CERTIFICATE

This is to certify that Samar Kumar and Prakash Kumar of class XI -SCIENCE have successfully completed this Computer project on the topic Bank Management System.

Making prescribe by Mrs Deepika Aggarwal during academic session 2021-22, as per the Guidelines by Central Board of Secondary Education.

# ACKNOWLEGDEMENT

We are extremely grateful to Mrs. Deepika Aggarwal, teacher of department of computer science, for her able guidance and useful suggestion, which helped us in completing the project work in time.

Finally, we would like to express my heartfelt thanks to our respective parents for their blessing and our friends for their help and wishes for the successful completion of project.

# INDEX

1. Introduction
2. Working Description
3. Source code
4. Output
5. Limitations
6. Conclusion
7. Bibliography

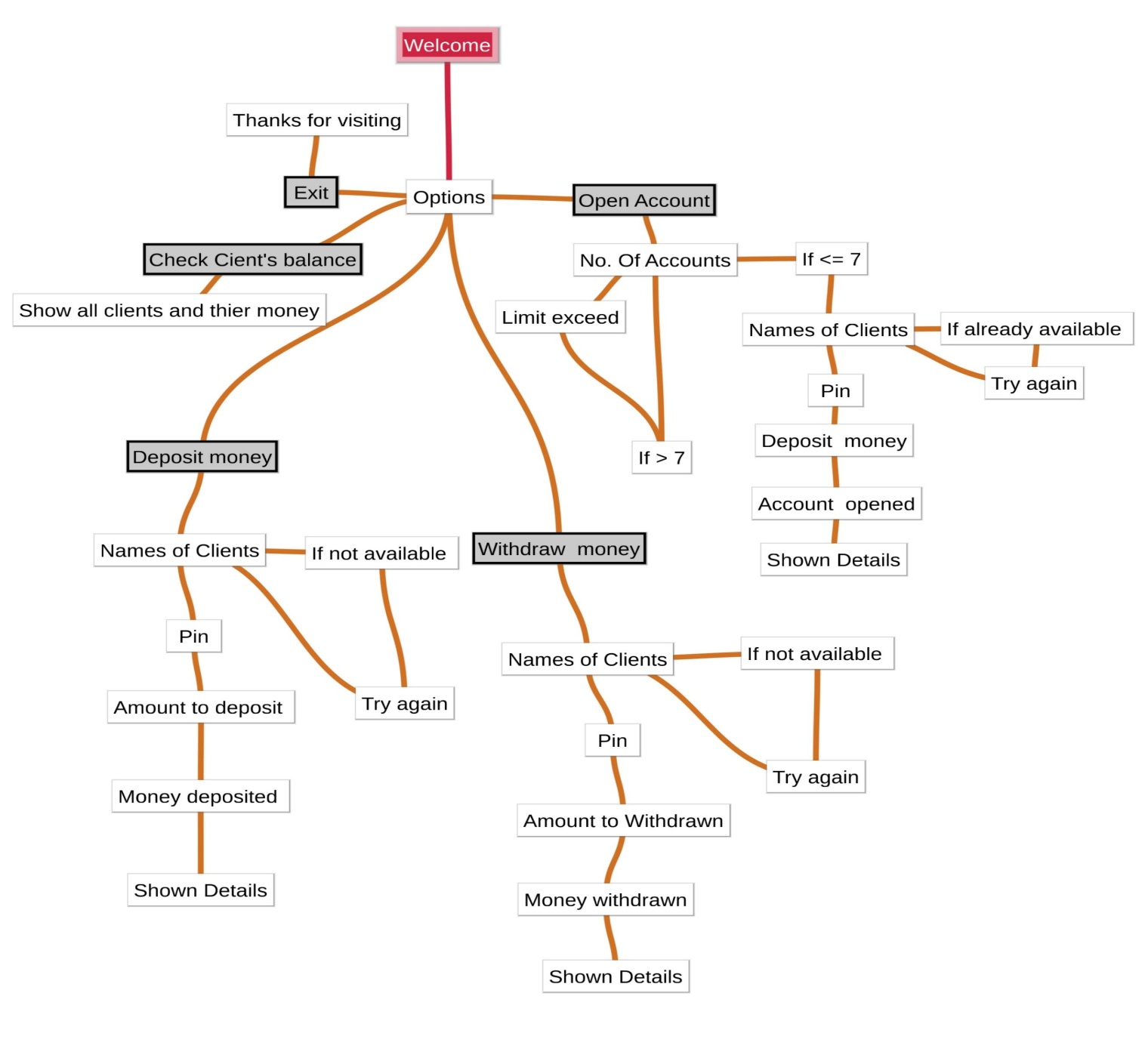
INTRODUCTION

Our project is a “The Bank Management System”, which helps us and world economy to function.

