REPORT

Adv OOP Lab

001910501005

Atanu Ghosh

JU BCSE UG-II Even Sem

Python Assignment - 2

Q1: Write a single python program to do the following operations on a text file by writing different user defined functions.

a. Remove all the special characters.
b. Remove all single characters.
c. Substitute multiple spaces with single space.
d. Convert all the words into Lowercase.
e. Convert the words into literal form from their contracted form (e.g., Couldn’t => Could not)

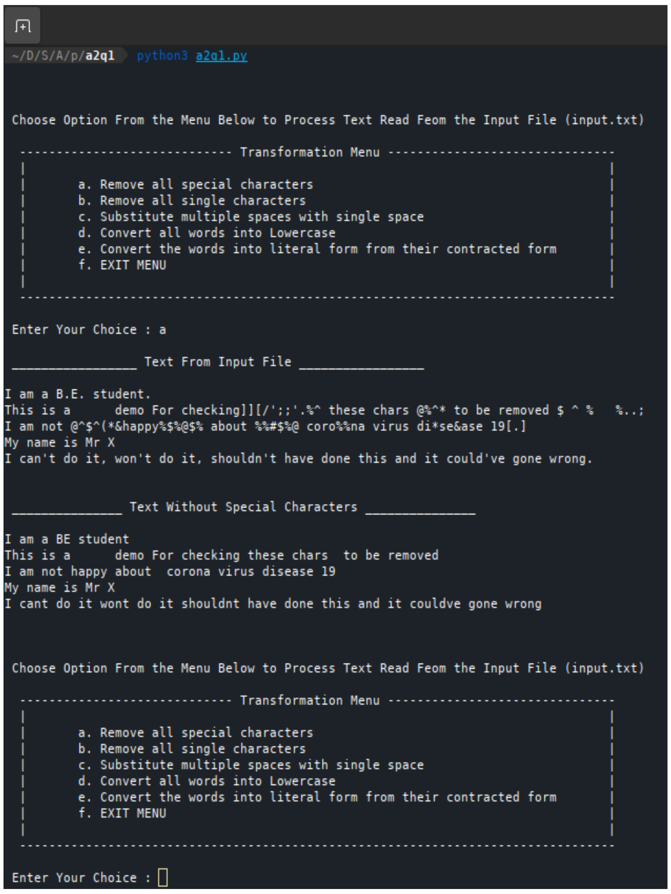
Sol :

Program :

