LONDON METROPOLITAN UNIVERSITY

islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CS4059NI Fundamental of Computing

Assessment Weightage & Type

60% Individual Coursework

Year and Semester

2021-22 Spring

Student Name: Saugat Basnet

Group: N7

London Met ID:22015870

College ID:np01nt4s220042

Assignment Due Date:

Assignment Submission Date:

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Acknowledgment

Firstly, I would like to be very thankful to our module lecturer Mr. Ayush Pandey for providing us with a platform to help develop skills. This project has given us the opportunity to improve our skills in problem-solving and doing research for solving the problems which arise while working on the project. Secondly, I would like to thank Islington college and London met university for providing us with an excellent environment for learning new skills and also helping us to learn about new technology. Lastly, I would like to thank my friends who suggested to me and helped me solve some of the problems that arise during my project.

Table of Contents

Table of Tables

Table of Figures

1.1 Introduction

A back-end program for a clothing rental company that can rent a variety of clothes based on the needs of the consumer and collect a fee is being developed for this assignment in fundamentals of computing. Using this program, a consumer can rent items from a list of clothing options at the rental shop. A bill containing the total cost and the custom that has been rented will be printed after the rental process is complete. Additionally, if the custom is rented for more than 5 days during the return procedure, a fine will be assessed against the consumer and added to the bill.

The software is developed using the programming language called python. Python is a general purpose, dynamic, high-level, and interpreted programming language. (Java T point, n.d.) It allows the development of programs using an object-oriented approach. It offers a large number of high-level data structures and is straightforward and simple to learn.

1.2 Goals and Objectives

The primary goal of this assignment is to create back-end software for a clothing rental business that will allow users to choose items from a list of clothing and enter the number of items they need. A bill is saved in a text file when the renting procedure is finished and the client's information is included. This bill is utilized when the consumer returns. In addition, a fine equal to the number of days the client was rented will be assessed if the rental period exceeds 5 days.
The program ought to be broken up into many functions, each of which should play a unique part in the overall scheme of things. The program needs to be carefully organized, well-commented and labelled.

2. Discussion and Analysis

2.1 Algorithm

An algorithm is a set of guidelines for resolving a dilemma or carrying out a task. A recipe, which consists of detailed directions for creating a dish or meal, is a typical illustration of an algorithm. Algorithms are used by every computerized equipment to carry out its operations in the form of hardware- or software-based routines.

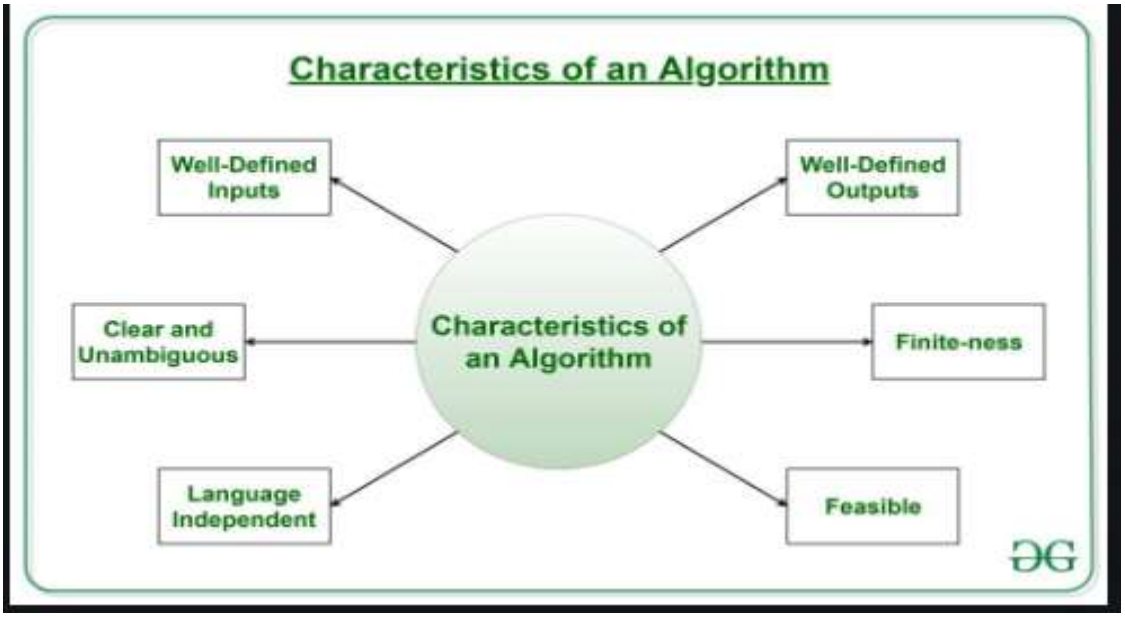


Figure 1 Algorithm

The algorithm of the given program is given below:

Step 1: Start

Step 2: create a function that contains the details of the shop.

Step 3: create a function to welcome the user

Step 4: call both the functions to the display

Step 5: call the text file containing the list of the clothes

Step 6: give the user the option to select between renting or returning the cloth or exit

Step 7: if the user selects the renting start the renting process

For Renting Process

Step 8: define a function for renting the clothes

Step 9: display the list of clothes in form of a table with qty, brand, and price

Step 10: change the list of clothes in the dictionary providing each with a unique key and details of the clothes as value

Step 11: give the user option to select the clothes and quantity required

Step 12: add the selected cloth key to an ArrayList

Step 13: give the user option to continue with the renting process or add more cloth to the list

Step 14: After the selection is completed display the list of selected clothes

Step 15: ask the user to input the name of the customer

Step 16: create a text file using the name of the customer

Step 17: add the details of the customer and a list of clothes to the text file

Step 18: create a function to total the price of the purchase using the for loop

Step 19: add the total price to the text file

Step 20: print the details of the text file in the terminal

For Returning Process

Step 21: ask the user the name of the customer as in the bill

Step 22: search the bill having the name of the customer

Step 23: ask the user for how many days the clothes were rented

Step 24: if the rented days have exceeded 5 days fine shall be charged

Step 25: call the function which adds the fine to the bill

Step 26: print the total and bill of the customer

2.2 Flowchart

An algorithm is graphically represented by a flowchart. It is frequently used by programmers as a technique for planning programs to address issues. It uses interconnected symbols to represent the movement of information and processing. "Flowcharting" is the process of creating a flowchart for an algorithm.

