

**NE BANKING SYSTEM**

**MR. KELVIN NG HAN YAO**

**TP-063580**

**LECTURER: MR. TANVEER KHALEEL SHAIKH**

Table of Contents

[1 INTRODUCTION 3](#_Toc75185358)

[2 ASSUMPTION 3](#_Toc75185359)

[3 DESIGN - PSEUDOCODE 4](#_Toc75185360)

[4 DESIGN - FLOWCHART 9](#_Toc75185361)

[5 SAMPLE PYTHON CODE 13](#_Toc75185362)

[5.1 Import file 13](#_Toc75185363)

[5.2 Variable 13](#_Toc75185364)

[5.3 Control Structure 13](#_Toc75185365)

[5.3.1 If 13](#_Toc75185366)

[5.3.2 If-else 13](#_Toc75185367)

[5.3.3 Nested if-else 14](#_Toc75185368)

[5.4 Looping Structure 14](#_Toc75185369)

[5.4.1 While loop 14](#_Toc75185370)

[5.4.2 For loop 14](#_Toc75185371)

[5.5 List Variable 14](#_Toc75185372)

[5.6 Function 15](#_Toc75185373)

[5.7 File 15](#_Toc75185374)

[5.7.1 Write data in text file 15](#_Toc75185375)

[5.7.2 Read data in text file 15](#_Toc75185376)

[5.7.3 Search or modify data of text file 16](#_Toc75185377)

[6 SCREENSHOT OF INPUT / OUTPUT 17](#_Toc75185378)

[7 CONCLUSION 23](#_Toc75185379)

[8 REFERENCES 24](#_Toc75185380)

# INTRODUCTION

Keeping track of numerous customer’s transactions can be quite a difficult task in every banking sector; therefore, a good online banking system is important to allow user to effectively manage all transactions of customers easily.

The main purpose of the system is to simulate customer’s transaction management for bank to maintain the transaction records. The system should also provide two login portal each for admin and customer, respectively.

Admin portal in the system should be able to create new customer’s profile and provide them login id and password to access the system, view, and search for specific customer’s profile and able to view all transactions of specific customer.

Customer portal in the system should be able to login to the system, conduct transactions action such as deposit and withdrawal and view their own past transactions.

# ASSUMPTION

1. I assume system can register customer.
2. I assume that with the help of this system user can login with username and password.
3. I assume both admin and customer portal can view transaction details.
4. I assume admin portal can search for specific customer’s profile.
5. I assume that customer can perform deposit and withdrawal action.
6. I assume that customer can view their own transaction details.
7. I assume that user can click “q” to terminate the system.

# DESIGN - PSEUDOCODE

Function view

Pass In: username, category(deposit/withdrawal)

Direct the operating system to read respective text file and sum up the records

Pass Out: sum of the value

End function

Function record

Pass In: username, category(deposit/withdrawal)

Read amt (value to withdraw/deposit)

Direct the operating system to read respective text file and sum up the record

Pass Out: sum of the value, amt input by user

End function

START

DECLARE amt, sum, role, start, login\_counter, wrong\_counter, username, password. afile, cfile, dfile, wfile, end, task, service, create, new\_username ,new\_password, profile\_name, profile\_age, profile\_gender, profile\_contact, psearch, pfile, balance, deposit, withdrawal, withdrawal\_balance, deposit\_balance, new\_balance

DISPLAY ("----------------------------------"

"\nWelcome to Online Banking System."

"\nPlease select your role."

"\n----------------------------------")

role = “”

DISPLAY("\n ----------------------------"

"\n| Enter 1 for Admin role. |"

"\n| Enter 2 for Customer role. |"

"\n| Enter q to quit. |"

"\n ----------------------------\n")

INPUT role

start = 0

WHILE start != 1

login\_counter = 0

wrong\_counter = 0

IF role == "1" or role == "2"

DISPLAY ("\nPlease enter your username: ")

READ username

DISPLAY ("Please enter your password: ")

READ password

user\_pass = username+","+password

IF len(username)==0 or len(password)==0

DISPLAY ("You must enter both username and password.")