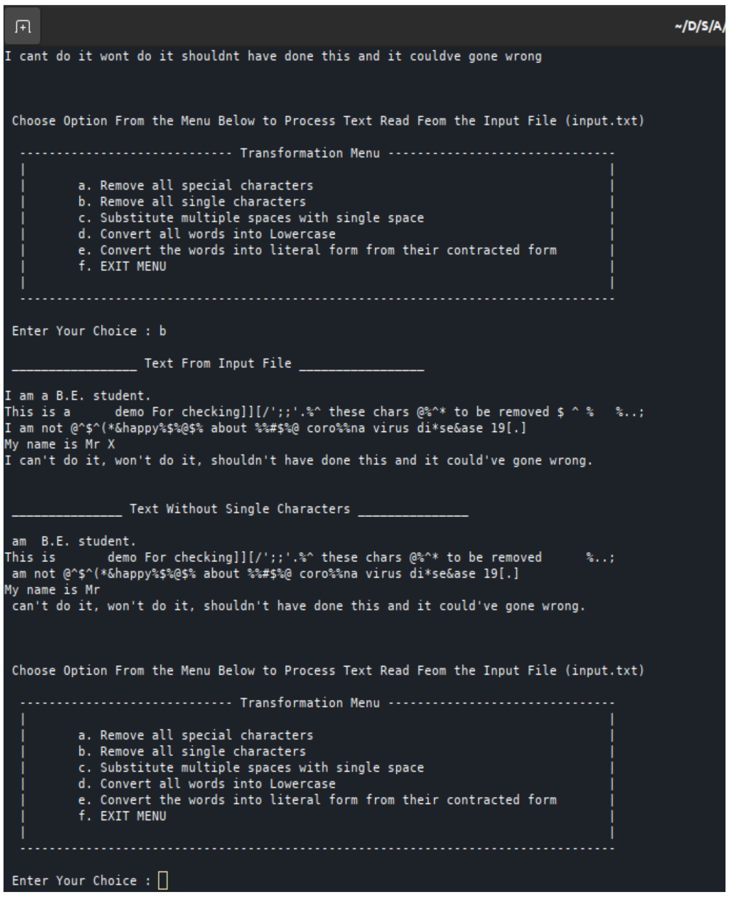
```
class TextProcessor:

    def __init__(self):
       """dictionary of words with contracted words as KEY and uncontracted words as VALUE"""
       self.contractions = {
            "ain't": "am not / are not",
            "aren't": "are not / am not",
            "can't": "cannot",
            "can't've": "cannot have",
            "'cause": "because",
            "could've": "could have",
            "couldn't": "could not",
            "couldn't've": "could not have",
            "didn't": "did not",
            "doesn't": "does not",
            "don't": "do not",
            "hadn't": "had not",
            "hadn't've": "had not have",
            "hasn't": "has not",
            "haven't": "have not",
            "he'd": "he had / he would",
            "he'd've": "he would have",
            "he'll": "he shall / he will",
            "he'll've": "he shall have / he will have",
            "he's": "he has / he is",
            "how'd": "how did",
            "how'd'y": "how do you",
            "how'll": "how will",
            "how's": "how has / how is",
            "i'd": "I had / I would",
            "i'd've": "I would have",
            "i'll": "I shall / I will",
            "i'll've": "I shall have / I will have",
            "i'm": "I am",
            "i've": "I have",
            "isn't": "is not",
            "it'd": "it had / it would",
            "it'd've": "it would have",
            "it'll": "it shall / it will",
            "it'll've": "it shall have / it will have",
            "it's": "it has / it is",
            "let's": "let us",
            "ma'am": "madam",
            "mayn't": "may not",
            "might've": "might have",
            "mightn't": "might not",
            "mightn't've": "might not have",
            "must've": "must have",
            "mustn't": "must not",
            "mustn't've": "must not have",
            "needn't": "need not",
            "needn't've": "need not have",
            "o'clock": "of the clock",
            "oughtn't": "ought not",
            "oughtn't've": "ought not have",
            "shan't": "shall not",
            "sha'n't": "shall not",
            "shan't've": "shall not have",
            "she'd": "she had / she would",
            "she'd've": "she would have",
            "she'll": "she shall / she will",
            "she'll've": "she shall have / she will have",
            "she's": "she has / she is",
            "should've": "should have",
            "shouldn't": "should not",
            "shouldn't've": "should not have",
            "so've": "so have",
            "so's": "so as / so is",
            "that'd": "that would / that had",
            "that'd've": "that would have",
            "that's": "that has / that is",
            "there'd": "there had / there would",
            "there'd've": "there would have",
            "there's": "there has / there is",
            "they'd": "they had / they would",
            "they'd've": "they would have",
            "they'll": "they shall / they will",
            "they'll've": "they shall have / they will have",
            "they're": "they are",
            "they've": "they have",
            "to've": "to have",
            "wasn't": "was not",
        "we'd": "we had / we would",
        "we'd've": "we would have",
        "we'll": "we will",
        "we'll've": "we will have",
        "we're": "we are",
        "we've": "we have",
        "weren't": "were not",
        "what'll": "what shall / what will",
        "what'll've": "what shall have / what will have",
        "what're": "what are",
        "what's": "what has / what is",
        "what've": "what have",
        "when's": "when has / when is",
        "when've": "when have",
        "where'd": "where did",
        "where's": "where has / where is",
        "where've": "where have",
        "who'll": "who shall / who will",
        "who'll've": "who shall have / who will have",
        "who's": "who has / who is",
        "who've": "who have",
        "why's": "why has / why is",
        "why've": "why have",
        "will've": "will have",
        "won't": "will not",
        "won't've": "will not have",
        "would've": "would have",
        "wouldn't": "would not",
        "wouldn't've": "would not have",
        "y'all": "you all",
        "y'all'd": "you all would",
        "y'all'd've": "you all would have",
        "y'all're": "you all are",
        "y'all've": "you all have",
        "you'd": "you had / you would",
        "you'd've": "you would have",
        "you'll": "you shall / you will",
        "you'll've": "you shall have / you will have",
        "you're": "you are",
        "you've": "you have"
    }


def del_special_char(self, s: str) -> str:
    """Remove special characters, i.e. characters which are not alphanumeric or whitespaces"""
    new_str = ''
    for element in s:
        if element.isalnum() or element.isspace():
            new_str += element
    print('\n _________________ Text From Input File _________________\n\n' + s)
    return new_str


def del_single_char(self, s: str) -> str:
    """Remove single characters, i.e. characters of length = 1 (a non-space character surrounded by
          whitespaces)"""
    new_str = ''
    for (i, ch) in enumerate(s):
        if not ch.isspace() and not ((i == 0 and s[i+1].isspace()) or (i == (len(s)-1) \
          and s[i-1].isspace()) or (s[i-1].isspace() and s[i+1].isspace())):
            new_str += ch
        elif ch.isspace():
            new_str += ch
    print('\n _________________ Text From Input File _________________\n\n' + s)
    return new_str


def sub_mul_space(self, s: list, t: str) -> str:
    """Remove multiple whitespaces and substitute those with a single whitespace"""
    new_str = ''
    for whole_line in s:
        char_list = whole_line.split()
        no_newline = ' '.join(char_list)
        new_str += '\n' + no_newline
    print('\n _________________ Text From Input File _________________\n\n' + t)
    return new_str


def convto_lower(self, s: str) -> str:
    """Convert all uppercase alphabatical characters to lowercase"""
    new_str = s.lower()
    print('\n _________________ Text From Input File _________________\n\n' + s)
    return new_str


def convto_literal(self, s: str) -> str:
    """Convert contracted forms of characters ('can't') to their uncontracted forms ('cannot')"""
    old_str = s
    for element in s.split():
        if element.lower() in self.contractions:
            not_contracted = self.contractions[element.lower()]
            s = s.replace(element, not_contracted)
    new_str = s
    print('\n _________________ Text From Input File _________________\n\n' + old_str)
    return new_str
```

```
class DriverCode:

    def proc_func(self):

       # creating a reference of class TextProcessor to implement different functionalities
        textP = TextProcessor()

        while True:
            print("\n\n\n Choose Option From the Menu Below to Process Text Read Feom the Input File (
              input.txt)\n")
            print("  ----------------------------- Transformation Menu ------------------------------- ")
            print("  |                                                                               | ")
            print("  |       a. Remove all special characters                                        | ")
            print("  |       b. Remove all single characters                                         | ")
            print("  |       c. Substitute multiple spaces with single space                         | ")
            print("  |       d. Convert all words into Lowercase                                     | ")
            print("  |       e. Convert the words into literal form from their contracted form       | ")
            print("  |       f. EXIT MENU                                                            | ")
            print("  |                                                                               | ")
            print("  --------------------------------------------------------------------------------- ")
            choice = input("\n Enter Your Choice : ")

            # using if-elif-else statement to pick different options
            if choice == 'a' or choice == 'A':
                try:
                    fin = open('input.txt', 'r')
                    lines = fin.read()
                    print('\n\n _______________ Text Without Special Characters _______________\n\n' +
                            textP.del_special_char(lines))
                except Exception as ex:
                    print('\n Exception Caught :', ex)
                finally:
                    fin.close()

            elif choice == 'b' or choice == 'B':
                try:
                    fin = open('input.txt', 'r')
                    lines = fin.read()
                    print('\n\n _______________ Text Without Single Characters _______________\n\n' +
                            textP.del_single_char(lines))
                except Exception as ex:
                    print('\n Exception Caught :', ex)
                finally:
                    fin.close()

            elif choice == 'c' or choice == 'C':
                try:
                    fin = open('input.txt', 'r')
                    file = open('input.txt', 'r')
                    lines = file.readlines()
                    text = fin.read()
                    print('\n\n _______________ Text Without Multiple Spaces _______________\n' +
                            textP.sub_mul_space(lines, text))
                except Exception as ex:
                    print('\n Exception Caught :', ex)
                finally:
                    fin.close()
                    file.close()

            elif choice == 'd' or choice == 'D':
                try:
                    fin = open('input.txt', 'r')
                    lines = fin.read()
                    print('\n\n _______________ Text Converted to LowerCase _______________\n\n' +
                            textP.convto_lower(lines))
                except Exception as ex:
                    print('\n Exception Caught :', ex)
                finally:
                    fin.close()

            elif choice == 'e' or choice == 'E':
                try:
                    fin = open('input.txt', 'r')
                    lines = fin.read()
                    print('\n\n _______________ Text Converted into LiteralForm _______________\n\n' +
                            textP.convto_literal(lines))
                except Exception as ex:
                    print('\n Exception Caught :', ex)
                finally:
                    fin.close()

            elif choice == 'f' or choice == 'F' or choice == '' or choice == '\n':
                print("\n EXITING MENU....EXITED.\n")
                break

            else:
                print("\n Choice OUT OF RANGE!\n Please Choose within {a, b, c, d, e, f}.")


dc_obj = DriverCode()


def main():
    dc_obj.proc_func()


if __name__ == '__main__':
    main()
```