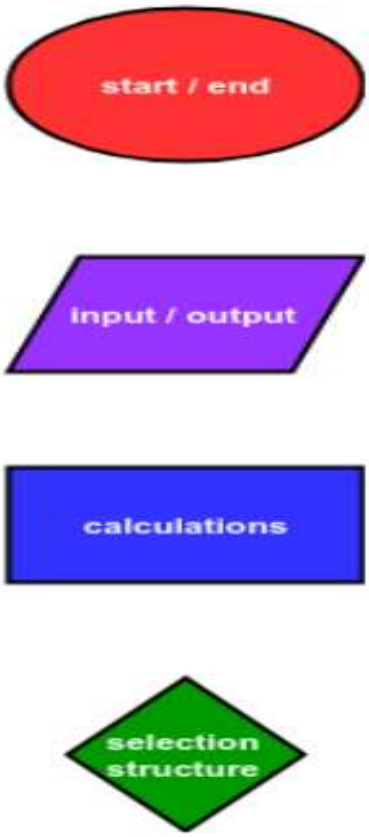


Figure 2 flow chart

Flowchart of the given program

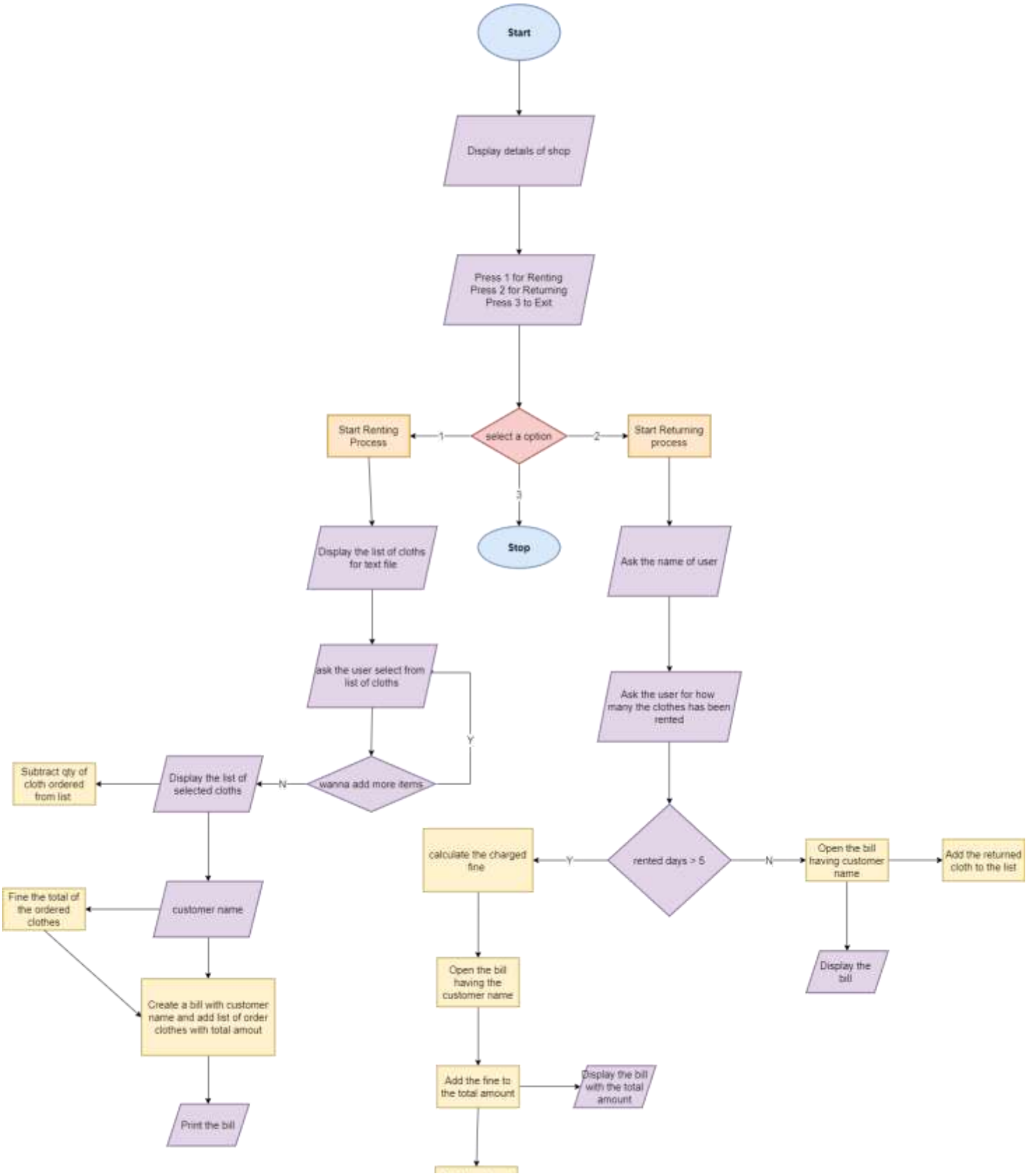


Figure 4 Flow chart of the Program

2.3 Pseudocode

Python pseudocode is more like an algorithmic representation of the code involved. This means when a code is expected to be formulated it cannot be directly drafted. The code will need to be first generated into a Python pseudocode and then it needs to be formulated into an actual code. (Eduba, n.d.)

What a Python pseudocode actually does is create an algorithmic representation that resembles an English sentence. A syntax-free representation of code is what the Python pseudocode is, to put it simply. As a result, the Python pseudocode contains no code. The algorithmic logic must closely match the Python pseudocode in every way. Each line of the Python pseudocode must be able to be translated proportionally into actual code. For those engaged who are not technically savvy, Python pseudocode makes the actual code easier to understand.

```
BEGIN
        FUNCTION shop_details(){
            PRINT details of the shop
             }

      FUNCTION welcome_(){
            PRINT welcome to the user
            PRINT press 1 to rent
            PRINT press 2 to return
            PRINT press 3 to exit
        }


CALL the function shop_details()
CALL  the function welcome_()
```

FUNCTION option_() {
DECLARE loop to True
WHILE loop is True
TRY:
INPUT from the user in integer form
EXCEPT:
PRINT select according to the option
CONTINUE
IF the user press 1
CALL function cloths_select()
CALL function dictionary_()
CALL function welcome_()
END IF user press 2
CALL function return_()
END IF user press 3
Exit from the loop
SET loop to False
ELSE:
PRINT invalid input
Continue the loop
}
CALL function option_()

FUNCTION cloth_Select():
DISPLAY the list of clothes from text file in form of table
GIVE key values to the cloths

}

FUNCTION dictionary_()

OPEN the text file containing the list of cloths

ADD the list of the cloths to the dictionary

CALL function cloth_selection()

}

FUNCTION cloth_selection(dic){

DECLARE list named list_

DECLARE list named list_qty

TRY:

TAKE input from the user asking to select the cloth according to the given key values in integer form

EXCEPT:

PRINT asks the user to input the correct values

CONTINUE

TAKE input from the user asking the quantity of cloth required

If the cloth is available in the store

THEN continue with the renting process

ASK the user if they want to add more item

IF yes **THEN** repeat the process

ADD the key values of the selected cloth to the list

ADD also add the quantity of the cloth to the list

IF no

THEN

PRINT the list of selected cloths

CALL the function customer_details()

}

FUNCTION Customer_details()
{
IMPORT list_ and list_qty from custom_selection()
ASK the user's name
MAKE a text file with the input name
CREATE a bill using the details of the customer and a list of ordered cloths
CALL function price_total()
IMPORT total price of the clothes from price_cloths
ADD the total price to the bill
WRITE the bill to the created text file
PRINT the bill in terminal
}
FUNCTION price_total(){
SET sum to 0
FOR i and j in list of cloth price and list of quantity ordered
ADD prices of clothes
RETURN sum

FUNCTION Return_(){
ASK the user name which was in the bill
OPEN the text file named as user given name
ASK the user number of days the clothes has been rented
IF the clothes have been rented for more than 5 days
C**ALL** the function charged_fine()
PRINT the total price with the charged amount
ELSE
OPEN the text file named as user given name
READ the file and give the total amount
}

FUNCTION charge_fine(){

OPEN text file named as user given name

EXTRACT the total price from the bill

IMPORT the number of days from the function return

CALCULATE the fine and add to the total price according to the number of days

}

END

2.4 Data Structure

Data Structure is crucial to organizing, managing, and storing data since doing so makes it easier to access and more effective to modify. You can arrange your data using data structures so that you can store groups of data, relate them, and carry out actions on them as necessary.

Type of Data Structure

Python has implicit support for Data Structures that enable you to store and access data. These structures are called List, Dictionary, Tuple, and Set. (edureka, n.d.)

Python allows the users may design their own Data Structures, they have complete control over how they work. The most common data structures, which are also available in other programming languages, include Stack, Queue, Tree, Linked List, and others.

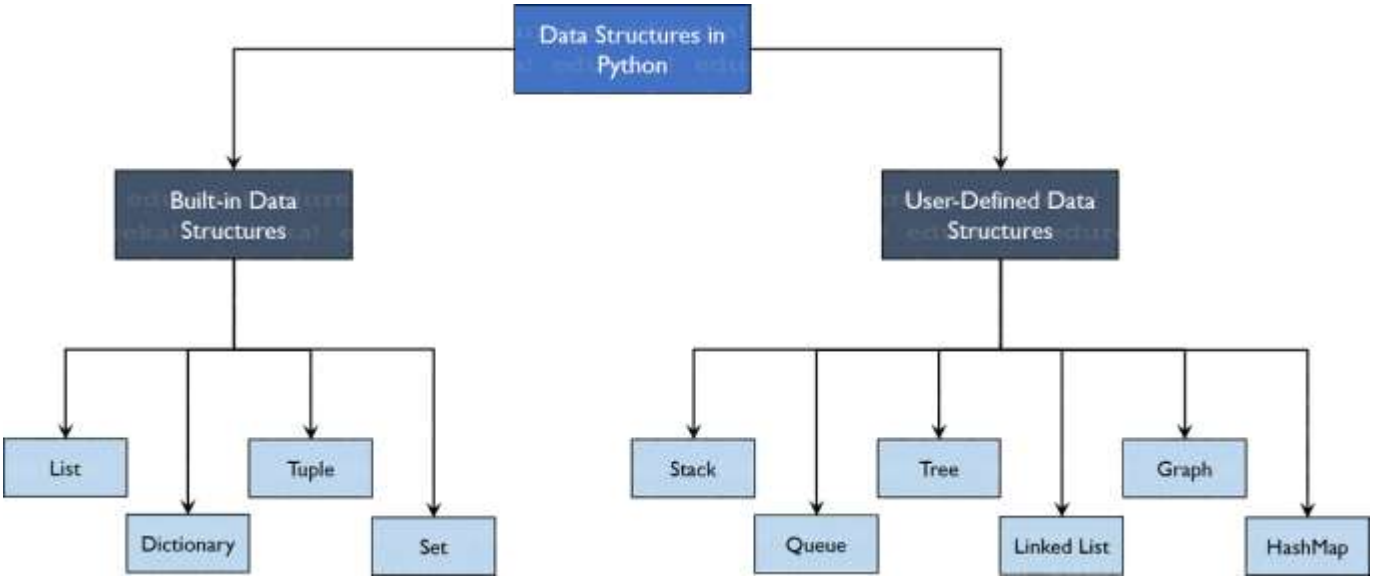


Figure 5 Data Structure

- **Built-in Data Structure**

I. **Lists**

Lists are used to sequentially store data of various data kinds. Every item in the list, also known as the Index, has an address allocated to it. The index value begins at 0 and continues until the final component, which is known as the positive index.

```
list_ = [1,2,3,4]
print(list)
```

II. **Dictionary**

Key-value pairs are stored in dictionaries. Imagine a phone book that has thousands of names and their matching numbers added to it in order to better understand. Name and Phone Numbers, which are referred to as the keys, are now the constant values in this situation. And the values that have been supplied to the keys are the various names and phone numbers.

```
my_dict = {} #empty dictionary
print(my_dict)
my_dict = {1: 'Python', 2: 'Java'} #dictionary with elements
print(my_dict)
```

III. **Sets**

Sets are collections of distinct, unordered components. The data would only be entered into the set once, even if it were repeated more than once. It is similar to the sets you have studied in mathematics. The operations are identical to how they are with arithmetic sets. An example program would make things clearer for you.

```
1 my_set = {1, 2, 3, 4, 5, 5, 5} #create set
2 print(my_set)
```

IV. **Tuples**

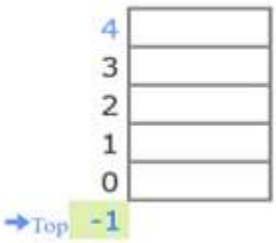
The only difference between tuples and lists is that once the data has been entered, it cannot be modified under any circumstances. The only exception is if the tuple's data is mutable, in which case it can be altered. You can better comprehend with the aid of the sample software.

```
1  my_tuple = (1, 2, 3) #create tuple
2  print(my_tuple)
```

- **User-defined data structures**

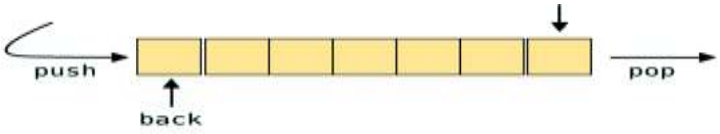
I. Stack

Stacks are linear Data Structures that are based on the principle of Last-In-First-Out (LIFO) where data that is entered last will be the first to get accessed. (edureka, n.d.)