ELSE

IF role == "1"

READ afile

FOR admin in afile

IF user\_pass == admin

DISPLAY ("Login Successful")

login\_counter = 1

start = 1

ENDIF

ENDFOR

IF login\_counter != 1

READ afile

FOR admin in afile

IF admin.startswith(username+",")

DISPLAY ("Wrong Password. Please login again.\n")

wrong\_counter = 1

ENDIF

ENDFOR

IF login\_counter != 1 and wrong\_counter!=1

DISPLAY ("Account not exist. Please contact bank for account creation.\n")

ENDIF

ENDIF

ENDIF

IF role == "2"

READ cfile

FOR customer in cfile

IF user\_pass == customer

DISPLAY ("Login Successful")

login\_counter = 2

start = 1

ENDIF

ENDFOR

IF login\_counter != 2

READ cfile

FOR customer in cfile

IF customer.startswith(username+",")

DISPLAY ("Wrong Password. Please login again.\n")

wrong\_counter = 2

ENDIF

ENDFOR

IF login\_counter != 2 and wrong\_counter!=2

DISPLAY ("Account not exist. Please contact bank for account creation.\n")

ENDIF

ENDIF

ENDIF

ENDIF

ENDIF

ENDWHILE

end = “”

WHILE end != "q"

task = ""

service = ""

create = 0

IF role == "1" and login\_counter == 1

DISPLAY("\n --------------------------------------------------------"

"\n| Enter 1 to create new customer's profile. |"

"\n| Enter 2 to view and search customer's profile. |"

"\n| Enter 3 to view all transactions of specific customer. |"

"\n| Enter q to quit. |"

"\n --------------------------------------------------------\n")

INPUT task

ENDIF

IF role == "2" and login\_counter == 2

DISPLAY ("\n -------------------------------"

"\n| Enter 1 to Deposit. |"

"\n| Enter 2 to Withdrawal. |"

"\n| Enter 3 to view transactions. |"

"\n| Enter q to quit. |"

"\n -------------------------------\n")

INPUT service

ENDIF

IF role == "q" or task == "q" or service == "q"

end = "q"

ENDIF

IF task == "1"

DISPLAY ("\nPlease enter a new username: ")

INPUT new\_username

DISPLAY ("Please enter a new password: ")

INPUT new\_password

FOR customer in cfile

IF customer.startswith(new\_username+",")

DISPLAY ("Account already exist.")

create = 1

ENDIF

ENDFOR

IF create != 1

DISPLAY ("\nPlease enter the profile correctly."

"\nYou CANT change it after the account is created")

DISPLAY ("\nEnter customer's name: ")

INPUT profile\_name

DISPLAY ("Enter customer's age: ")

INPUT profile\_age

DISPLAY ("Enter customer's gender: ")

INPUT profile\_gender

DISPLAY ("Enter customer's phone number: ")

INPUT profile\_contact

DISPLAY ("\nProfile created.")

INPUT (new\_username+" profile.txt")

ENDIF

ENDIF

IF task == "2"

READ ufile

DISPLAY ("\nUsername List"

"\n-------------")

FOR username in ufile

DISPLAY username

ENDFOR

DISPLAY ("\nEnter username to view customer's profile: ")

INPUT pfile

FOR profile in pfile

DISPLAY profile

ENDFOR

ENDIF

IF task == "3"

READ ufile

DISPLAY ("\nUsername List"

"\n-------------")

FOR username in ufile

DISPLAY username

ENDFOR

DISPLAY ("\nEnter username to view customer's transaction: ")

INPUT psearch

READ tfile

DISPLAY ("\nCustomer's Transaction"

"\n----------------------")

FOR transaction in tfile

IF (transaction.startswith(psearch+",")

DISPLAY transaction

ENDIF

ENDFOR

ENDIF

IF service == "1"

call: view

balance = view(username,"deposit") - view(username,"withdrawal")

DISPLAY ("Your account balance is",balance)

call: record

deposit , deposit\_balance = record(username,"deposit")

new\_balance = deposit\_balance - view(username,"withdrawal")

READ ("transaction list.txt")

DISPLAY ("You have deposit",deposit,"and your new balance

is",new\_balance)

ENDIF

IF service == "2"

call: view

balance = view(username,"deposit") - view(username,"withdrawal")

IF balance > 0

DISPLAY ("Your account balance is "+str(balance)+"."

"\nYou can withdraw "+str(balance)+" from your account.") call: record

withdrawal , withdrawal\_balance = record(username,"withdrawal")

IF withdrawal <= balance

new\_balance = view(username,"deposit") - withdrawal\_balance READ ("transaction list.txt")

DISPLAY ("You have withdraw",withdrawal,"and your new balance is",new\_balance)

ELSE

DISPLAY ("\nYour account balance is not sufficient."

"\nYou can only withdraw "+str(balance)+" from your account.") ENDIF

ELSE

DISPLAY ("Your account balance is not sufficient.")

