Name: Obiora Ekene Donald

Track: Backend Development, cohort 2

Write a USSD banking program using python .Scope

1. Deposit

2. Transfer

3. Withdraw

4. Check Balance

Make use of functions and classes.

Answer

class account:

def \_\_init\_\_(self, balance=5000):

self.balance = balance

def deposit(self, amount):

self.balance += amount

return f"New balance: N{self.balance}"

def withdraw(self, amount):

if amount <= self.balance:

self.balance -= amount

return f"new balance: N{self.balance}"

return "Not enough money"

def transfer(self, amount, destination):

if amount <= self.balance:

self.balance -= amount

target\_account.deposit(amount) # type: ignore

return f"tranferred N{amount}. New balance: N{self.balance}"

return "Not enough money!"

def check\_balance(self):

return f"balance: N{self.balance}"

def ussd\_banking():

my\_account = account(5000)

friend\_account = account() #create second account to tranfer money to

while 'True':

print("1. Deposit\n2. Withdraw\n3. Transfer\n4. Check Balance\n5. Exit")

choice = input("Choose an option: ")

if choice == "1":

amount = float(input("Enter deposit amount: "))

print(my\_account.deposit(amount))

elif choice == "2":

amount = float(input("Enter withdrawal amount: "))

print(my\_account.withdraw(amount))

elif choice == "3":

account\_number = input("Enter recipient account number: ")

banks = ["Bank A", "Bank B", "Bank C"]

print("Select bank:")

for i, bank in enumerate(banks, 1):

print(f"{i}. {bank}")

bank\_choice = int(input("Choose a bank: "))

if 1 <= bank\_choice <= len(banks):

amount = float(input("Enter transfer amount: "))

if my\_account.transfer(amount, friend\_account):

print(f"Successfully transferred ${amount} to account {account\_number} at {banks[bank\_choice - 1]}.")

else:

print("Not enough money to complete the transfer.")

else:

print("Invalid bank choice.")

elif choice == "4":

print(my\_account.check\_balance())

elif choice == "5":

break

else:

print("Try again!")

ussd\_banking()