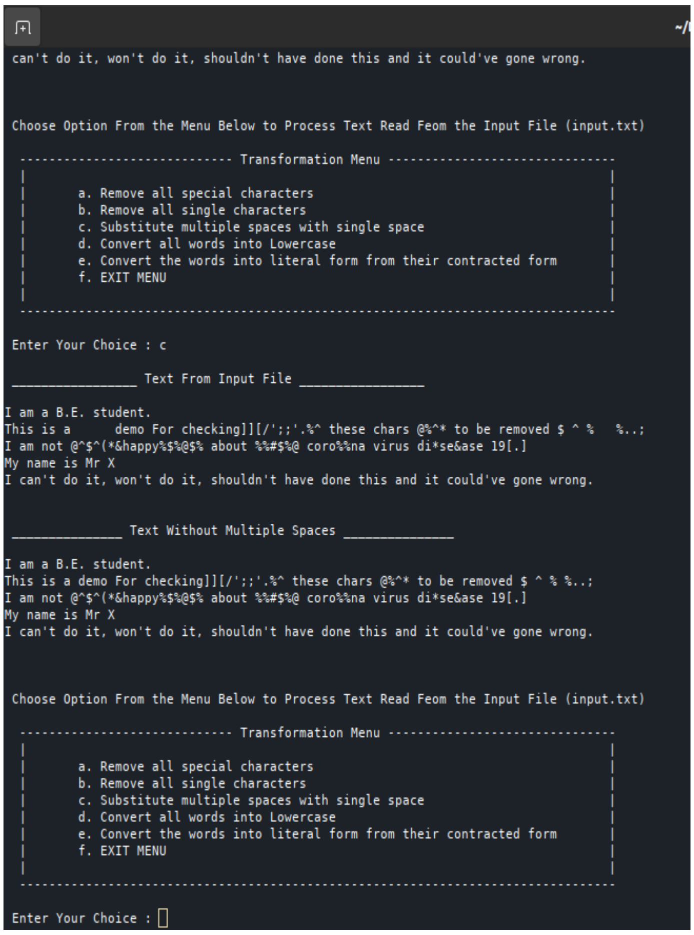
Output :