ENDIF

IF service == "3"

READ tfile

DISPLAY ("\nMy Transactions"

"\n---------------")

FOR transaction in tfile

IF (transaction.startswith(username+",")

DISPLAY transaction

ENDIF

ENDFOR

ENDIF

ENDWHILE

END

# DESIGN - FLOWCHART

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

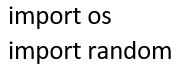
Description automatically generated

Diagram

Description automatically generated

# SAMPLE PYTHON CODE

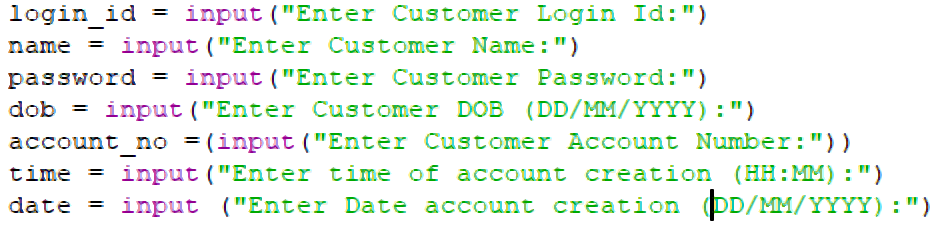
## Import file



Import os imports os module which is a part of the standard library within Python. Even though it comes with Python installation, but it still needs to be imported.

Import random imports random module which relate with random number generation such as random () function that can generates random numbers between 0 and 1.

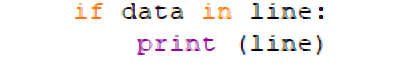
## Variable



Variable is a named placeholder where programmer can store data and later retrieve the data using the variable name. Programmers can name the variables and change the contents of a variable in a later statement.

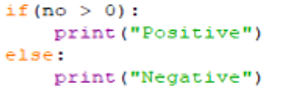
## Control Structure

### If



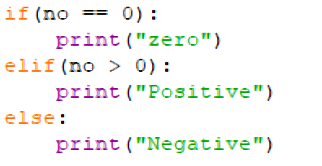
If statement is one-way decision-making operations. It runs only when the if statement is true. If the condition is false, since there is no else statement, it will proceed with next codes.

### If-else



If-else statement is two-way decision-making operations. It runs only when the if statement is true. If the condition is false, then the else statement runs before proceeding to next codes.

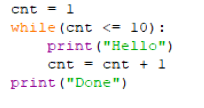
### Nested if-else



Nested if-else statement is multi-way decision-making operations. A nested if-else is an if-else statement embed inside another if-else statement. It is possible to nested multiple if-else statement inside an if-else statement.

## Looping Structure

### While loop



While loop support a set of statements is to be repeated for several times. While loops are controlled by an initialized loop variables and usually loop variables are initialized right before the loop itself.

### For loop



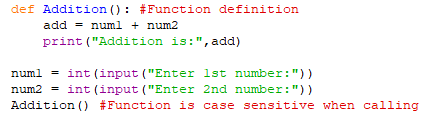
For loop repeats the of statement for each element by sequence. For loop end when it reaches the end of the sequence.

## List Variable



List group similar items in a square bracket. Lists are heterogeneous which means the elements in a list need not be of the same type and can contain both integers and strings.

## Function



Function is a piece of code used to deal with a specific task. Function break programs into manageable small chunks of code. Huge programs are difficult to manage it is better to split them with the help of functions.

## File

### Write data in text file



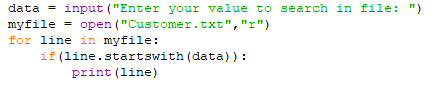
Before writing data into text file, system need to open the text file with the name provided by user. “w” in the open file function will create a new text file if the text file with name provided by user is not exist. After writing data into the text file, user need to use the close file function to close the text file so the data will be saved in the text file.

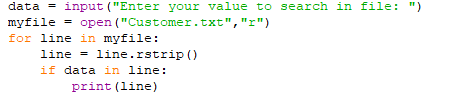
### Read data in text file



Before reading data from text file, system need to open the text file with the name provided by user. “r” in the open file function will only allow system to read the text file with name provided by user. After reading data from the text file and display it, user need to use the close file function to close the text file.

