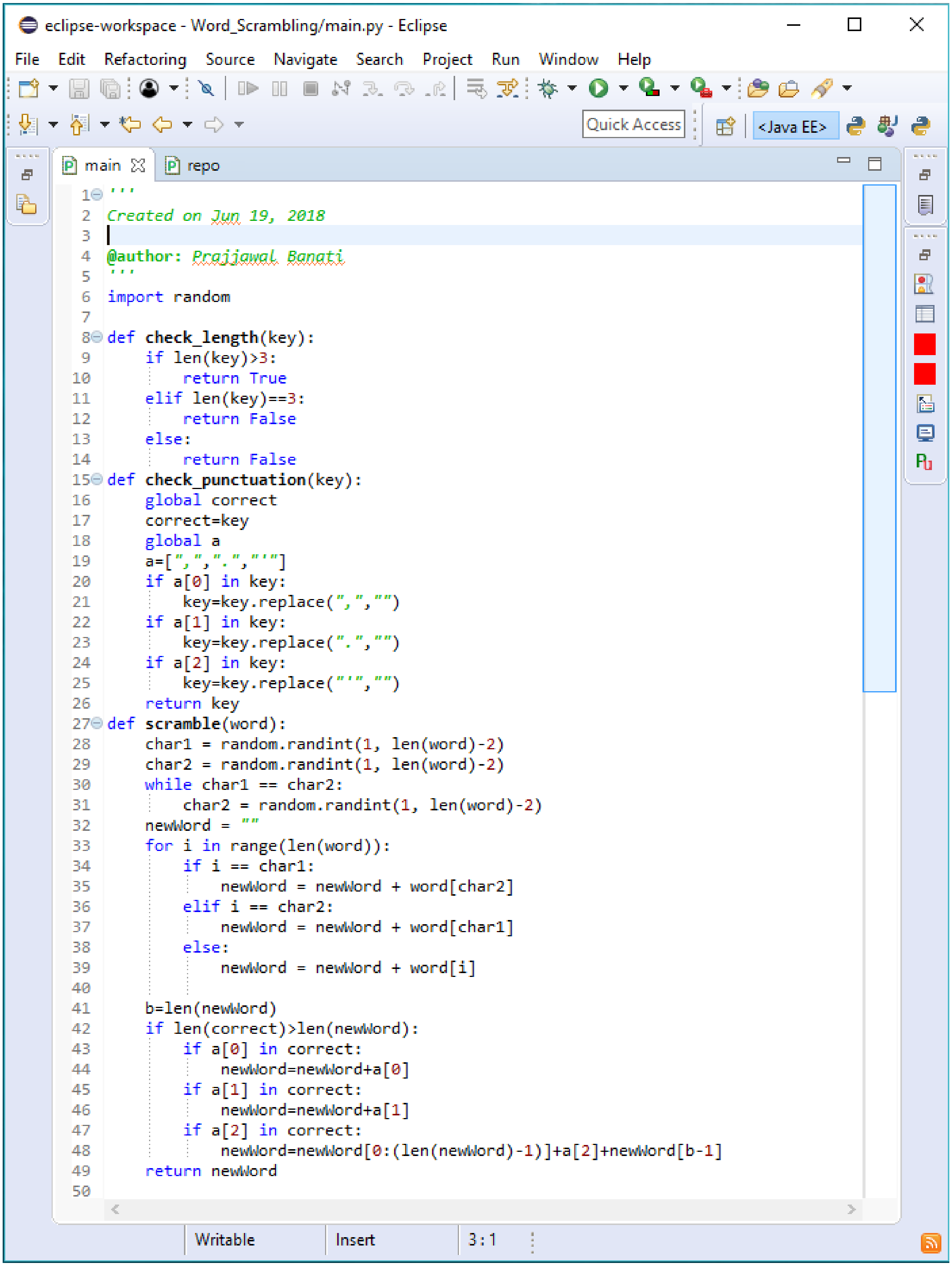
Branch: CSE-DEVOPS

MODULE 1-PROJECT

WORD SCRAMBLING

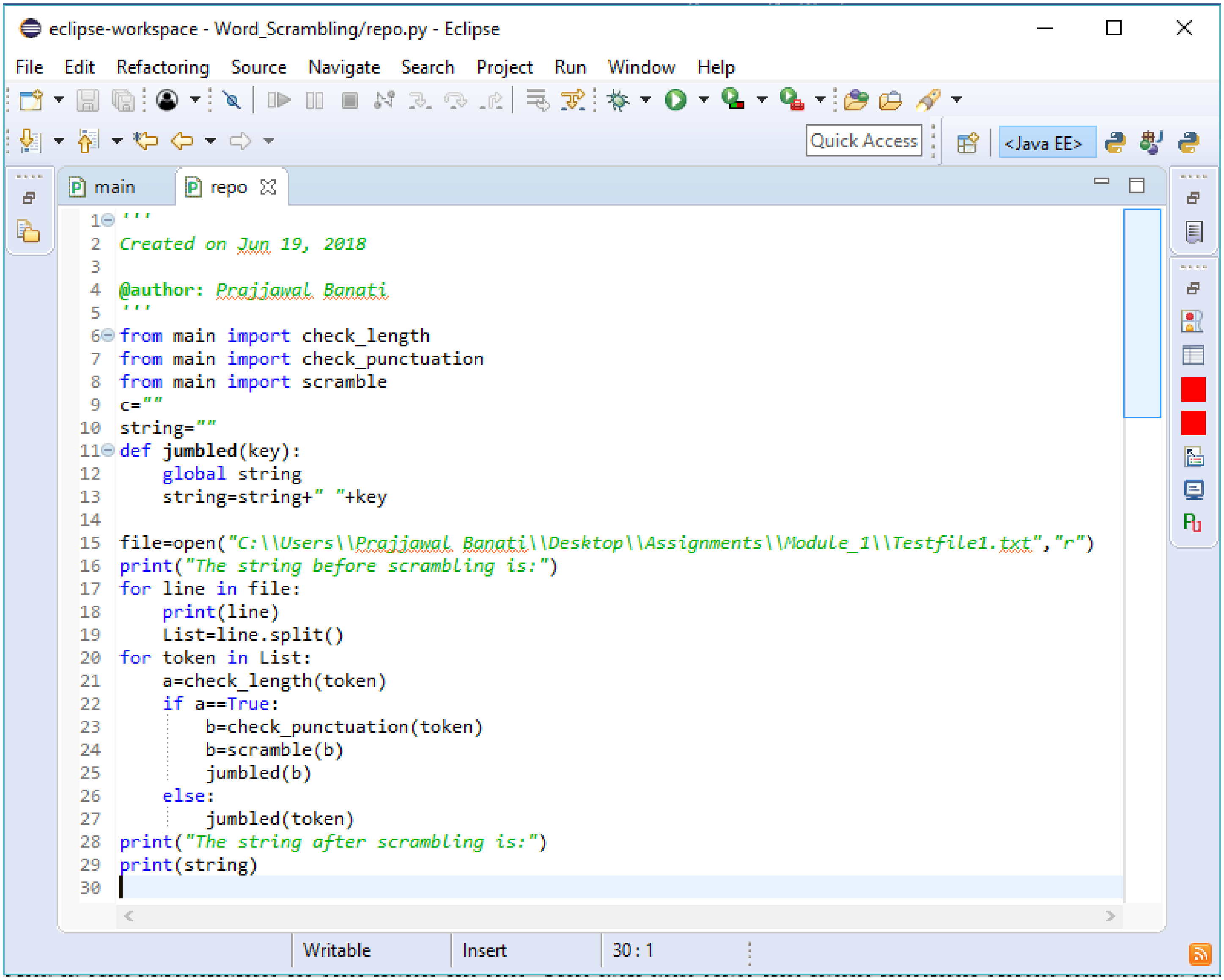
Algorithim:

1. Firstly open the file which you want to scramble in read mode.
2. Then define functions of checking the length, punctuation marks and scramble in a main.py file.
3. Import all those functions into your code for simplicity and to make the code short.
4. Then display the scrambled string.