Talking about the machine, it includes all of the fundamental features required in a bank. There is no login system as this is a mini project.

Besides, this means the person can use all those available capabilities without difficulty without any restrictions. It’s too simple to use, the person can look at the facts of total financial institution account without problems.

WORKING DESCRIPTION



SOURCE CODE

NamesOFClients = ['Steve Jobs', 'Bill Gates', 'Elon Musk', 'Samar', 'Prakash', 'Rajnesh Chandra Jain', 'Feralick']

ClientPins = ['0001', '0002', '0003', '0004', '0005', '0006', '0007']

ClientBalances = [1000000000, 120000000000, 321000000000, 4000000000, 1000050000, 600000000, 7000000000]

ClientDeposition = 0

ClientWithdrawal = 0

ClientBalance = 0

disk1 = 1

disk2 = 7

u = 0

while True:

# os.system("cls")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("========== WELCOME TO THE WORLD'S RICHEST BANK ==========")

print("========== FERALICK BANKING SYSTEM ==========")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("========== (a). Open New Client Account ============")

print("========== (b). The Client Withdraw a Money ============")

print("========== (c). The Client Deposit a Money ============")

print("========== (d). Check Clients' Balance ============")

print("========== (e). Quit ============")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

EnterLetter = input("Select a Letter from the Above Box menu : ")

if EnterLetter == "a":

print(" Letter a is Selected by the Client")

NumberOfClient = eval(input("Number of Clients : "))

u = u + NumberOfClient

if u > 7:

print("\n")

print("Client registration exceed reached or Client registration too low")

u = u - NumberOfClient

else:

while disk1 <= u:

name = input("Write Your Fullname : ")

while ( name in NamesOFClients):

print("This name is already available! try another..\n")

name = input("Type different name : ")

NamesOFClients.append(name)

pin = str(input("Please Write a Pin to Secure your Account : "))

ClientPins.append(pin)

ClientBalance = 0

ClientDeposition = eval(input("Please Insert a Money to Deposit to Start an Account : "))

ClientBalance = ClientBalance + ClientDeposition

ClientBalances.append(ClientBalance)

print("\nName=", end=" ")

print(NamesOFClients[disk2])

print("Pin=", end=" ")

print(ClientPins[disk2])

print("Balance=", "P", end=" ")

print(ClientBalances[disk2], end=" ")

disk1 = disk1 + 1

disk2 = disk2 + 1

print("\nYour name is added to Client Table")

print("Your pin is added to Client Table")

print("Your balance is added to Client Table")

print("----New Client account created successfully !----")

print("\n")

print("Your Name is Available on the Client list now : ")

print(NamesOFClients)

print("\n")

print("Note! Please remember the Name and Pin")

print("========================================")

mainMenu = input("Press Enter Key to go Back to Main Menu to Conduct Another Transaction or Quit\_")

elif EnterLetter == "b":

v = 0

print(" letter b is Selected by the Client")

while v < 1:

w = -1

name = input("Please Insert a name : ")

while ( name not in NamesOFClients):

print("This name is not available! try another..\n")

name = input("Type different name : ")

pin = input("Please Insert a pin : ")

while w < len(NamesOFClients) - 1:

w = w + 1

if name == NamesOFClients[w]:

if pin == ClientPins[w]:

v = v + 1

print("Your Current Balance:", "P", end=" ")

print(ClientBalances[w], end=" ")

print("\n")

ClientBalance = (ClientBalances[w])

ClientWithdrawal = eval(input("Insert value to Withdraw : "))

if ClientWithdrawal > ClientBalance:

deposition = eval(input(

"Please Deposit a higher Value because your Balance mentioned above is not enough : "))

ClientBalance = ClientBalance + deposition

print("Your Current Balance:", "P", end=" ")

print(ClientBalance, end=" ")

ClientBalance = ClientBalance - ClientWithdrawal

print("-\n")

print("----Withdraw Successfully!----")

ClientBalances[w] = ClientBalance

print("Your New Balance: ", "P", ClientBalance, end=" ")

print("\n\n")

else:

ClientBalance = ClientBalance - ClientWithdrawal

print("\n")

print("----Withdraw Successfully!----")