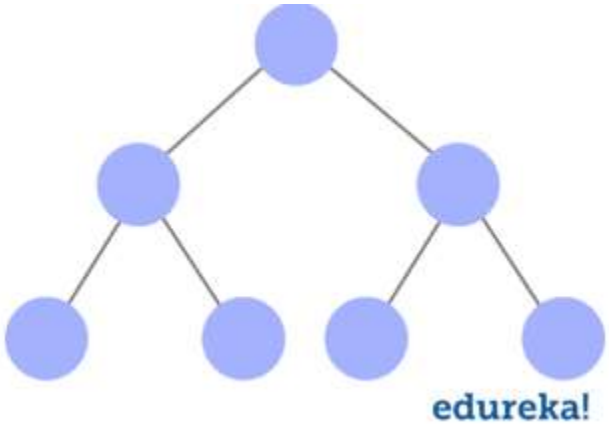
II. Queue

The data input first will be accessed first in a queue, which is likewise a linear data structure based on the FIFO principle. It is constructed using an array structure and has operations that may be carried out from either the head-tail or front-back ends of the queue. The terms "En-Queue" and "De-Queue" refer to operations like adding and removing elements, respectively, and accessing the elements is possible.



III. Tree

Non-linear data structures called trees have a root and nodes. The additional data points that are available to us are called nodes, and the root is the node from whence the data originated. The parent node comes before, while the child node comes after.



IV. Linked List

Linked lists are linear Data Structures that are not stored consequently but are linked with each other using pointers. (edureka, n.d.) The node of a linked list is composed of data and a pointer called next. These structures are most widely used in image viewing applications, music player applications and so forth. (edureka, n.d.)

V. Graph

Data collections of points known as vertices (nodes) and edges are stored in graphs (edges). The most realistic representation of a real-world map may be said to be a graph. They are employed to determine the various cost-to-distances between the various nodes, also known as data points, and afterward determine the least path. Numerous programs, including Google Maps, Uber, and many others, employ graphs to determine the shortest route and maximize revenues.

2.5 Data Structure used in this Assignment

Two types of data structures are being used for this program one being a list and another dictionary. Both data structures are used multiple times in this program and play a very crucial role in this program.

Dictionary

Dictionary is used to extract the details of the cloth from a text file and store it a certain dictionary by giving key values to the cloths.

```
def dictionary_():# Extracting details of the cloths from the text file and adding it to the dictionary as values for keys
    f = open("clothes.txt","r")
    dic ={}
    id = 1
    for line in f :
        line = line.replace("/n" ,"")
        dic.update({id:line.split(",")})
        id = id +1
    f.close()
    cloth_selection(dic)
```

Figure 6 Dictionary

```
{1: ['Jeans     ', ' Addidas ', ' $20.40', '10\n'], 2: ['Khurtha ', 'Nike      ', '$30.50', '50\n'], 3: ['Track     ', 'Gucci     ', '$12.50', '60\n'], 4: ['Jack
et     ', 'Essential ', '$30.20', '12\n'], 5: ['Sari        ', 'Kolkata  ', '$21.30', '30\n'], 6: ['pant      ', 'Levi      ', '$40.50', '5\n'], 7: ['T-Shirt
', 'Hm     ', '$10.40', '12\n'], 8: ['Shirt       ', 'polo    ', '$22.30', '11']}
Press Y if this in your final selection or N for adding  another item:
```

Lists

The list is used multiple times in the program like storing the ordered quantity of the cloth and ordered clothes key values.

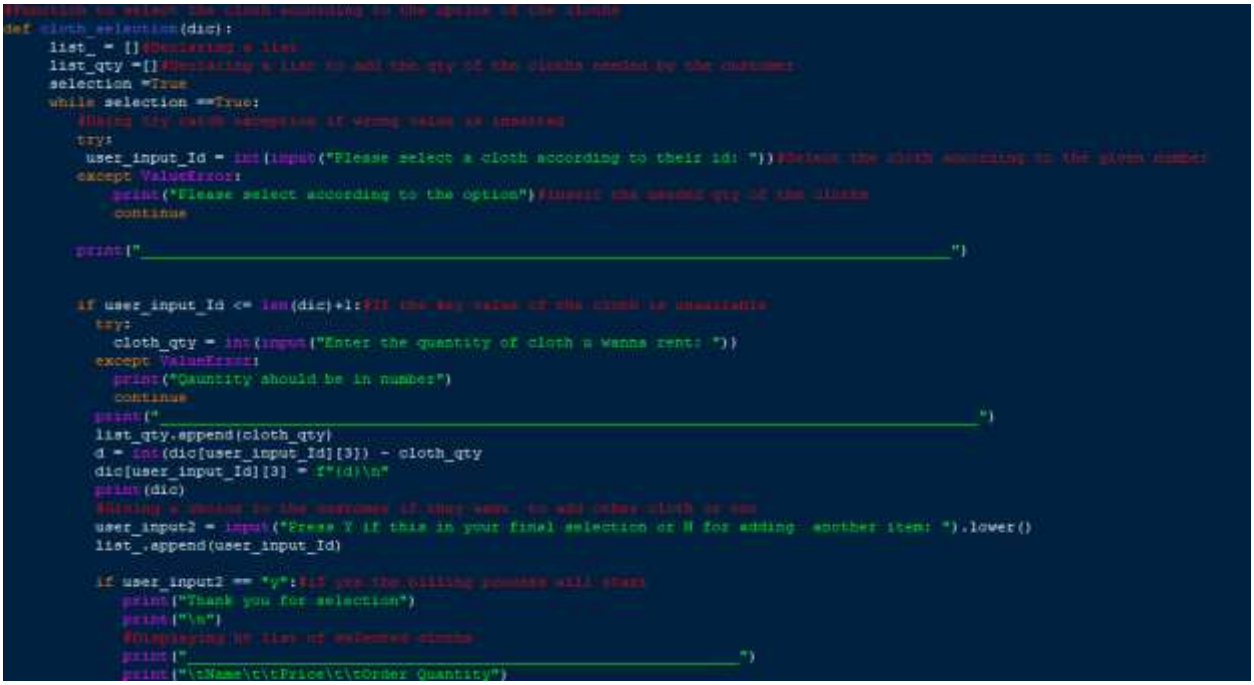


Figure 7 List

3.1 Program

The developed program of custom rental shops enables the user to rent clothes from the rental shop. When the program is run, a message is displayed showing the details of the shop and welcoming the user to the program with the 3 options to select (Press 1 for renting, press 2 for returning, and press 3 to exit). If the user selects from any of these three options, the program will continue as per the option or any other input is considered invalid input.

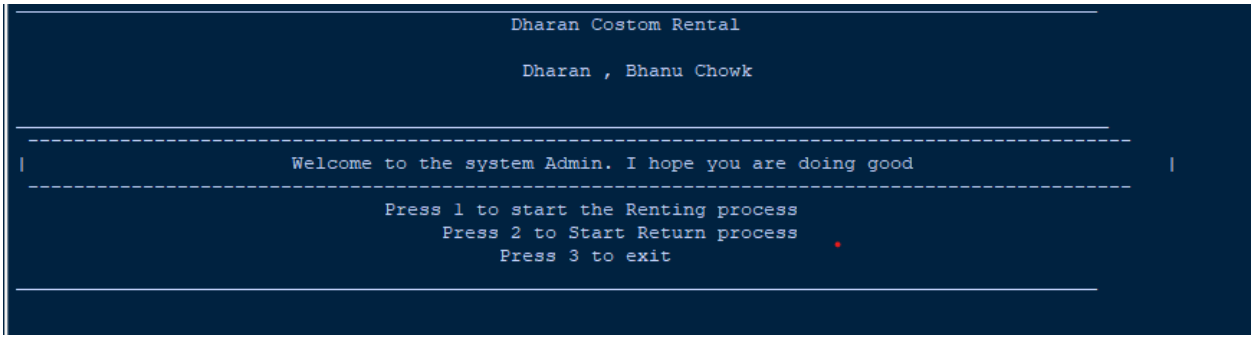


Figure 8 Program welcome

Renting

If the customer presses 1 for renting process a table showing list of clothes, price, brand, and available quantity is displayed on the terminal, these details of the is imported from the text file containing these details.

```
Displaying all the details of clothes-------------------------------------------------------------------

-----------------------------------------------------------------------------------------------
ID              Costom Name             Brand           Rent Price              Quantity
-----------------------------------------------------------------------------------------------
1               Jeans                    Addidas                $20.40          20
2               Khurtha                 Nike                    $30.50          50
3               Track                   Gucci                   $12.50          60
4               Jacket                  Essential               $30.20          12
5               Sari                    Kolkata                 $21.30          30
6               pant                    Levi                    $40.50          5
7               T-Shirt                 Ktm                     $10.40          12
8               Shirt                   polo                    $22.30          11
-----------------------------------------------------------------------------------------------
```

Figure 9 Program cloth display

After showing the list of clothes, the program gives the option to the user to select the cloth according to the key ID of the clothes. If the given id is not present in the table a message showing select again is displayed.

```
Displaying all the details of clothes-------------------------------------------------------------------

-----------------------------------------------------------------------------------------------
ID              Costom Name             Brand           Rent Price              Quantity
-----------------------------------------------------------------------------------------------
1               Jeans                    Addidas                $20.40          20
2               Khurtha                 Nike                    $30.50          50
3               Track                   Gucci                   $12.50          60
4               Jacket                  Essential               $30.20          12
5               Sari                    Kolkata                 $21.30          30
6               pant                    Levi                    $40.50          5
7               T-Shirt                 Ktm                     $10.40          12
8               Shirt                   polo                    $22.30          11
-----------------------------------------------------------------------------------------------
Please select a cloth according to their id: 10
_______________________________________________________________________________________________
Please select a cloth according to their id: 10
_______________________________________________________________________________________________
```

Figure 10 Program Exception

After the correct id is inserted the program asks the user to enter the required quantity of the clothes.