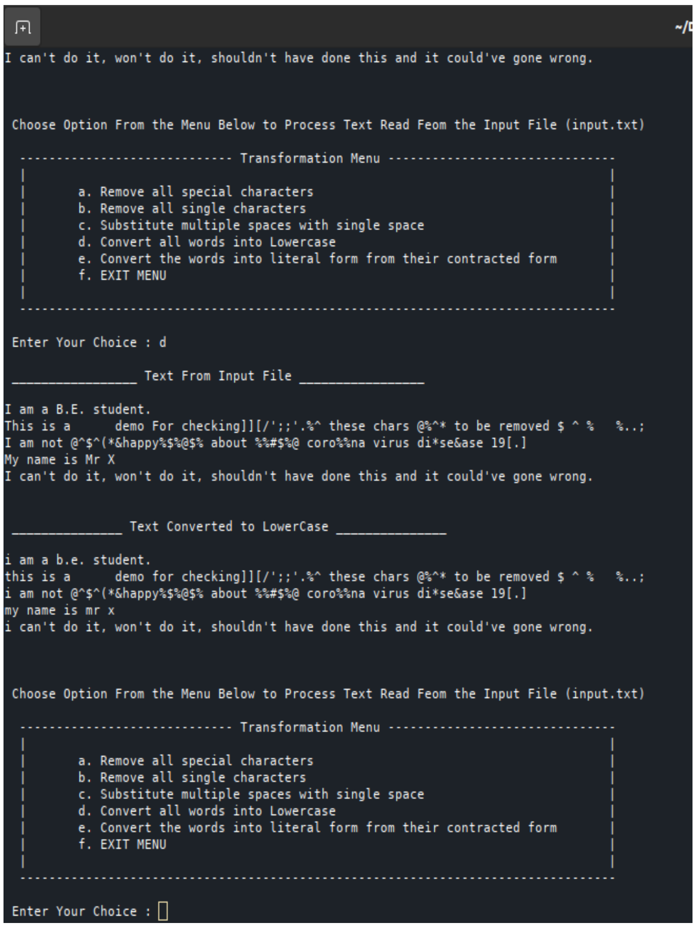
```
~/D/S/A/p/a2q1  python3 a2q1.py



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : a

 ________________ Text From Input File ________________

I am a B.E. student.
This is a      demo For checking]][/';;'.%^ these chars @%^* to be removed $ ^ %   %..;
I am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr X
I can't do it, won't do it, shouldn't have done this and it could've gone wrong.


 _______________ Text Without Special Characters _______________

I am a BE student
This is a      demo For checking these chars  to be removed
I am not happy about  corona virus disease 19
My name is Mr X
I cant do it wont do it shouldnt have done this and it couldve gone wrong



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : 
```