ClientBalances[w] = ClientBalance

print("Your New Balance: ", "P", ClientBalance, end=" ")

print("\n")

if v < 1:

print("Your name and pin does not match!\n")

break

mainMenu = input("Press Enter Key to go Back to Main Menu to Conduct Another Transaction or Quit\_")

elif EnterLetter == "c":

print("Letter c is selected by the Client")

x = 0

while x < 1:

w = -1

name = input("Please Insert a name : ")

while ( name not in NamesOFClients):

print("This name is not available! try another..\n")

name = input("Type different name : ")

pin = input("Please Insert a pin : ")

while w < len(NamesOFClients) - 1:

w = w + 1

#if name != NamesOFClients[w]:

#print("Your name or pin does not match!\n")

if name == NamesOFClients[w]:

if pin == ClientPins[w]:

x = x + 1

print("Your Current Balance: ", "P", end=" ")

print(ClientBalances[w], end=" ")

ClientBalance = (ClientBalances[w])

print("\n")

ClientDeposition = eval(input("Enter the value you want to deposit : "))

ClientBalance = ClientBalance + ClientDeposition

ClientBalances[w] = ClientBalance

print("\n")

print("----Deposition successful!----")

print("Your New Balance: ", "P", ClientBalance, end=" ")

print("\n")

if x < 1:

print("Your name or pin does not match!\n")

break

# if name ==! NameOFClients[w]:

# print("Your name or pin does not match!\n")

mainMenu = input("Press Enter Key to go Back to Main Menu to Conduct Another Transaction or Quit\_")

elif EnterLetter == "d":

print("Letter d is selected by the Client")

w = 0

print("Client name list and balances mentioned below : ")

print("\n")

while w <= len(NamesOFClients) - 1:

print(w + 1, "Customer =", NamesOFClients[w])

print(w + 1, "Balance =", "P", ClientBalances[w], end=" ")

print("\n")

w = w + 1

mainMenu = input("Press Enter Key to go Back to Main Menu to Conduct Another Transaction or Quit\_ ")

elif EnterLetter == "e":

print("letter e is selected by the client")

print("Thank you for using our banking system!")

print("\n")

print("Thank You and Come again")

print("God Bless")

break

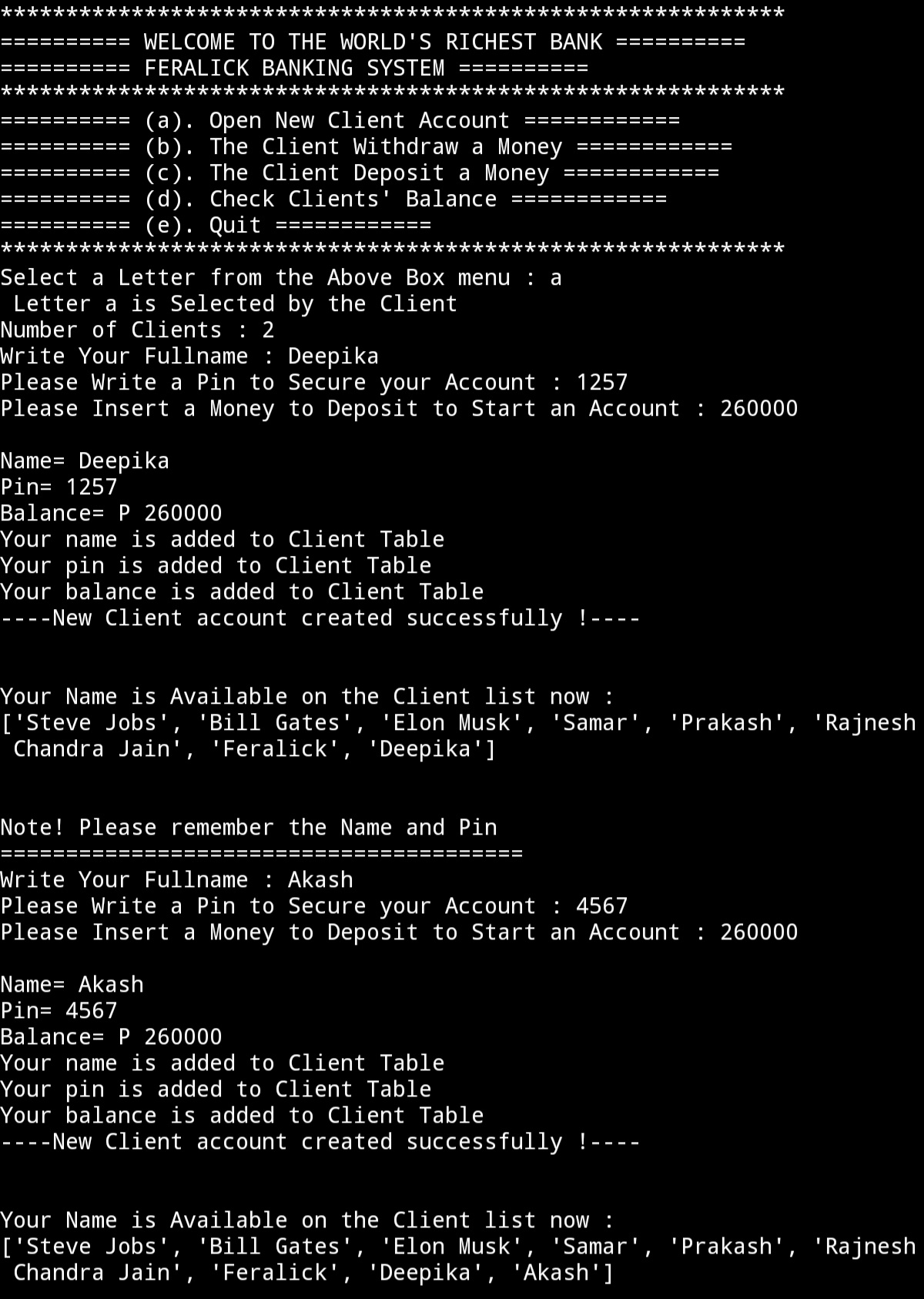
else:

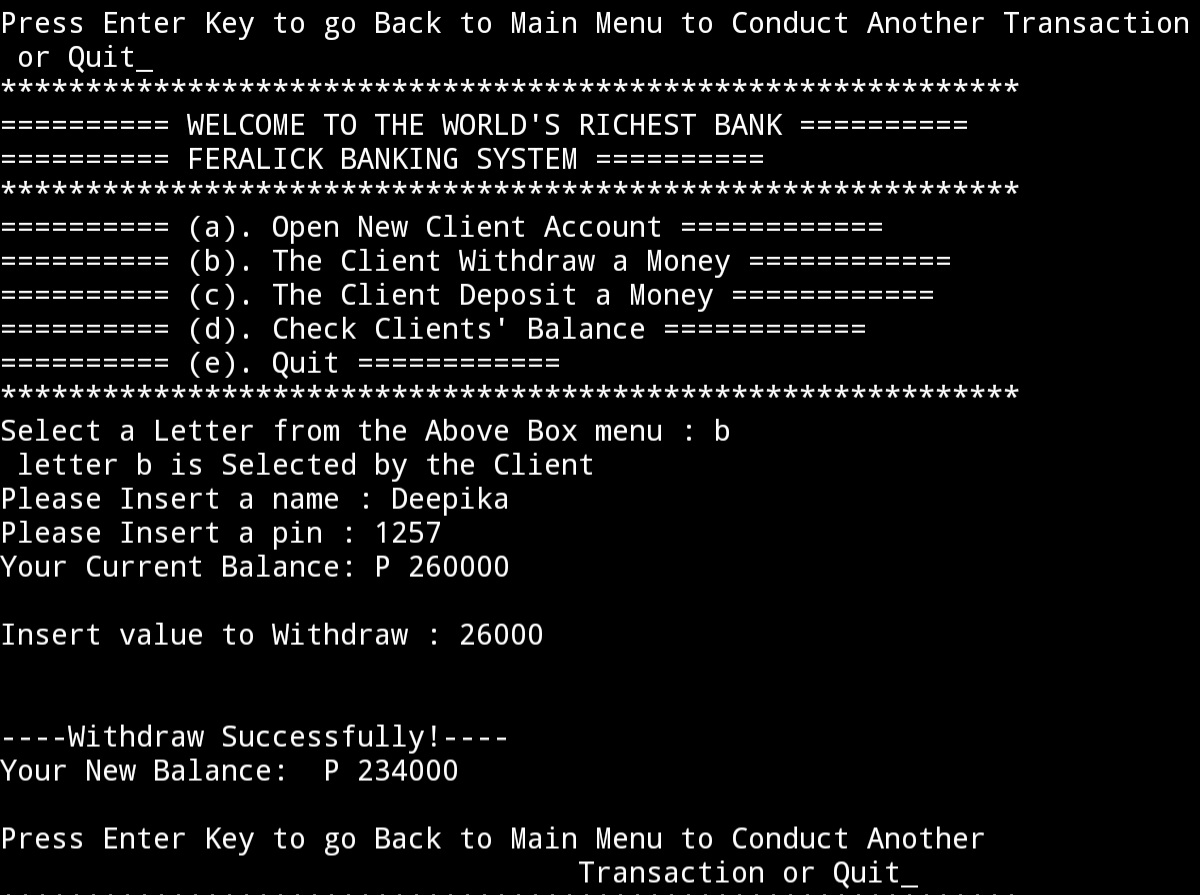
print("Invalid option selected by the Client")

