import sys

from datetime import datetime

class User:

def \_\_init\_\_(self, user\_id, pin, balance=0):

self.user\_id = user\_id

self.pin = pin

self.balance = balance

self.transactions = []

def check\_pin(self, pin):

return self.pin == pin

def add\_transaction(self, transaction):

self.transactions.append(transaction)

class ATM:

def \_\_init\_\_(self):

self.users = {}

def add\_user(self, user):

self.users[user.user\_id] = user

def get\_user(self, user\_id):

return self.users.get(user\_id, None)

class Transaction:

def \_\_init\_\_(self, amount, transaction\_type):

self.amount = amount

self.transaction\_type = transaction\_type

self.date = datetime.now().strftime('%Y-%m-%d %H:%M:%S')

class ATMSystem:

def \_\_init\_\_(self, atm):

self.atm = atm

self.current\_user = None

def authenticate\_user(self):

user\_id = input("Enter User ID: ")

pin = input("Enter PIN: ")

user = self.atm.get\_user(user\_id)

if user and user.check\_pin(pin):

self.current\_user = user

print("Authentication successful!")

else:

print("Invalid User ID or PIN.")

def show\_menu(self):

while True:

print("\n1. Transactions History\n2. Withdraw\n3. Deposit\n4. Transfer\n5. Quit")

choice = input("Choose an option: ")

if choice == '1':

self.show\_transactions()

elif choice == '2':

self.withdraw()

elif choice == '3':

self.deposit()

elif choice == '4':

self.transfer()

elif choice == '5':

print("Exiting... Thank you!")

sys.exit()

else:

print("Invalid choice. Please try again.")

def show\_transactions(self):

if self.current\_user.transactions:

for transaction in self.current\_user.transactions:

print(f"{transaction.date} - {transaction.transaction\_type}: ${transaction.amount}")

else:

print("No transaction history.")

def withdraw(self):

amount = float(input("Enter amount to withdraw: "))

if amount > self.current\_user.balance:

print("Insufficient balance.")

else:

self.current\_user.balance -= amount

transaction = Transaction(amount, 'Withdrawal')

self.current\_user.add\_transaction(transaction)

print("Withdrawal successful.")

def deposit(self):

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

self.current\_user.balance += amount

transaction = Transaction(amount, 'Deposit')

self.current\_user.add\_transaction(transaction)

print("Deposit successful.")

def transfer(self):

target\_user\_id = input("Enter the target User ID: ")

target\_user = self.atm.get\_user(target\_user\_id)

if not target\_user:

print("Target user not found.")

return

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

if amount > self.current\_user.balance:

print("Insufficient balance.")

else:

self.current\_user.balance -= amount

target\_user.balance += amount

transaction = Transaction(amount, 'Transfer')

self.current\_user.add\_transaction(transaction)

target\_user.add\_transaction(transaction)

print("Transfer successful.")

def main():

atm = ATM()

atm.add\_user(User('user1', '1234', 500))

atm.add\_user(User('user2', '5678', 1000))

atm\_system = ATMSystem(atm)

atm\_system.authenticate\_user()

atm\_system.show\_menu()

if \_\_name\_\_ == "\_\_main\_\_":

main()

output:

Enter User ID: user1

Enter PIN: 1234

Authentication successful!

1. Transactions History

2. Withdraw

3. Deposit

4. Transfer

5. Quit

Choose an option: 2

Enter amount to withdraw: 200

Withdrawal successful.

1. Transactions History

2. Withdraw

3. Deposit

4. Transfer

5. Quit

Choose an option: 3

Enter amount to deposit: 200

Deposit successful.

1. Transactions History

2. Withdraw

3. Deposit

4. Transfer

5. Quit

Choose an option: 4

Enter the target User ID: user2

Enter amount to transfer: 30

Transfer successful.

1. Transactions History

2. Withdraw

3. Deposit

4. Transfer

5. Quit

Choose an option: 1

2024-06-02 11:24:51 - Withdrawal: $200.0

2024-06-02 11:25:05 - Deposit: $200.0

2024-06-02 11:25:28 - Transfer: $30.0

1. Transactions History

2. Withdraw

3. Deposit

4. Transfer

5. Quit

Choose an option: 1

2024-06-02 11:24:51 - Withdrawal: $200.0

2024-06-02 11:25:05 - Deposit: $200.0

2024-06-02 11:25:28 - Transfer: $30.0

1. Transactions History

2. Withdraw

3. Deposit

4. Transfer

5. Quit

Choose an option: 5

Exiting... Thank you!

An exception has occurred, use %tb to see the full traceback.

SystemExit