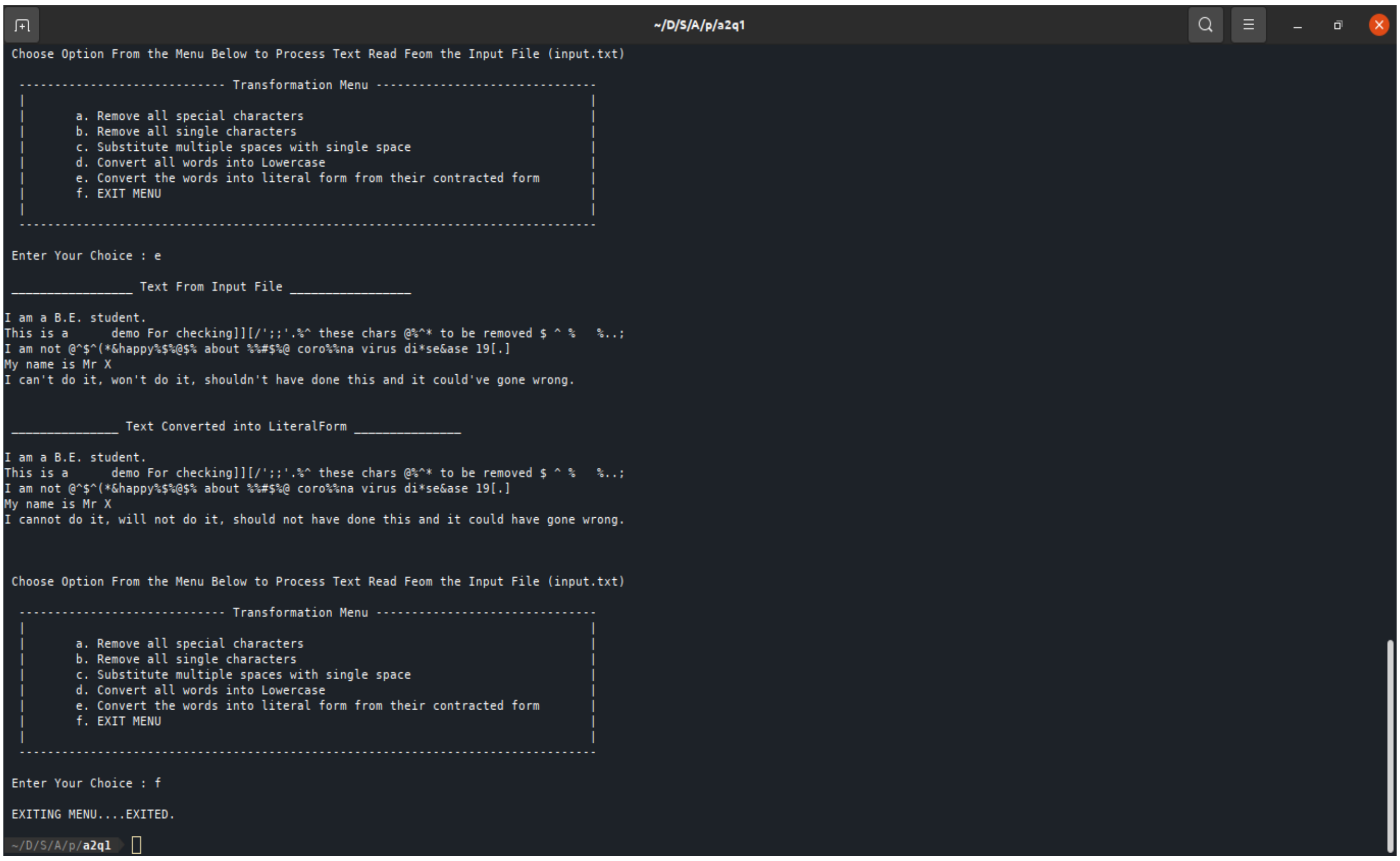
```
~/D/S/A/
I cant do it wont do it shouldnt have done this and it couldve gone wrong



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : b

 ________________ Text From Input File ________________

I am a B.E. student.
This is a      demo For checking]][/';;'.%^ these chars @%^* to be removed $ ^ %   %..;
I am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr X
I can't do it, won't do it, shouldn't have done this and it could've gone wrong.


 _______________ Text Without Single Characters _______________

 am  B.E. student.
This is       demo For checking]][/';;'.%^ these chars @%^* to be removed      %..;
 am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr
 can't do it, won't do it, shouldn't have done this and it could've gone wrong.



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : 
```



```
~/
 can't do it, won't do it, shouldn't have done this and it could've gone wrong.



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : c

 ________________ Text From Input File ________________

I am a B.E. student.
This is a      demo For checking]][/';;'.%^ these chars @%^* to be removed $ ^ %   %..;
I am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr X
I can't do it, won't do it, shouldn't have done this and it could've gone wrong.


 _______________ Text Without Multiple Spaces _______________

I am a B.E. student.
This is a demo For checking]][/';;'.%^ these chars @%^* to be removed $ ^ % %..;
I am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr X
I can't do it, won't do it, shouldn't have done this and it could've gone wrong.



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : 
```



```
~/
I can't do it, won't do it, shouldn't have done this and it could've gone wrong.



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : d

 ________________ Text From Input File ________________

I am a B.E. student.
This is a      demo For checking]][/';;'.%^ these chars @%^* to be removed $ ^ %   %..;
I am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr X
I can't do it, won't do it, shouldn't have done this and it could've gone wrong.


 _______________ Text Converted to LowerCase _______________

i am a b.e. student.
this is a      demo for checking]][/';;'.%^ these chars @%^* to be removed $ ^ %   %..;
i am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
my name is mr x
i can't do it, won't do it, shouldn't have done this and it could've gone wrong.



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  --------------------------- Transformation Menu ------------------------------
  |                                                                            |
  |       a. Remove all special characters                                     |
  |       b. Remove all single characters                                      |
  |       c. Substitute multiple spaces with single space                      |
  |       d. Convert all words into Lowercase                                  |
  |       e. Convert the words into literal form from their contracted form    |
  |       f. EXIT MENU                                                         |
  |                                                                            |
  ------------------------------------------------------------------------------

 Enter Your Choice : 
```



```
~/D/S/A/p/a2q1

 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  ---------------------------- Transformation Menu ------------------------------
  |                                                                              |
  |       a. Remove all special characters                                       |
  |       b. Remove all single characters                                        |
  |       c. Substitute multiple spaces with single space                        |
  |       d. Convert all words into Lowercase                                    |
  |       e. Convert the words into literal form from their contracted form      |
  |       f. EXIT MENU                                                           |
  |                                                                              |
  --------------------------------------------------------------------------------

 Enter Your Choice : e

 _________________ Text From Input File _________________

I am a B.E. student.
This is a      demo For checking]][/';;'.%^ these chars @%^* to be removed $ ^ %   %..;
I am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr X
I can't do it, won't do it, shouldn't have done this and it could've gone wrong.


 _______________ Text Converted into LiteralForm _______________

I am a B.E. student.
This is a      demo For checking]][/';;'.%^ these chars @%^* to be removed $ ^ %   %..;
I am not @^$^(*&happy%$%@$% about %%#$%@ coro%%na virus di*se&ase 19[.]
My name is Mr X
I cannot do it, will not do it, should not have done this and it could have gone wrong.



 Choose Option From the Menu Below to Process Text Read Feom the Input File (input.txt)

  ---------------------------- Transformation Menu ------------------------------
  |                                                                              |
  |       a. Remove all special characters                                       |
  |       b. Remove all single characters                                        |
  |       c. Substitute multiple spaces with single space                        |
  |       d. Convert all words into Lowercase                                    |
  |       e. Convert the words into literal form from their contracted form      |
  |       f. EXIT MENU                                                           |
  |                                                                              |
  --------------------------------------------------------------------------------

 Enter Your Choice : f

 EXITING MENU....EXITED.

~/D/S/A/p/a2q1
```

Q2: Implement one multi-threaded server with socket programming in python.