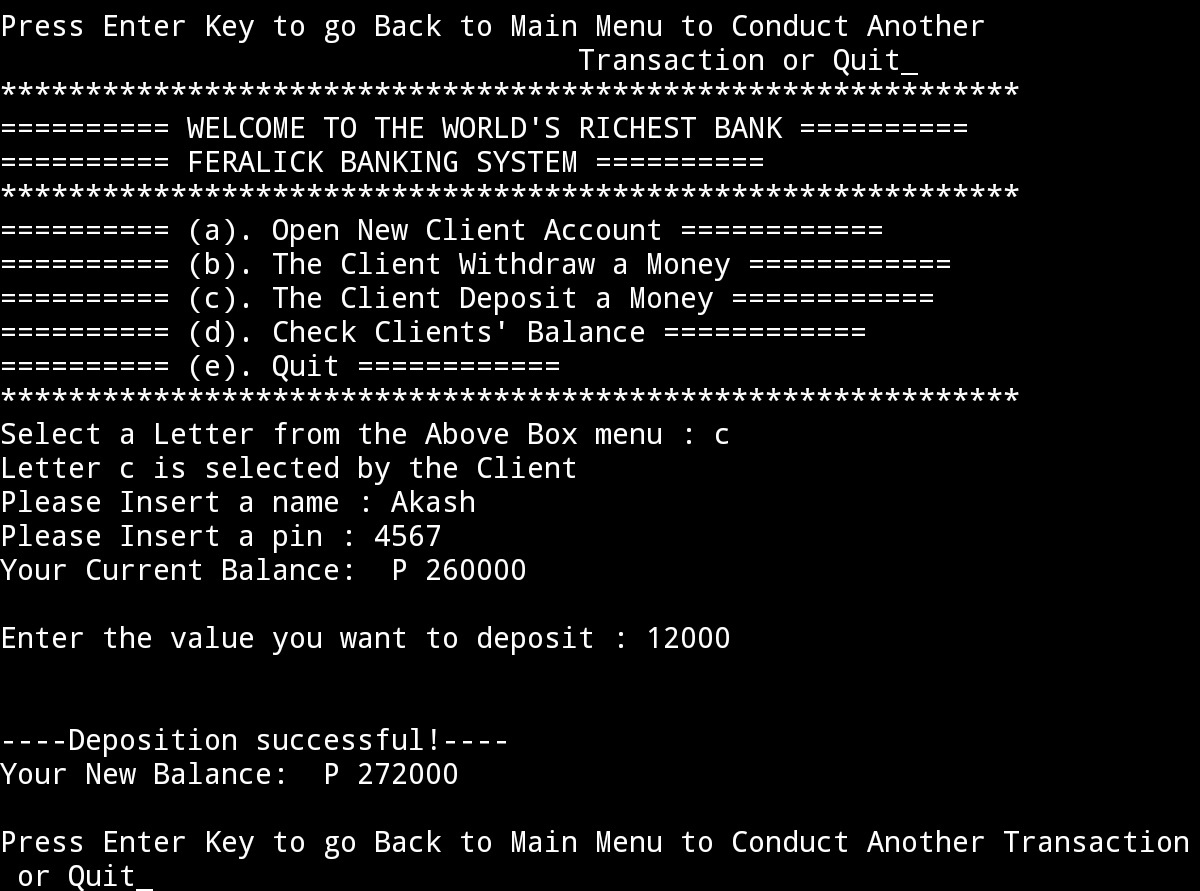
print("Please Try again!")