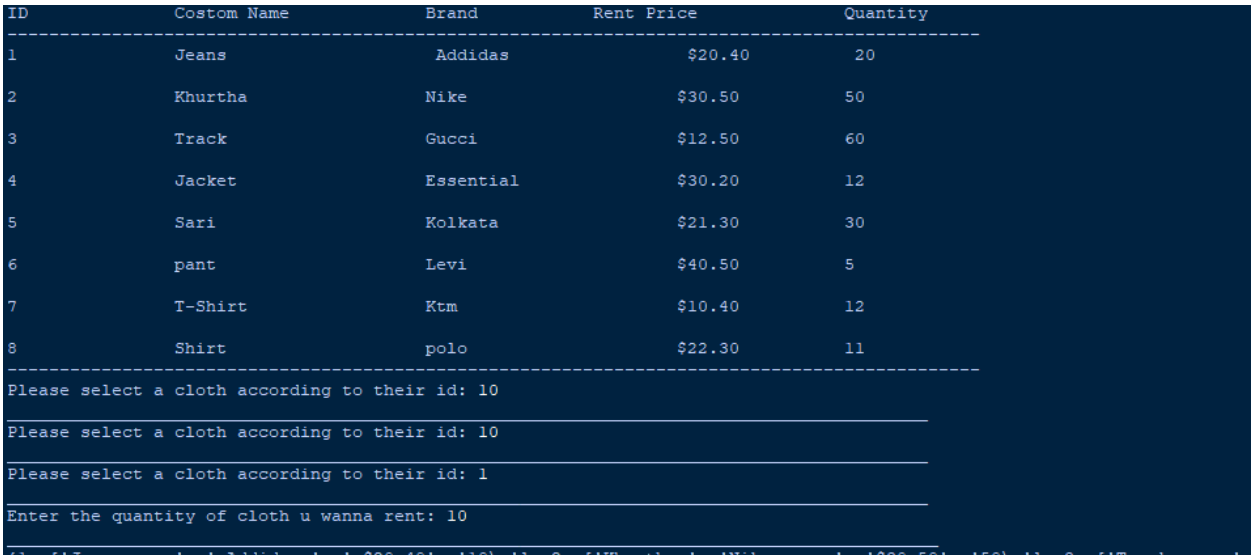


Figure 11 Program select Option

If the customer wants to add more items to the list, the program enables the user to do so.

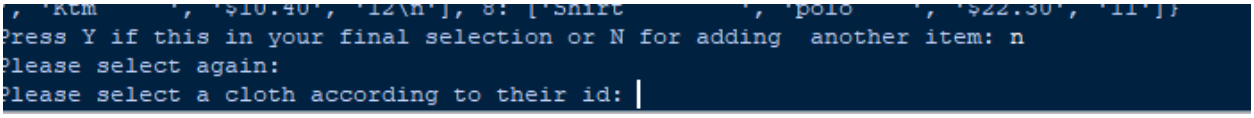


Figure 12 Program quantity

After the user selects the clothes, the program displays the selected cloths in list and ask the user to input the name of the user to make the bill.

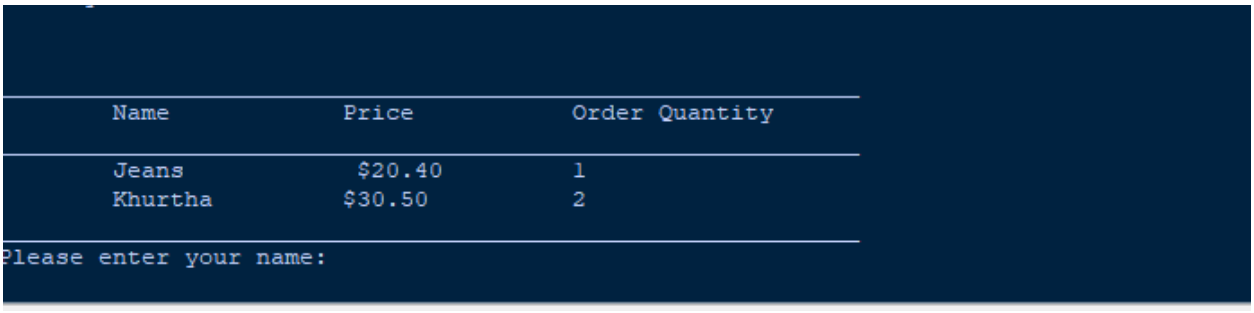


Figure 13 Program display selected Iteam

This name is used to make the bill of the customer and save it as text file for future use.

```
Please enter your name:
Saugat
--------------------------------------------------------------------------------

               Customer Details

--------------------------------------------------------------------------------

                Name of customer: saugat

                                    2022-08-26 00:36:24.386526

________________________________________________________________________

Jeans     ( Addidas )..................................................... 20.4              qty:1

Khurtha (Nike      )..................................................... 30.5              qty:2

________________________________________________________________________

 ..............................                                    Total:$81.4
```

Figure 14 Program display bill

The bill is printed in the terminal and the same bill is saved in a text file.

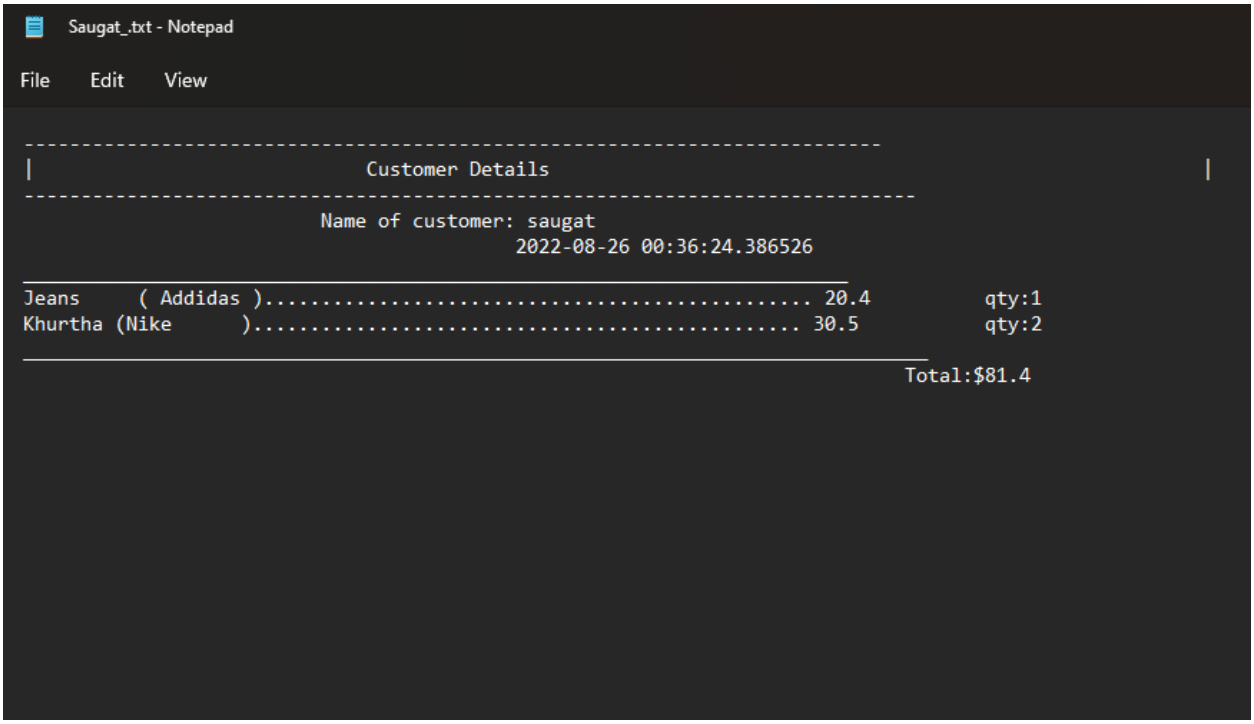


Figure 15 Bill in Text

After the renting process is completed the program will restart again for another customer.

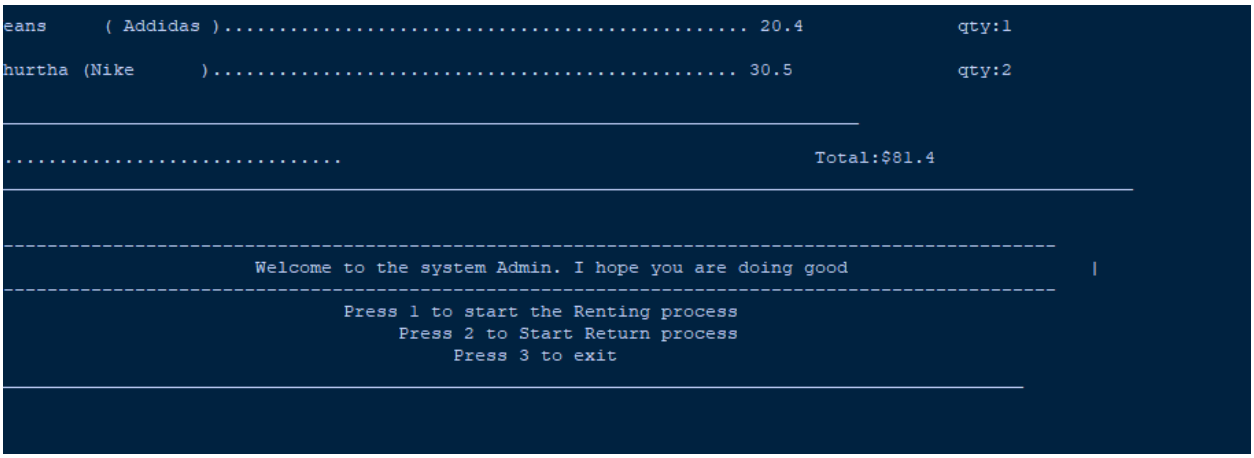


Figure 16 Loop

Returning

If the customer selects the returning process the program will continue with the returning process.