Sol :

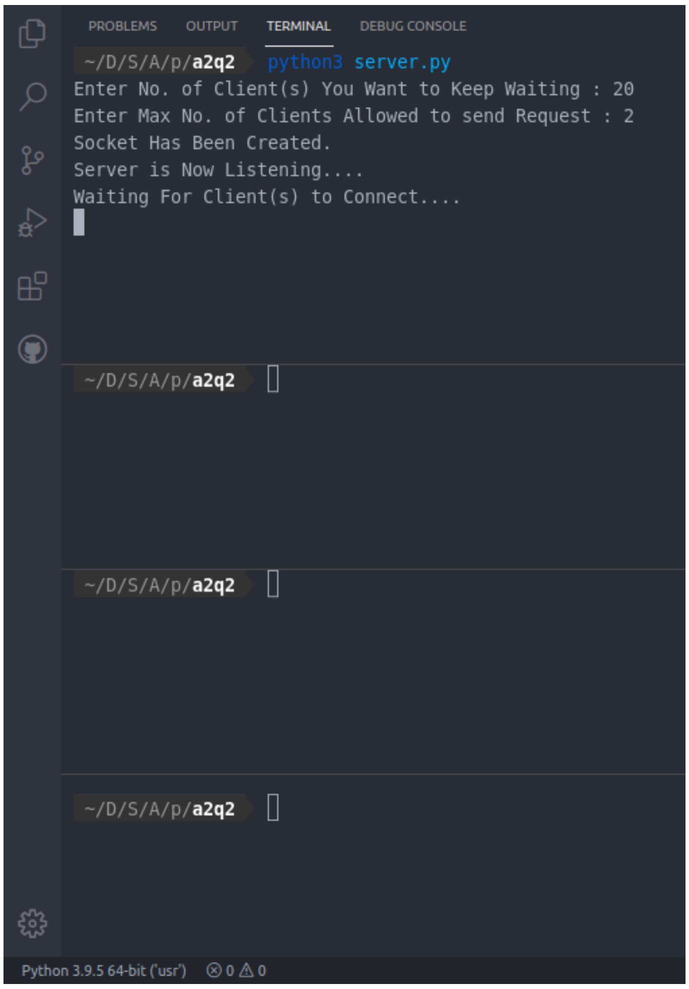
Server Side Program

```
# import required pacakges / modules
import sys
import socket
from _thread import start_new_thread


def start_server():
    host = '127.0.0.1'
    port = 2021                                          # arbitrary non-privileged port
    ThreadCount = 0
    ServerSideSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

    try:
        ServerSideSocket.bind((host, port))
        n = int(input("Enter No. of Client(s) You Want to Keep Waiting : "))
        max = int(input("Enter Max No. of Clients Allowed to send Request : "))
        ServerSideSocket.listen(n)                       # queue up to 'n' requests
        print("Socket Has Been Created." + "\nServer is Now Listening...." +
              "\nWaiting For Client(s) to Connect....")

    except socket.error as e:
        print("Bind Failed! Error : " + str(sys.exc_info()))
        print("Exception Caught : " + str(e))
        ServerSideSocket.close()
        sys.exit()

    except ValueError as v:
        print("Exception Caught : " + str(v))
        print("Server Has Been Terminated." +
              "\nEnter Integer Only From Next Time.")
        ServerSideSocket.close()
        sys.exit()

    while ThreadCount <= max:
        client, address = ServerSideSocket.accept()
        ip, port = str(address[0]), str(address[1])
        print("\nConnected to : " + ip + ':' + port)
        start_new_thread(client_thread, (client, ))
        ThreadCount += 1
        print("Thread Number = Client Number : " + str(ThreadCount))

    # ServerSideSocket.close()
    print(f"\nMore Than ({max}) Clients Can't Send Request(s).")
    print("Server Has Been Closed.")


def client_thread(connection):
    connection.send(str.encode("Server is Working : "))
    while True:
        client_request = connection.recv(2048)
        response = "Server Response : " + client_request.decode('utf-8')
        if not client_request:
            break
        connection.sendall(str.encode(response))
    connection.close()


def main():
    start_server()

if __name__ == "__main__":
    main()
```

Client Side Program

```
# import required pacakges / modules
import sys
import socket


def start_clients():
    host = '127.0.0.1'
    port = 2021
    ClientSideSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

    try:
        ClientSideSocket.connect((host, port))
    except socket.error as e:
        print("Server is Currently Inactive. Connection Failed!\n\nError : " +
              str(sys.exc_info()))
        print("\nException Caught : " + str(e))
        sys.exit()

    print("Server is Currently Active.\nSend Client-Request to Get Response Fom Server.")

    server_response = ClientSideSocket.recv(1024)

    while True:
        request = input("\nClient Request (Enter 0 to Exit) : ")
        if request == '0':
            ClientSideSocket.close()
            print("\nConnection Has Been Closed.\nClient Has Exited.")
            sys.exit()

        try:
            ClientSideSocket.sendall(request.encode("utf-8"))
            server_response = ClientSideSocket.recv(1024)
            print(server_response.decode("utf-8"))

        except Exception as e:
            print("Exception Caught : " + str(e))
            print("Server Has Been Closed.")
            ClientSideSocket.close()
            sys.exit()


def main():
    start_clients()


if __name__ == "__main__":
    main()
```