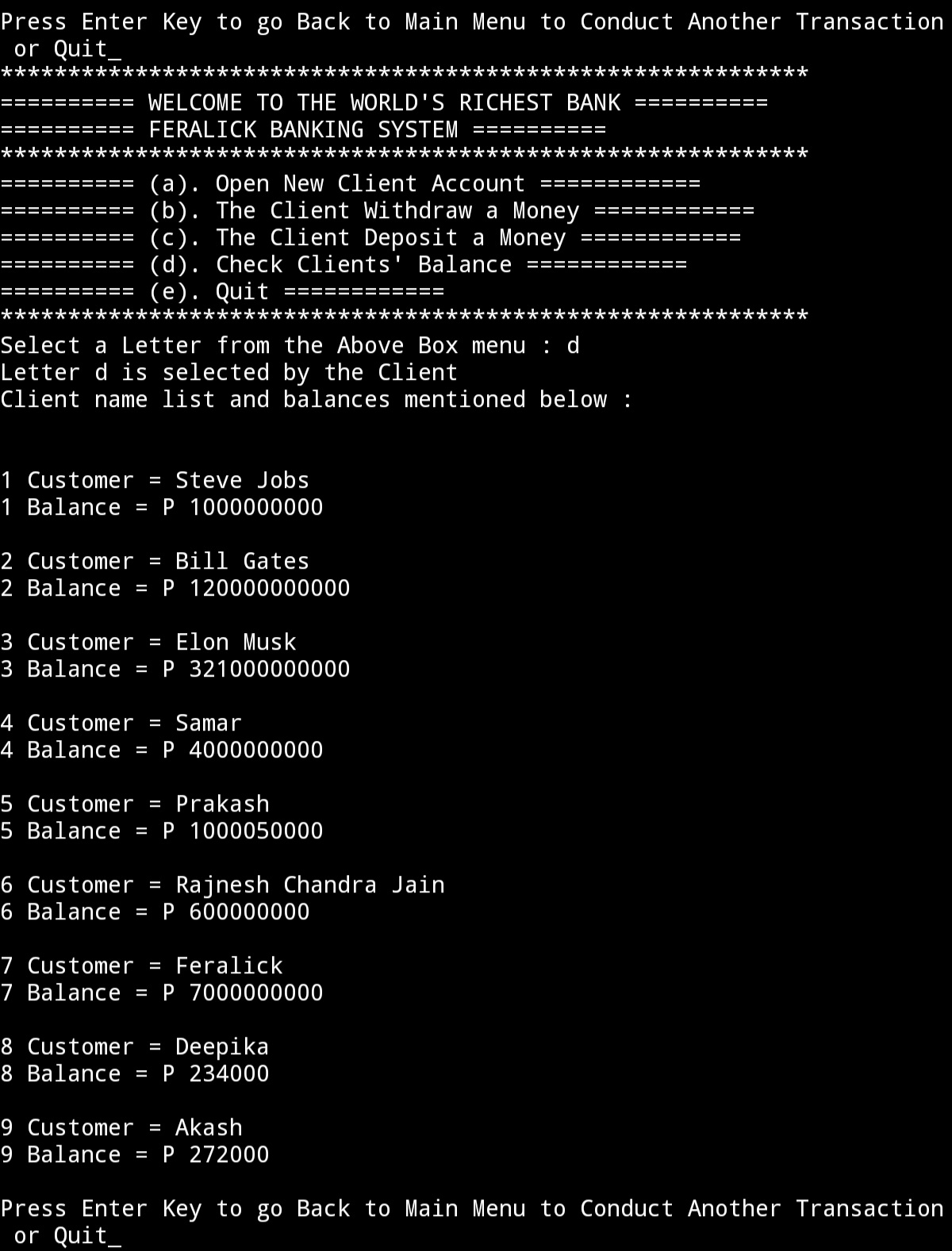
mainMenu = input("Press Enter Key to go Back to Main Menu to Conduct Another Transaction or Quit\_")

OUTPUT











LIMITATIONS

1. This program is offline
2. No involvement of real money
3. Account created lost as soon as we exit program

CONCLUSION

It was a wonderful and learning experience for me while working on this incredible project. While making this project "The World's Richest Bank : Feralick". I made progress by solving a number of problems. Solutions to each problem is done by our Computer Science teacher and it was the important part of project. This provided me experiences which will help me in future.

Some important things that I learned include designing of program architecture and converting real life situation into an efficient code and to write a good, easily readable and as well as time and memory efficient code.

I enjoyed each and every bit of work I had put into this project. This project is further extendable.

BIBLIOGRAPHY

1. <https://itsourcecode.com>

THANKS