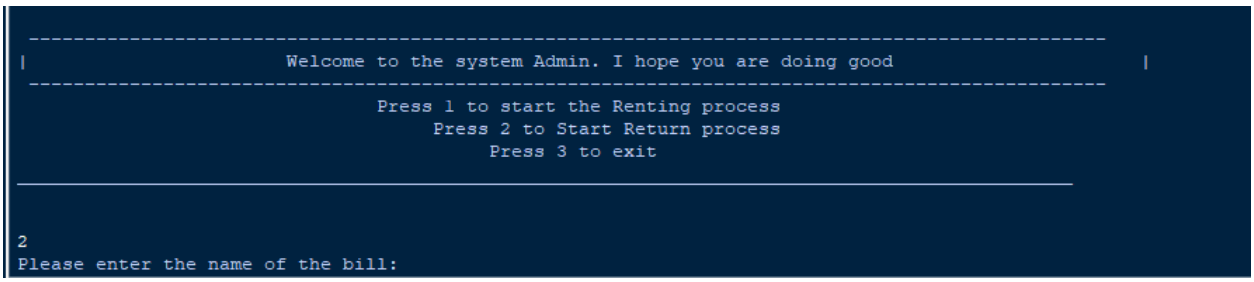
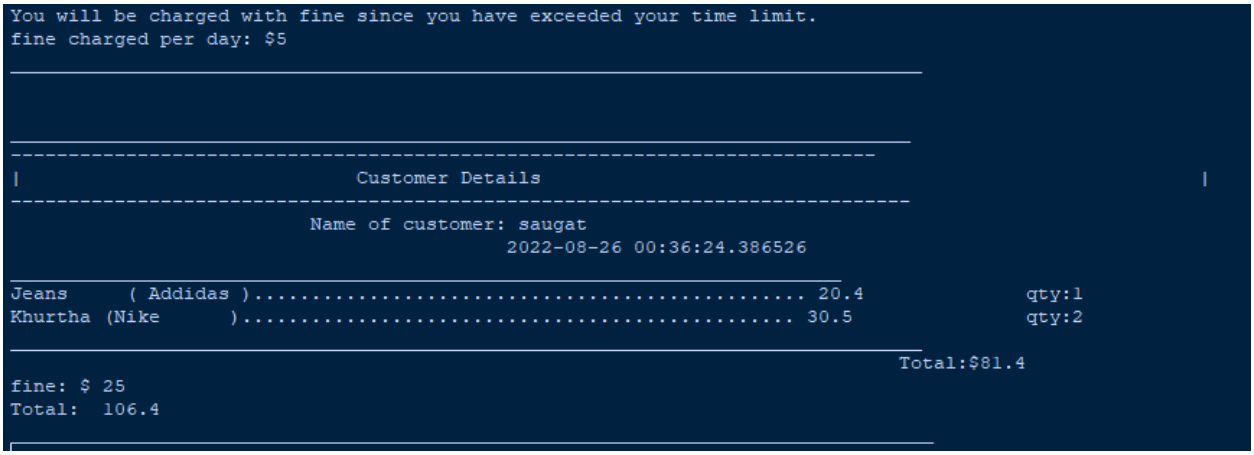


Figure 17 Returning

The program will ask the user to insert the name of the customer on which the bill was formed, and if the rented days have exceeded 5 days fine will be added to the total amount.

Figure 18 Display bill



Exit from the Program

The program will continue to run until and unless the exit option is selected.

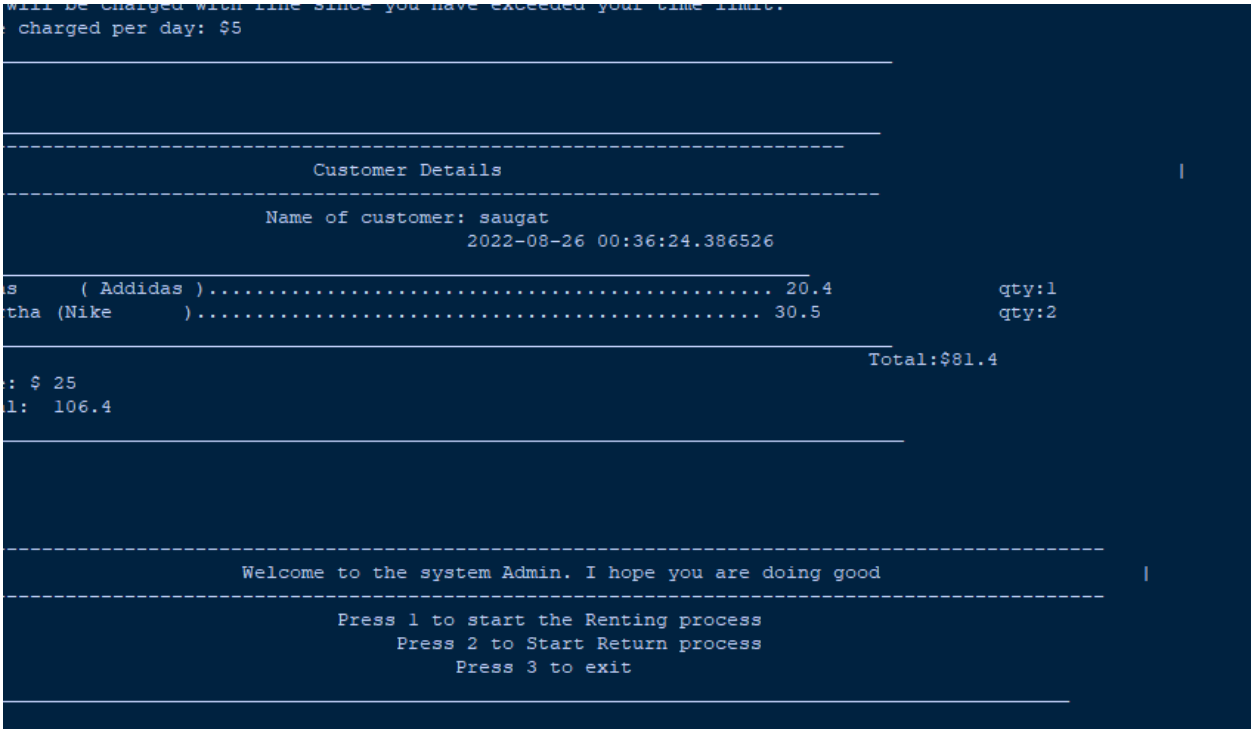
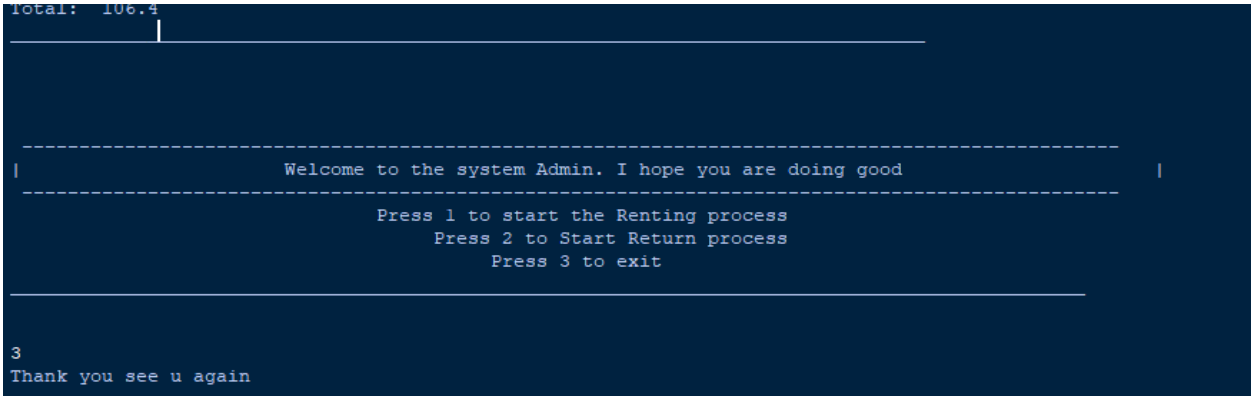


Figure 19 Exit from the program



The program is finally closed.

4. Testing

4.1 Test 1

Table 1 Test 1

Objective	To show implement try and except
Action	Insert invalid input
Expected Result	Message showing select again should be displayed
Actual Result	Message showing select again or invalid input was display
Conclusion	Test Successful

```
--------------------------------------------------------------------------------
ID              Costom Name             Brand           Rent Price
--------------------------------------------------------------------------------
1               Jeans                    Addidas                $20.40

2               Khurtha                 Nike                   $30.50

3               Track                   Gucci                  $12.50

4               Jacket                  Essential              $30.20

5               Sari                    Kolkata                $21.30

6               pant                    Levi                   $40.50

7               T-Shirt                 Ktm                    $10.40

8               Shirt                   polo                   $22.30
--------------------------------------------------------------------------------
Please select a cloth according to their id: r
Please select according to the option
Please select a cloth according to their id:
```