eclipse-workspace - Word_Scrambling/main.py - Eclipse
File Edit Refactoring Source Navigate Search Project Run Window Help
Quick Access
<Java EE>
main
repo
Writable
Insert
3 : 1

```python
'''
Created on Jun 19, 2018

@author: Prajjawal Banati
'''
import random

def check_length(key):
    if len(key)>3:
        return True
    elif len(key)==3:
        return False
    else:
        return False
def check_punctuation(key):
    global correct
    correct=key
    global a
    a=[",",".","'"]
    if a[0] in key:
        key=key.replace(",","")
    if a[1] in key:
        key=key.replace(".","")
    if a[2] in key:
        key=key.replace("'","")
    return key
def scramble(word):
    char1 = random.randint(1, len(word)-2)
    char2 = random.randint(1, len(word)-2)
    while char1 == char2:
        char2 = random.randint(1, len(word)-2)
    newWord = ""
    for i in range(len(word)):
        if i == char1:
            newWord = newWord + word[char2]
        elif i == char2:
            newWord = newWord + word[char1]
        else:
            newWord = newWord + word[i]

    b=len(newWord)
    if len(correct)>len(newWord):
        if a[0] in correct:
            newWord=newWord+a[0]
        if a[1] in correct:
            newWord=newWord+a[1]
        if a[2] in correct:
            newWord=newWord[0:(len(newWord)-1)]+a[2]+newWord[b-1]
    return newWord
```

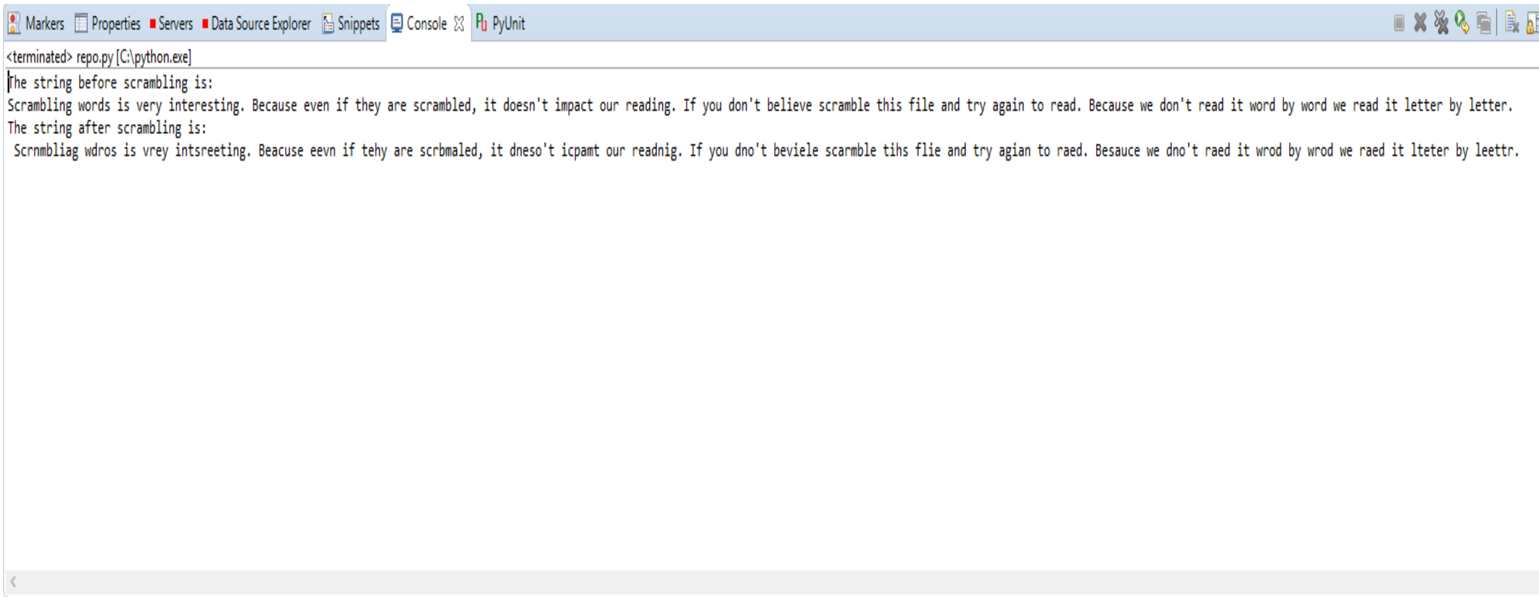
This is the screenshot of the main.py file. You can see that we have defined three functions here which will do there respective work mentioned above in the algorithim


eclipse-workspace - Word_Scrambling/repo.py - Eclipse
File Edit Refactoring Source Navigate Search Project Run Window Help
Quick Access
<Java EE>
main
repo
Writable
Insert
30 : 1