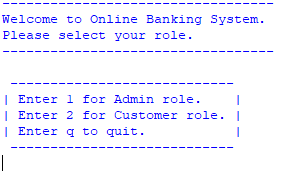
### Search or modify data of text file



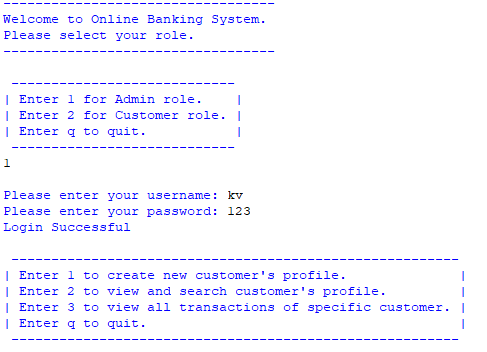


Before searching or modify data from text file, system need to open the text file with the name provided by user. “r” in the open file function will only allow system to read the text file with name provided by user. After reading data from the text file, a for loop is use to search and modify each and every line in the text file.

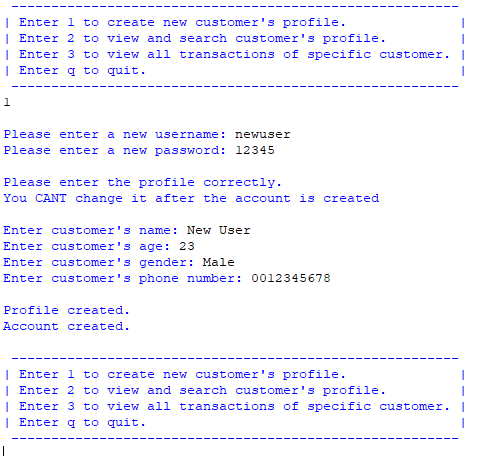
# SCREENSHOT OF INPUT / OUTPUT



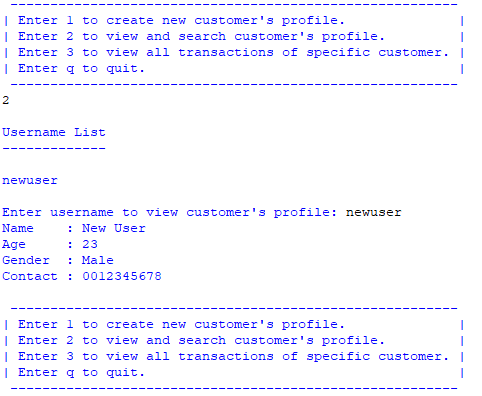
As soon as user execute the system, it will lead user to the starting interface and require user to input their selection on role.



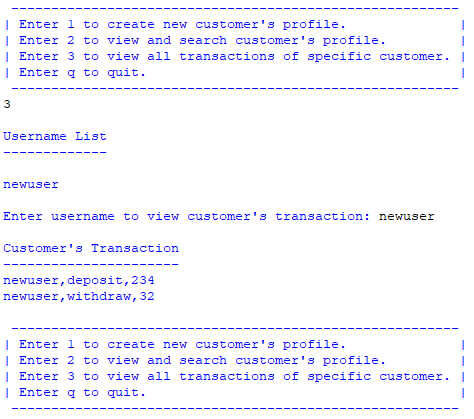
Once user select 1 for admin role, it will prompt user to enter their username and password. If the username and password is correct, the system will show the task for admin role to be selected by user.



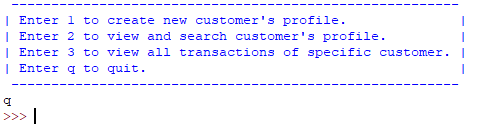
If user selected 1 to create new customer’s profile, the system will prompt use to enter new username and new password. Next, system will request user to enter the customer’s detail correctly. Once all the details are entered and the username is first time creation, system will stored the newly created profile and account details in the text file and back to task selection page.



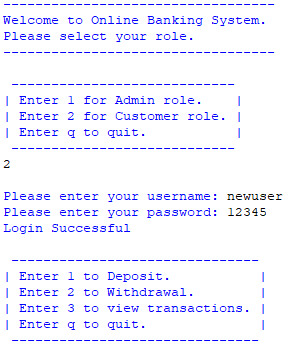
If user select 2 to view and search customer’s profile, system will show all existing username in text file and user may enter username that want to be searched and the profile details of the selected username will be shown. Once the details is shown, system will back to task selection page.



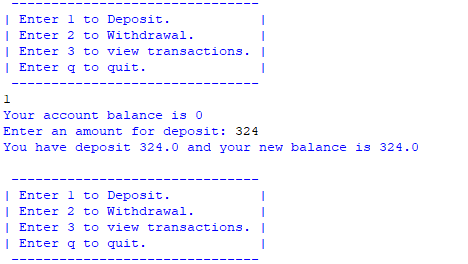
If user select 3 to view all transactions of specific customer, system will show all existing username to user and once user enter the specific username, system will show all the transaction details of the selected username and back to the task selection page.