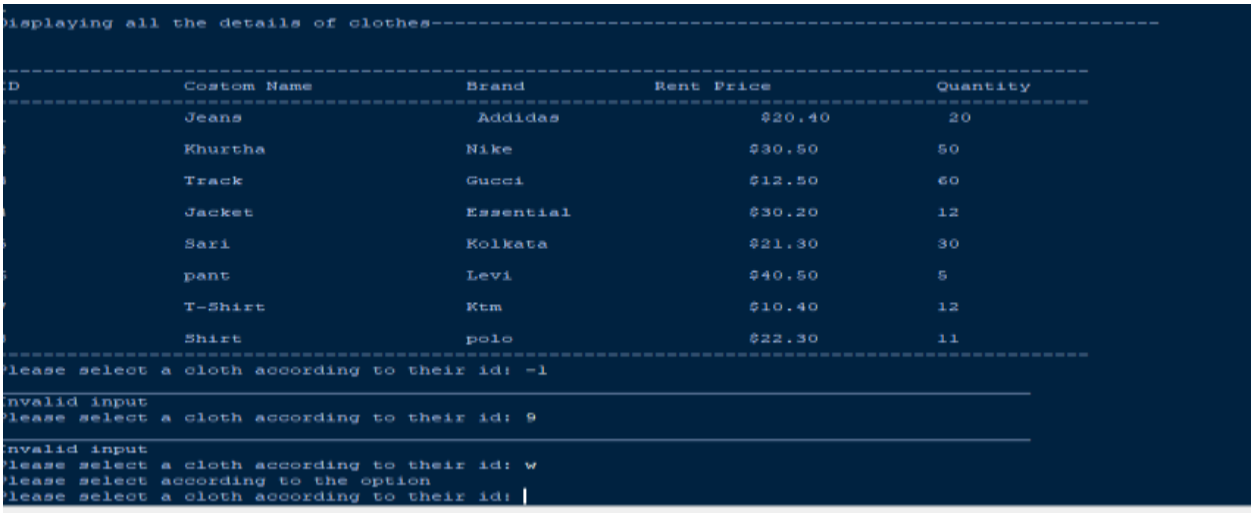
```
Please select a cloth according to their id: r
Please select according to the option
Please select a cloth according to their id: 1
_____________________________________________________________________________
Enter the quantity of cloth u wanna rent: 1
_____________________________________________________________________________
{1: ['Jeans     ', ' Addidas ', ' $20.40', '19\n'], 2: ['Khurtha ', 'Nike      ',
et     ', 'Essential ', '$30.20', '12\n'], 5: ['Sari        ', 'Kolkata  ', '$21
', 'Ktm     ', '$10.40', '12\n'], 8: ['Shirt       ', 'polo    ', '$22.30', '11'
Press Y if this in your final selection or N for adding  another item: 1
Invalid input
Please select a cloth according to their id:
```

Figure 20 Test 1

4.2 Test 2

Table 2 Test 2

Objective	Check what happens when invalid input is given as custom id
Action	Invalid input is given to the program
Expected Result	Message showing Invalid input should be displayed
Actual Result	Message showing invalid input was displayed
Conclusion	Test successful

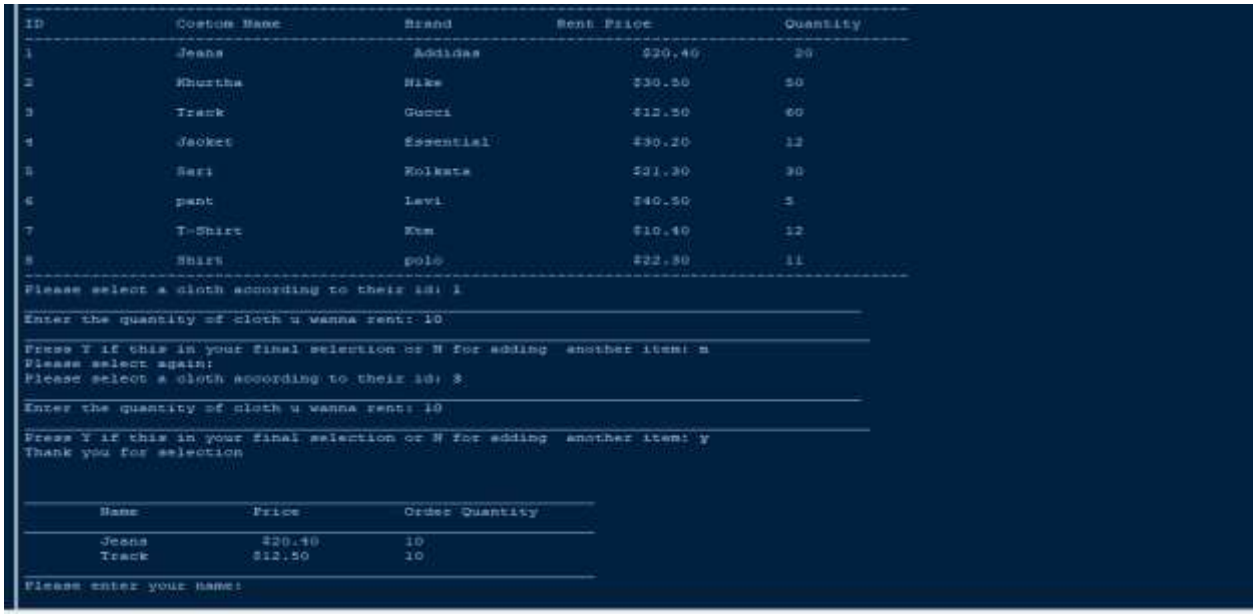


4.3Test 3

Table 3 Test 3

Objective	To show complete renting process
Action	Show complete renting process
Expected Result	Successful renting process
Actual Result	Renting process was successful
Conclusion	Test Successful

Figure 21 Test 2



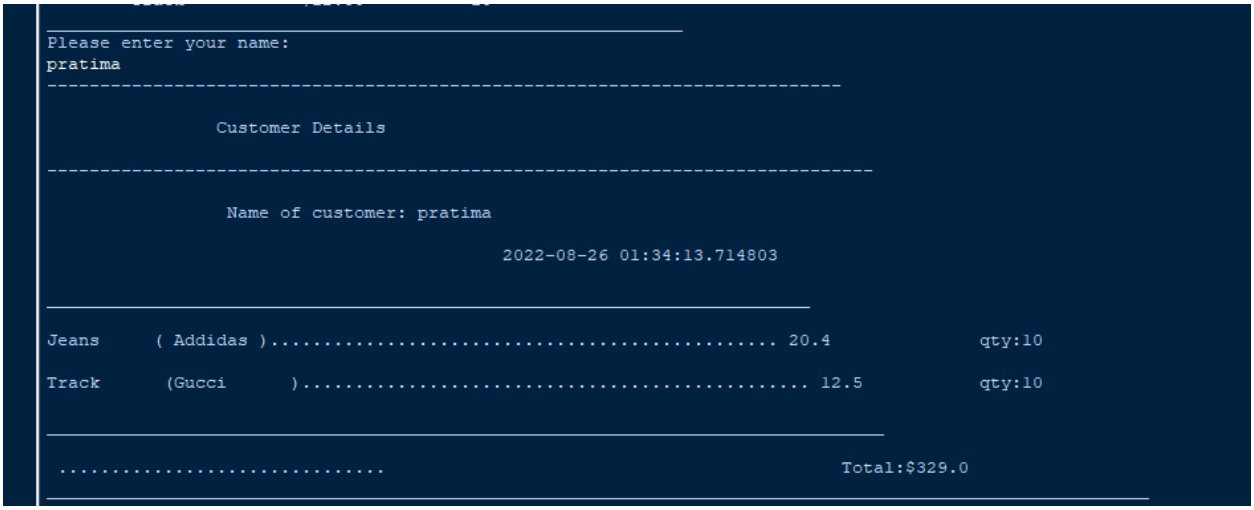


Figure 22 Test 3

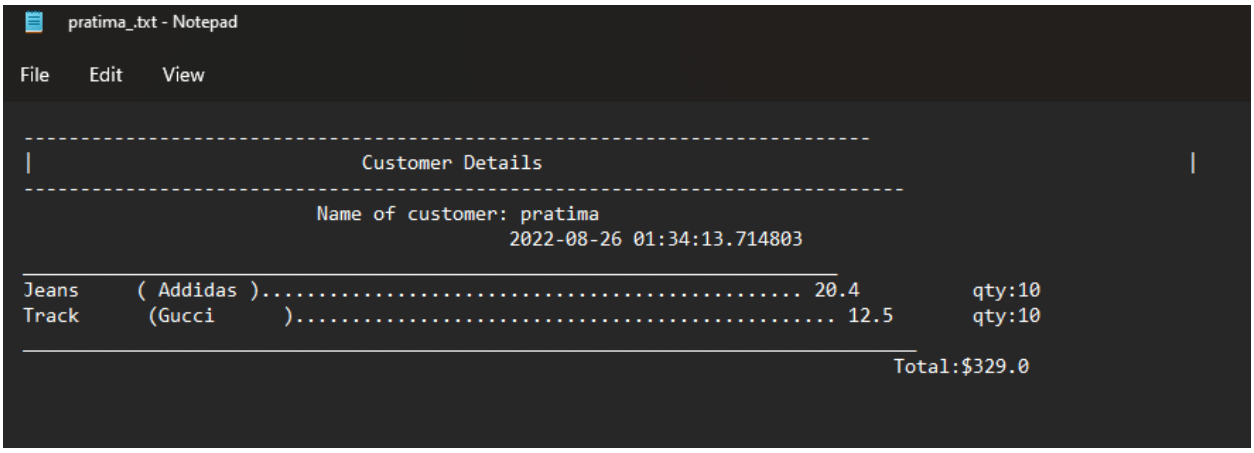
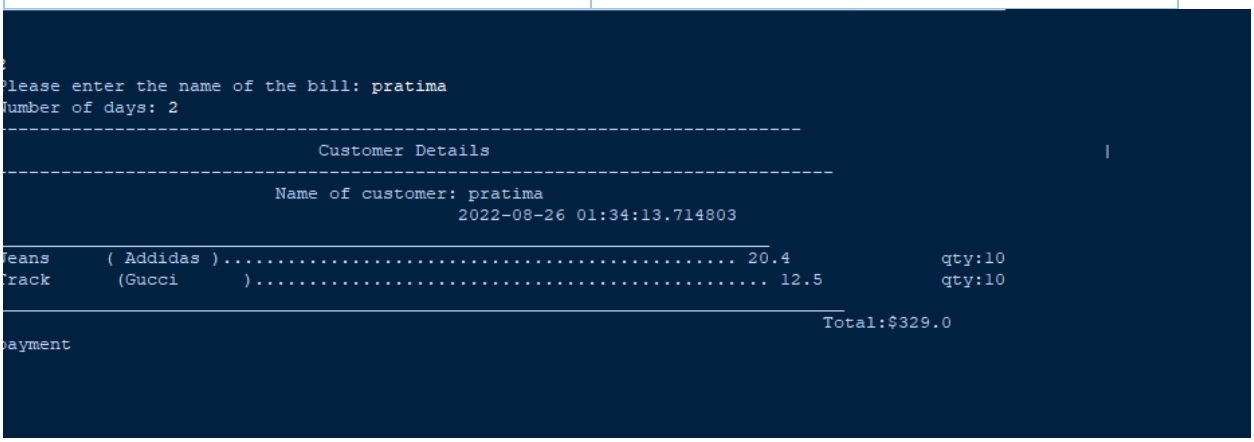