Output :


PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

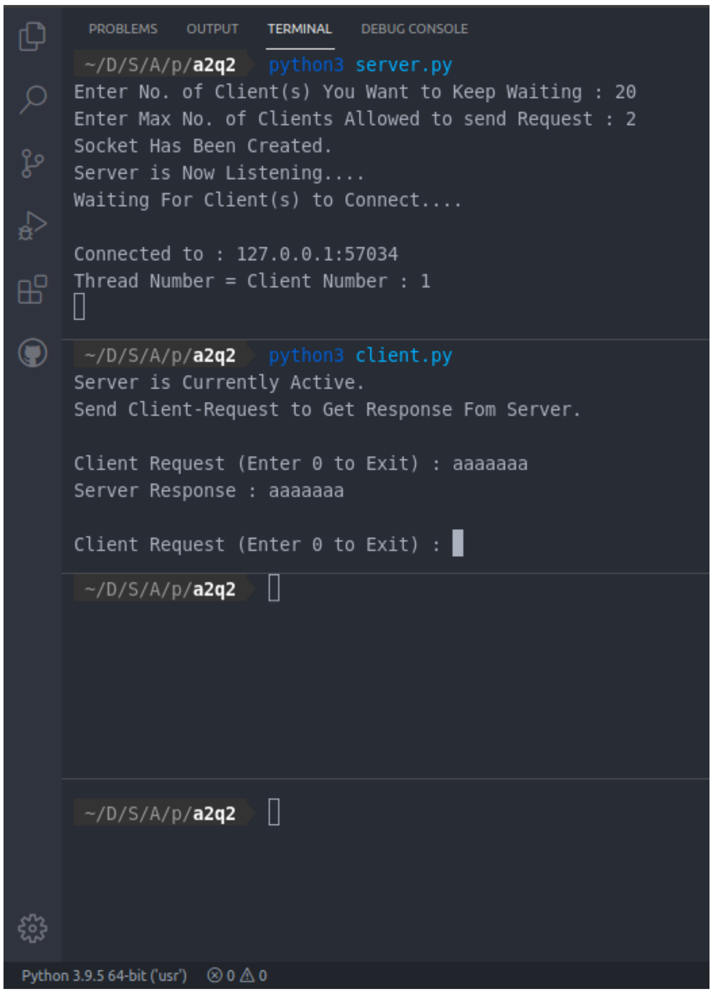
```
~/D/S/A/p/a2q2  python3 server.py
Enter No. of Client(s) You Want to Keep Waiting : 20
Enter Max No. of Clients Allowed to send Request : 2
Socket Has Been Created.
Server is Now Listening....
Waiting For Client(s) to Connect....
```

```
~/D/S/A/p/a2q2
```

```
~/D/S/A/p/a2q2
```

```
~/D/S/A/p/a2q2
```

Python 3.9.5 64-bit ('usr') ⊗ 0 ⚠ 0


PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

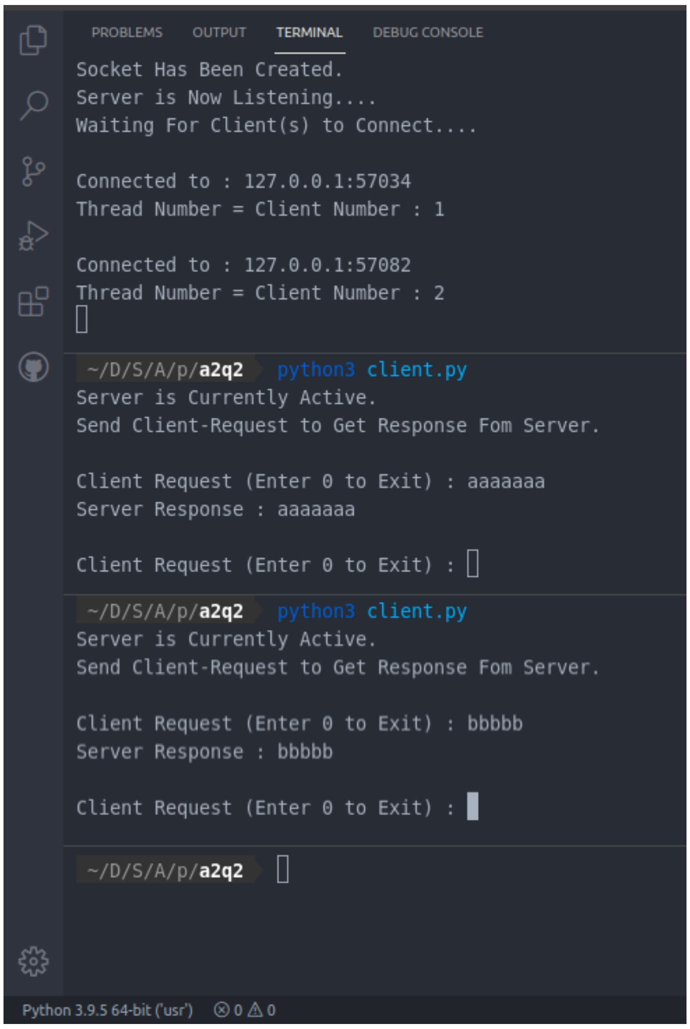
```
~/D/S/A/p/a2q2  python3 server.py
Enter No. of Client(s) You Want to Keep Waiting : 20
Enter Max No. of Clients Allowed to send Request : 2
Socket Has Been Created.
Server is Now Listening....
Waiting For Client(s) to Connect....

Connected to : 127.0.0.1:57034
Thread Number = Client Number : 1
```

```
~/D/S/A/p/a2q2  python3 client.py
Server is Currently Active.
Send Client-Request to Get Response Fom Server.

Client Request (Enter 0 to Exit) : aaaaaaa
Server Response : aaaaaaa

Client Request (Enter 0 to Exit) :
```

```
~/D/S/A/p/a2q2
```

```
~/D/S/A/p/a2q2
```