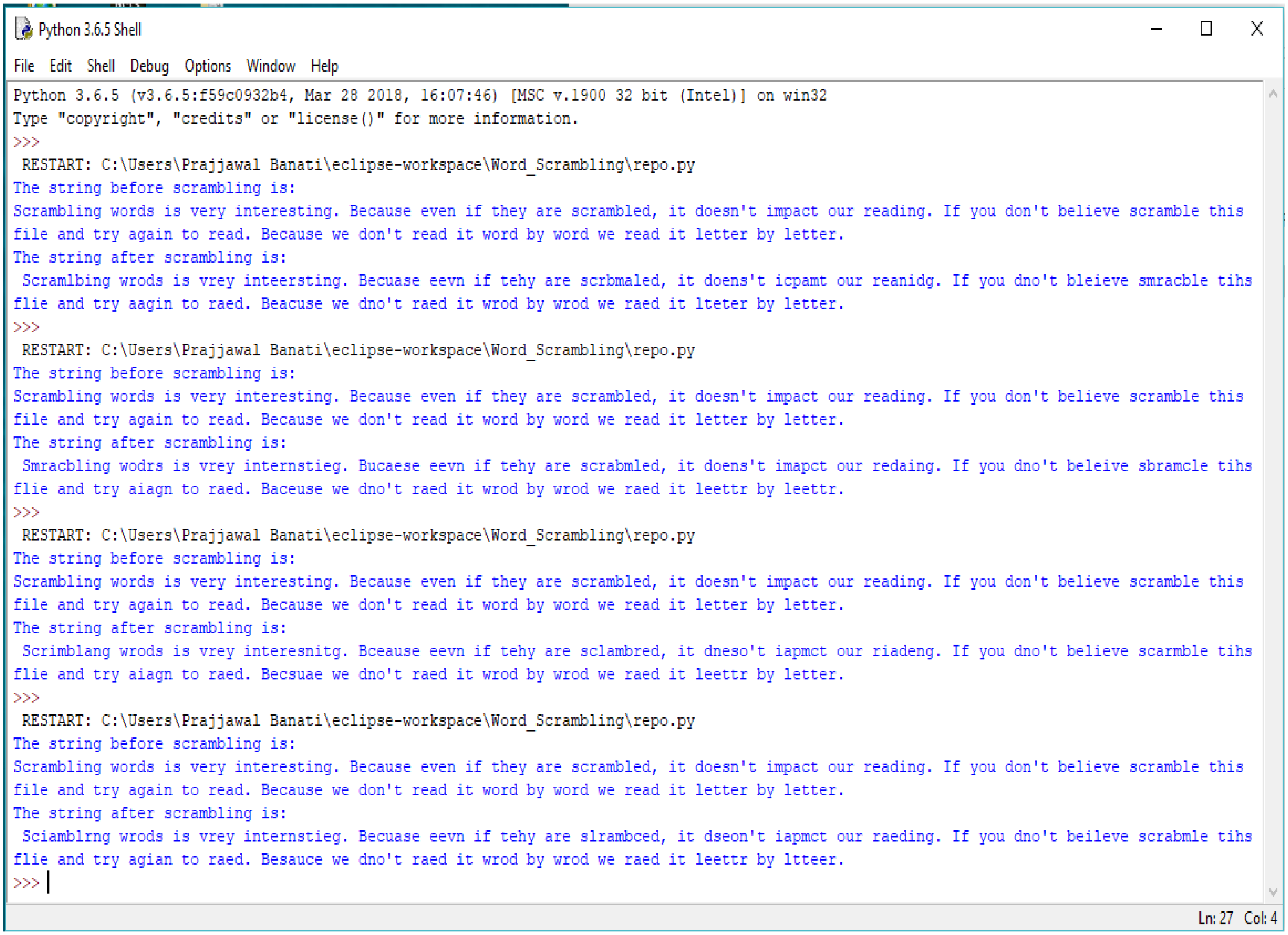
```
'''
Created on Jun 19, 2018

@author: Prajjawal Banati
'''
from main import check_length
from main import check_punctuation
from main import scramble
c=""
string=""
def jumbled(key):
    global string
    string=string+" "+key

file=open("C:\\Users\\Prajjawal Banati\\Desktop\\Assignments\\Module_1\\Testfile1.txt","r")
print("The string before scrambling is:")
for line in file:
    print(line)
    List=line.split()
for token in List:
    a=check_length(token)
    if a==True:
        b=check_punctuation(token)
        b=scramble(b)
        jumbled(b)
    else:
        jumbled(token)
print("The string after scrambling is:")
print(string)