Figure 23 Test 3 bill

4.4Test 4

Table 4 Test 4

Objective	Show returning process
Action	Show complete returning process
Expected Result	Successful returning process
Actual Result	The returning process was successfully completed
Conclusion	Test Successful



```
lease enter the name of the bill: pratima
umber of days: 10

ou will be charged with fine since you have exceeded your time limit.
ine charged per day: $5


                          Customer Details

                     Name of customer: pratima
                                  2022-08-26 01:34:13.714803

eans     ( Addidas ).......................................... 20.4          qty:10
rack      (Gucci      ).......................................... 12.5          qty:10

                                                                    Total:$329.0
ine: $ 25
otal:  354.0
```

Figure 24 Test 4

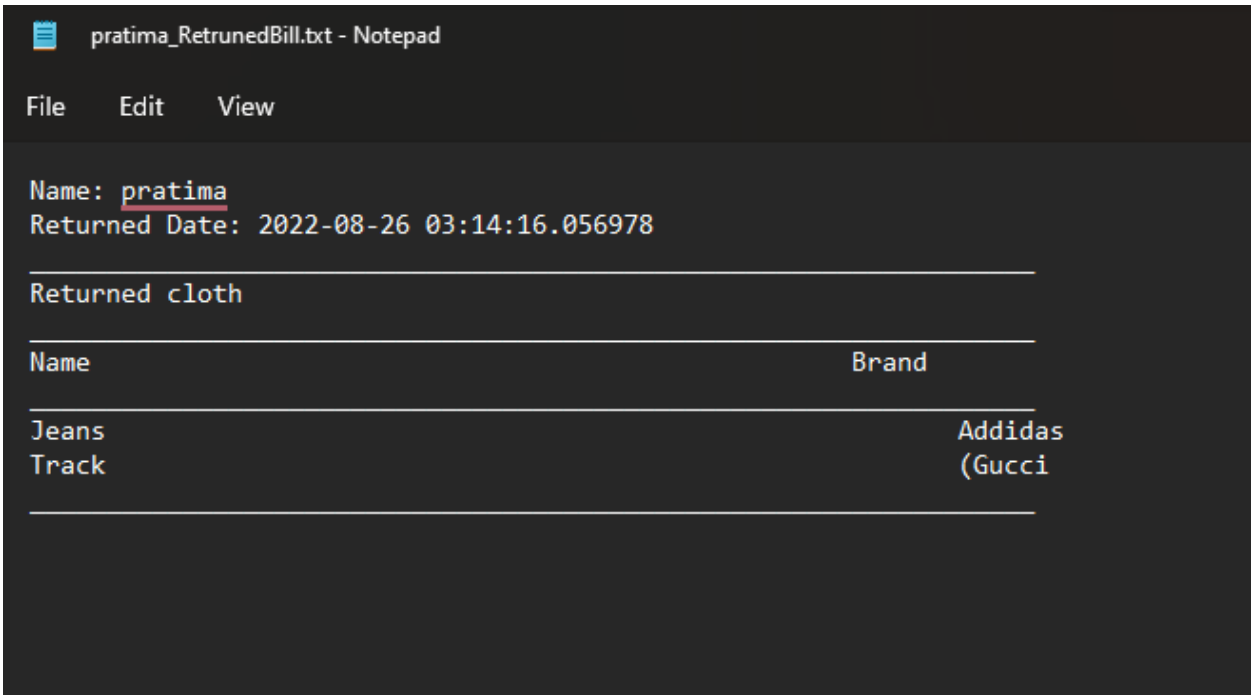


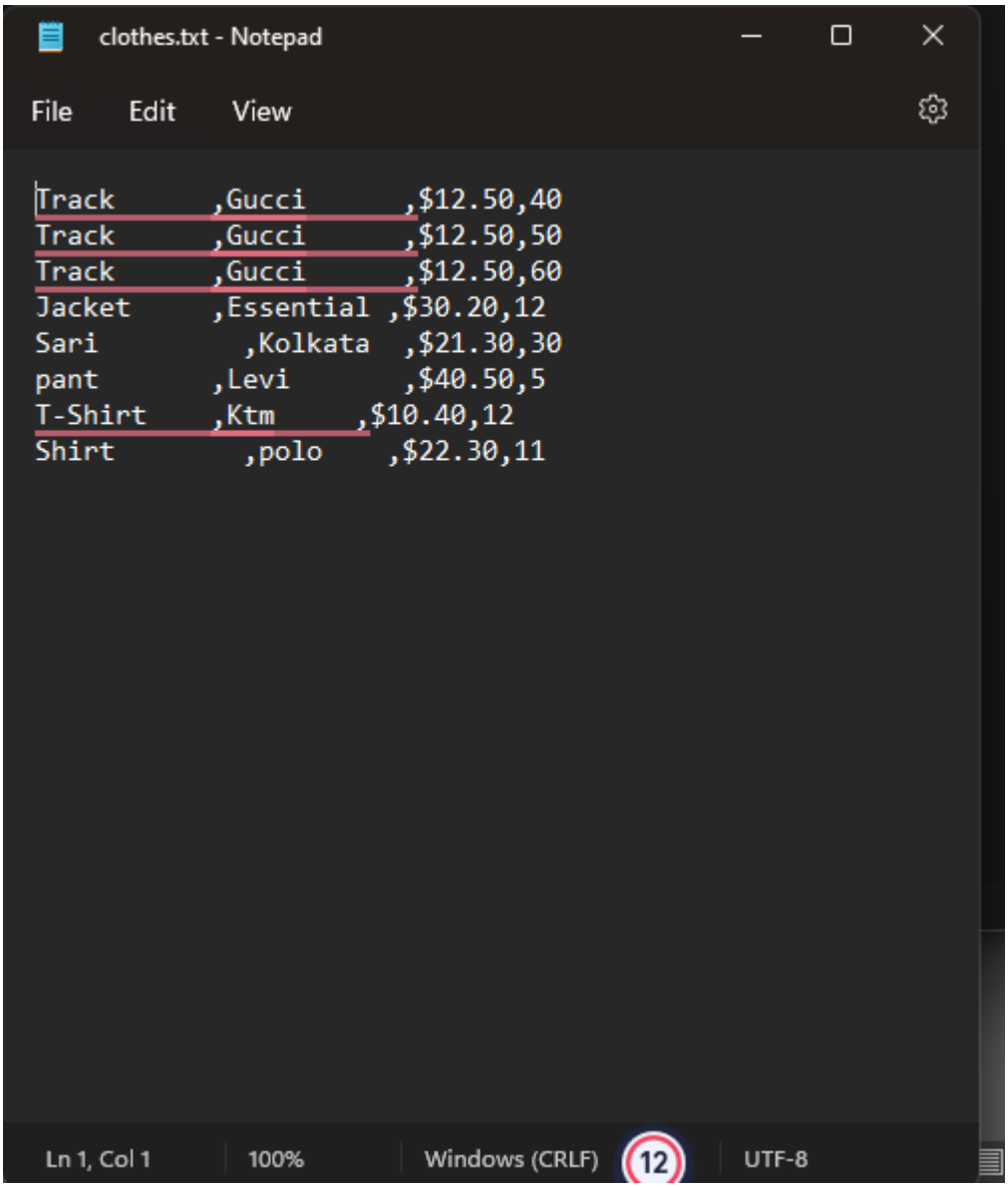
Figure 25 Test 4 bill

4.5 Test 5

Table 5 Test 5

Objective	Display the details of clothes text file after rented and returned
Action	Rent the cloth and returned the clothes
Expected Result	There should be Change in text file of details of clothes
Actual Result	The cloth details was changed
Conclusion	Test successful

```
1
Displaying all the details of clothes-----------------------------------------------------------------


-------------------------------------------------------------------------------------------------
ID              Costom Name             Brand           Rent Price              Quantity
-------------------------------------------------------------------------------------------------
1               Khurtha                 Nike                    $30.50          49

2               Track                   Gucci                   $12.50          50

3               Track                   Gucci                   $12.50          60

4               Jacket                  Essential               $30.20          12

5               Sari                    Kolkata                 $21.30          30

6               pant                    Levi                    $40.50          5

7               T-Shirt                 Ktm                     $10.40          12

8               Shirt                   polo                    $22.30          11

9

10

-------------------------------------------------------------------------------------------------
Please select a cloth according to their id: 1
_____________________________________________________________________________________
Enter the quantity of cloth u wanna rent: 10
_____________________________________________________________________________________
Press Y if this in your final selection or N for adding  another item:
```


clothes.txt - Notepad
File
Edit
View
Track ,Gucci ,$12.50,40
Track ,Gucci ,$12.50,50
Track ,Gucci ,$12.50,60
Jacket ,Essential ,$30.20,12
Sari ,Kolkata ,$21.30,30
pant ,Levi ,$40.50,5
T-Shirt ,Ktm ,$10.40,12
Shirt ,polo ,$22.30,11
Ln 1, Col 1
100%
Windows (CRLF)
12
UTF-8

5. Conclusion

The assignment was completed on time before the deadline of the submission with the help of teachers and my fellow mates. This was a very long and challenging assignment that taught me a lot about python and its functionality. This assignment helped me to solve real-time problems of programming and think wisely while doing programming.

This assignment was about developing back-end software for cloth renting from a store that gives the clothes on rent to the customers. After the renting process is complete the program prints out the bill according to the ordered clothes. The bill contains the detail of the user and details of the clothes rented. When the customer wants to return the rented costume, the bill having the details of the customer opens and displays the total. If the rented days have exceeded 5 days fine would be added to the total price.

Since the whole program was written in the python language this help me to learn many things about python and solve real-time problems which may occur in the real world.