Python 3.9.5 64-bit ('usr') ⊗ 0 ⚠ 0


PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

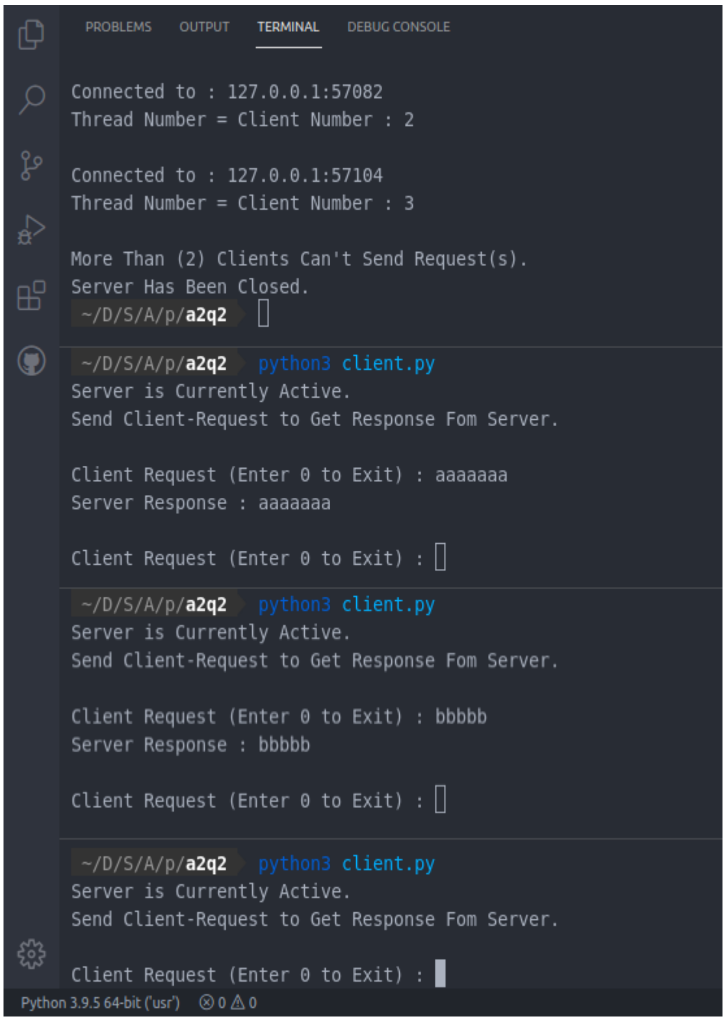
```
Socket Has Been Created.
Server is Now Listening....
Waiting For Client(s) to Connect....

Connected to : 127.0.0.1:57034
Thread Number = Client Number : 1

Connected to : 127.0.0.1:57082
Thread Number = Client Number : 2
```

```
~/D/S/A/p/a2q2  python3 client.py
Server is Currently Active.
Send Client-Request to Get Response Fom Server.

Client Request (Enter 0 to Exit) : aaaaaaa
Server Response : aaaaaaa

Client Request (Enter 0 to Exit) :
```

```
~/D/S/A/p/a2q2  python3 client.py
Server is Currently Active.
Send Client-Request to Get Response Fom Server.

Client Request (Enter 0 to Exit) : bbbbb
Server Response : bbbbb

Client Request (Enter 0 to Exit) :
```

```
~/D/S/A/p/a2q2
```

Python 3.9.5 64-bit ('usr') ⊗ 0 ⚠ 0


PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
Connected to : 127.0.0.1:57082
Thread Number = Client Number : 2

Connected to : 127.0.0.1:57104
Thread Number = Client Number : 3

More Than (2) Clients Can't Send Request(s).
Server Has Been Closed.
~/D/S/A/p/a2q2
```

```
~/D/S/A/p/a2q2  python3 client.py
Server is Currently Active.
Send Client-Request to Get Response Fom Server.

Client Request (Enter 0 to Exit) : aaaaaaa
Server Response : aaaaaaa

Client Request (Enter 0 to Exit) :
```

```
~/D/S/A/p/a2q2  python3 client.py
Server is Currently Active.
Send Client-Request to Get Response Fom Server.

Client Request (Enter 0 to Exit) : bbbbb
Server Response : bbbbb

Client Request (Enter 0 to Exit) :
```

```
~/D/S/A/p/a2q2  python3 client.py
Server is Currently Active.
Send Client-Request to Get Response Fom Server.

Client Request (Enter 0 to Exit) :
```

Python 3.9.5 64-bit ('usr') ⊗ 0 ⚠ 0