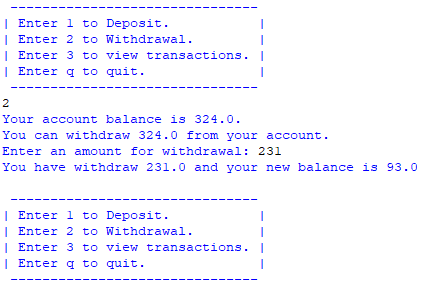
If user select q, system will terminate.



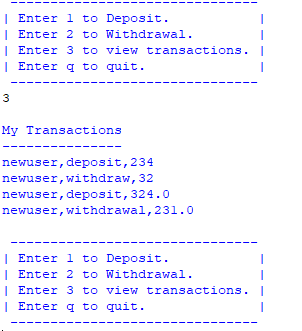
Once user select 2 for customer role, it will prompt user to enter their username and password. If the username and password is correct, the system will show the services for customer role to be selected by user.



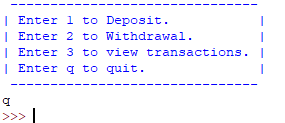
If user select 1 to deposit, system will show the account balance of user and prompt user to enter an amount for deposit. Once a deposit amount is entered, the value will be stored in text file and system will display the amount stored by user and the new account balance. Then, system will back to services selection page.



If user select 2 to withdrawal, system will show the account balance of user and prompt user to enter an amount for withdrawal. Once a withdrawal amount is entered, system will check whether there’s sufficient account balance to withdraw. If the account balance is sufficient, the withdrawal value will be stored in text file and system will display the amount withdraw by user and the new account balance. Then, system will back to services selection page.



If user select 3 to view all transactions, system will search for same username and display all the transaction details of user and back to the task selection page.



If user select q, system will terminate.

# CONCLUSION

1. I conclude that system can register customer successfully.
2. I conclude that with the help of this system user can login with username and password.
3. I conclude that both admin and customer portal can view transaction details.
4. I conclude that admin portal can search for specific customer’s profile.
5. I conclude that customer can perform deposit and withdrawal action successfully.
6. I conclude that customer can view all their own transaction details.
7. I conclude that user can click “q” to terminate the system successfully.

# REFERENCES

Abuse, S., 2020. *How to Check if List is Empty in Python.* [Online]   
Available at: https://stackabuse.com/how-to-check-if-list-is-empty-in-python  
[Accessed 20 June 2021].

Kalkman, P., 2020. *Python Error Handling.* [Online]   
Available at: https://blog.devgenius.io/python-error-handling-8bed3f5b5769  
[Accessed 13 June 2021].

Macharia, N., 2018. *How to write Pseudocode: A beginner’s guide.* [Online]   
Available at: https://blog.usejournal.com/how-to-write-pseudocode-a-beginners-guide-29956242698  
[Accessed 12 June 2021].

naina024, 2019. *Python exit commands.* [Online]   
Available at: https://www.geeksforgeeks.org/python-exit-commands-quit-exit-sys-exit-and-os-\_exit/  
[Accessed 16 June 2021].

poker158149, 2012. *Using Try, Except to catch a blank input.* [Online]   
Available at: https://www.daniweb.com/programming/software-development/threads/381248/using-try-except-to-catch-a-blank-input  
[Accessed 17 June 2021].

Pool, P., 2021. *7 Quick Ways to Check If String is Empty in Python.* [Online]   
Available at: https://www.pythonpool.com/empty-string-python/  
[Accessed 19 June 2021].

Programiz, 2021. *Python Functions.* [Online]   
Available at: https://www.programiz.com/python-programming/function  
[Accessed 8 June 2021].

Project, O. B., n.d. *Dictionaries, sets, files, and modules.* [Online]   
Available at: https://www.openbookproject.net/books/bpp4awd/ch06.html  
[Accessed 6 June 2021].

SmritiSatyan, 2019. *How to check for an empty string in Python?.* [Online]   
Available at: https://www.studytonight.com/post/how-to-check-for-an-empty-string  
[Accessed 15 June 2021].

Sturtz, J., n.d. *Defining Your Own Python Function.* [Online]   
Available at: https://realpython.com/defining-your-own-python-function/  
[Accessed 14 June 2021].

w3schools, n.d. *Python Lists.* [Online]   
Available at: https://www.w3schools.com/python/python\_lists.asp  
[Accessed 16 June 2021].