6. References

Eduba. (n.d.). Retrieved from https://www.educba.com/python-pseudocode/
edureka. (n.d.). Retrieved from https://www.edureka.co/blog/data-structures-in-python/
Java T point. (n.d.). Retrieved from https://www.javatpoint.com/python-tutorial
JAVA TPOINT. (n.d.). *JavatPOINT*. Retrieved from https://www.javatpoint.com/java-tutorial

Appendix

```
import datetime



#Function for details of the custom rental shop

def shop_details():

print("_____
_____")

    print("                              Dharan Costom Rental
\n ")

    print("                               Dharan , Bhanu Chowk
\n")

print("_____
_____")
```

```
#Function to welcome the customer and give option to select to the customer

def welcome_():


    print(" ----------------------------------------------------------------------------------------------")

    print("|                    Welcome to the system Admin. I hope you are doing good
|")

    print(" ----------------------------------------------------------------------------------------------")


    print("                          Press 1 to start the Renting process")#If 1 is pressed rental
process will start

    print("                              Press 2 to Start Return process")#If 2 is pressed
returning process will start

    print("                                  Press 3 to exit")#To exit from the program


print("_____
_____")

    print("\n")


shop_details()# Calling thr function to display th shop details

welcome_()#Calling the function to display the welcome program


#To find the total of the clothing iteam

def price_total(a,b,dic):

    sum = 0
```

```
    for (i,j) in zip(a,b):
        dic[i][2] = float(dic[i][2].replace("$",""))
        sum=sum+dic[i][2]*j
    return sum


# When Customer clicks the option 1 the cloth selection process will start
def cloth_select():
    #Displaying the details of the cloths
    print("-------------------------------------------------------------------------------------------")
    print("ID\t\tCostom Name\t\tBrand\t\tRent Price\t\tQuantity    ")
    print("-------------------------------------------------------------------------------------------")
    f = open("clothes.txt","r")
    a =1# Declaring the key to the cloths item

    for line in f:
        print(a,"\t\t" + line.replace(",","\t\t"))
        a=a+1

    print("-------------------------------------------------------------------------------------------")
```

```
#A function to develop th bill after the renting process in completed
def customer_details(list_,list_qty):
    print("Please enter your name: ")
    user_name = input("").lower()#Using lower to make name of the file in lower case
    f = open("clothes.txt","r")
    #Creating dictionary using txt file
    dic ={}
    id = 1# Declaring th key value for the dictionary
    for line in f :
        line = line.replace("/n" ,"")
        dic.update({id:line.split(",")})
        id = id +1
    f.close()
    print
    date  = datetime.datetime.now()#Importing the functions of th python for present time
    total_price = price_total(list_,list_qty,dic)#Calling the function which calculate the total


   #Printing the customer bill in the terminal
    print("-------------------------------------------------------------------------\n")
    print("\t\tCustomer Details   \n")
    print("-------------------------------------------------------------------------\n")
    print(f"\t\t Name of customer: {user_name}   \n")
    print(f"                                {date}                                \n")
```

```
print(f"_____
_____\n")

    for(i,j)  in zip(list_,list_qty):

      print(f"{dic[i][0]}({dic[i][1]})..... {dic[i][2]}\t \tqty:{j}\n")


print("_____
_____\n")

    print(f" .....                                    Total:${total_price}  ")



    #Writing the same bill in the txt file using open fuction of python

    bill = open(f"customer/{user_name}_.txt",'w')#Giving the name to the file as name of
the customer

    bill.write("---------------------------------------------------------------------\n")

    bill.write("|                        Customer Details
|\n")

    bill.write("---------------------------------------------------------------------\n")

    bill.write(f"                    Name of customer: {user_name}   \n")

    bill.write(f"                                {date}                          \n")


bill.write(f"_____
_____\n")

    for(i,j)  in zip(list_,list_qty):

       bill.write(f"{dic[i][0]}({dic[i][1]})..... {dic[i][2]}\t \tqty:{j}\n")


bill.write("_____
_____\n")
```

```
    bill.write(f"                                                    Total:${total_price}  ")



#Function to select the cloth according to the option of the cloths
def cloth_selection(dic):
    list_ = []#Declaring a list
    list_qty =[]#Declaring a list to add the qty of the cloths needed by the customer
    selection =True
    while selection ==True:
      #Using try catch exception if wrong value is inserted
      try:
       user_input_Id = int(input("Please select a cloth according to their id: "))#Select the cloth according to the given number
      except ValueError:
          print("Please select according to the option")#insert the needed qty of the cloths
          continue


print("_____
_____")
```

```
        if 0 < user_input_Id <= len(dic):#If the key value of the cloth is unaailable
          try:
            cloth_qty = int(input("Enter the quantity of cloth u wanna rent: "))
          except ValueError:
            print("Qauntity should be in number")
            continue

print("_____
_____")
          list_qty.append(cloth_qty)
          d = int(dic[user_input_Id][3]) - cloth_qty
          dic[user_input_Id][3] = f"{d}\n"


          #Giving a choice to the customer if they want  to add other cloth or not
          user_input2 = input("Press Y if this in your final selection or N for adding  another
item: ").lower()
          list_.append(user_input_Id)


          if user_input2 == "y":#if yes the billing process will start
             print("Thank you for selection")
             print("\n")
             #Displaying ht list of selected cloths

print("_____")
             print("\tName\t\tPrice\t\tOrder Quantity")
```

```
print("_____")
            for(i,j)  in zip(list_,list_qty):
               print(f"\t{dic[i][0]}\t{dic[i][2]}\t\t{j}")

print("_____")
            customer_details(list_,list_qty)


            selection = False
         elif user_input2 =="n":# If no the loop will start again for adding new list of the clothes
            print("Please select again: ")
          else:
            print("Invalid input")
        else:
            print("Invalid input")



#Imporiting txt file and chaning it into a dictionary

def dictionary_():# Extracting details of the cloths from the text file and adding it to the dictionary as values for keys
    f = open("clothes.txt","r")
    dic ={}
    id = 1
    for line in f :
```

```
        line = line.replace("/n" ,"")
        dic.update({id:line.split(",")})
        id = id +1
    f.close()
    cloth_selection(dic)


    #If the customer rents for more than 5 days , fine will be charged to the customer
def charge_fine(no_days,user_name1):
     with open(f"customer/{user_name1}_.txt","r") as f:   #Opening the txt file for the customer
        table = f.readlines()
        a  = (table[-1]).split(":")
        a[1] = a[1].replace("$","")
        fine = (no_days-5)*5 #Charging the fine to customer for 5 dollar per day
        print("fine: $",fine)
        print("Total: ", float(a[1])+ fine) # Displaying the total to the customer as per the bill wiht the fine


def returned_bill(user_name1):
     cloth=[]
     brand=[]
     with open(f"customer/{user_name1}_.txt","r") as f:
        table = f.readlines()
        for i in range(6,len(table)-2):
```

```
        a = table[i].split(" ")
        cloth.append(a[0])
        brand.append(a[6])
    return_bill = open(f"Return/{user_name1}_RetrunedBill.txt",'w')
    date = datetime.datetime.now()
    return_bill.write(f"Name: {user_name1}\n")
    return_bill.write(f"Returned Date: {date}\n")

return_bill.write("_____
_____\n")
    return_bill.write("Returned cloth\n")

return_bill.write("_____
_____\n")
    return_bill.write("Name                                    Brand\n")

return_bill.write("_____
_____\n")
    for i,j in zip(cloth,brand):
        return_bill.write(f"{i}                                    {j}\n")

return_bill.write("_____
_____\n")



#Return process
def return_():
```

```
    user_name1 = input("Please enter the name of the bill: ").lower()

    no_days = int(input("Number of days: "))#Asking the customer for how many days they have rented the custom

    if no_days>5: #Using the for loop if the customer has exceeded the time limit

print("_____")

        print("You will be charged with fine since you have exceeded your time limit.")

        print('fine charged per day: $5')

print("_____")

        print("\n")

print("_____")

        with open(f"customer/{user_name1}_.txt","r") as f:

            print(f.read())


        charge_fine(no_days,user_name1)

print("_____")

        #If the customer is returned in time no fine shall be charged

    else:

        with open(f"customer/{user_name1}_.txt","r") as f:

            print(f.read())
```

```
            print("payment")
    returned_bill(user_name1)
    print("\n")
    print("\n")
    welcome_()


#Giving the customer the option for renting or returning of the custom
def option_():
    loop = True
    #Using the loop so the loop will continously run until the customer exits
    while loop == True:
        try:
         user_input = int(input(""))
        except ValueError:
            print("**************************************")
            print("Select according to th options.")
            print("**************************************")
            continue
         #If the user pressed 1 the renting process will start
        if user_input == 1:
            print("Displaying all the details of clothes----------------------------------------------------
---------\n\n")
            cloth_select()
```

```
            dictionary_()

print("_____
_____")
            print("\n")
            welcome_()
        #If the customer presses 2 returning process will start


        elif user_input == 2:
            return_()
        elif user_input == 3:
            print("Thank you see u again")
            loop = False


        else:
            print("Invalid input")
            welcome_()


#Calling the fuction option to start the program
option_()
```

Originality report

COURSE NAME
CS4051NI - Fundamentals of Computing

STUDENT NAME
Saugat Basnet Networking

FILE NAME
22015870 Saugat Basnet (AutoRecovered)

REPORT CREATED
26 Aug 2022

Summary

Flagged passages	3	2%
Cited/quoted passages	6	4%

Web matches

edureka.co	5	3%
educba.com	2	1%
dev.to	1	0.6%
coursehero.com	1	0.4%

1 of 9 passages
Student passage FLAGGED

Python pseudocode is more like an algorithmic representation of the code involved. This means when a code is expected to be formulated it cannot be directly drafted

Top web match

Introduction to Python pseudocode **Python pseudocode is more like an algorithmic representation of the code involved. This means when a code is expected to be formulated it cannot be directly drafted.**
Complete Guide to Python pseudocode - eduCBA https://www.educba.com/python-pseudocode/