```

This is the code which will read the file and will scramble the words and will give us the output

NOW LET'S SEE THE OUTPUT RESULTS



```
<terminated> repo.py [C:\python.exe]
The string before scrambling is:
Scrambling words is very interesting. Because even if they are scrambled, it doesn't impact our reading. If you don't believe scramble this file and try again to read. Because we don't read it word by word we read it letter by letter.
The string after scrambling is:
 Scrnmbliag wdros is vrey intsreeting. Beacuse eevn if tehy are scrbmaled, it dneso't icpamt our readnig. If you dno't beviele scarmble tihs flie and try agian to raed. Besauce we dno't raed it wrod by wrod we raed it lteter by leettr.
```



```
Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\Prajjawal Banati\eclipse-workspace\Word_Scrambling\repo.py
The string before scrambling is:
Scrambling words is very interesting. Because even if they are scrambled, it doesn't impact our reading. If you don't believe scramble this
file and try again to read. Because we don't read it word by word we read it letter by letter.
The string after scrambling is:
 Scramlbing wrods is vrey inteersting. Becuase eevn if tehy are scrbmaled, it doens't icpamt our reanidg. If you dno't bleieve smracble tihs
flie and try aagin to raed. Beacuse we dno't raed it wrod by wrod we raed it lteter by letter.
>>>
 RESTART: C:\Users\Prajjawal Banati\eclipse-workspace\Word_Scrambling\repo.py
The string before scrambling is:
Scrambling words is very interesting. Because even if they are scrambled, it doesn't impact our reading. If you don't believe scramble this
file and try again to read. Because we don't read it word by word we read it letter by letter.
The string after scrambling is:
 Smracbling wodrs is vrey internstieg. Bucaese eevn if tehy are scrabmled, it doens't imapct our redaing. If you dno't beleive sbramcle tihs
flie and try aiagn to raed. Baceuse we dno't raed it wrod by wrod we raed it leettr by leettr.
>>>
 RESTART: C:\Users\Prajjawal Banati\eclipse-workspace\Word_Scrambling\repo.py
The string before scrambling is:
Scrambling words is very interesting. Because even if they are scrambled, it doesn't impact our reading. If you don't believe scramble this
file and try again to read. Because we don't read it word by word we read it letter by letter.
The string after scrambling is:
 Scrimblang wrods is vrey interesnitg. Bceause eevn if tehy are sclambred, it dneso't iapmct our riadeng. If you dno't believe scarmble tihs
flie and try aiagn to raed. Becsuae we dno't raed it wrod by wrod we raed it leettr by letter.
>>>
 RESTART: C:\Users\Prajjawal Banati\eclipse-workspace\Word_Scrambling\repo.py
The string before scrambling is:
Scrambling words is very interesting. Because even if they are scrambled, it doesn't impact our reading. If you don't believe scramble this
file and try again to read. Because we don't read it word by word we read it letter by letter.
The string after scrambling is:
 Sciamblrng wrods is vrey internstieg. Becuase eevn if tehy are slrambced, it dseon't iapmct our raeding. If you dno't beileve scrabmle tihs
flie and try agian to raed. Besauce we dno't raed it wrod by wrod we raed it leettr by ltteer.
>>>
```

So that’s all about my project. I hope you liked it also I am uploading this project on git(My username on git is 500060722) so that the version control sense any type of changes in the file

Thank you