```
import pickle
import os
import datetime
import numpy as np

# File to store account details
ACCOUNTS_FILE = "accounts.pkl"
TRANSACTIONS_FILE = "transactions.pkl"

class BankAccount:
    def __init__(self, name, account_type, initial_balance):
        self.name = name
        self.account_number = self.generate_account_number()
        self.account_type = account_type
        self.balance = initial_balance
        self.transactions = []

    def generate_account_number(self):
        return str(100000 + len(load_accounts()) + 1)

    def deposit(self, amount):
        if amount > 0:
            self.balance += amount
            self.add_transaction("Deposit", amount)
            return f"Deposited ₹{amount}. New balance: ₹
{self.balance}"
        return "Invalid deposit amount."

    def withdraw(self, amount):
        if 0 < amount <= self.balance:
            self.balance -= amount
            self.add_transaction("Withdrawal", amount)
            return f"Withdrawn ₹{amount}. New balance: ₹
{self.balance}"
        return "Insufficient funds or invalid amount."

    def transfer(self, target_account, amount):
        if 0 < amount <= self.balance:
            self.balance -= amount
            target_account.balance += amount
            self.add_transaction("Transfer Out", amount,
target_account.account_number)
            target_account.add_transaction("Transfer In", amount,
self.account_number)
            return f"Transferred ₹{amount} to Account
{target_account.account_number}. New balance: ₹{self.balance}"
        return "Insufficient funds or invalid amount."

    def add_transaction(self, transaction_type, amount,
target_account=None):
```

```
        transaction = {
            "date": str(datetime.datetime.now()),
            "type": transaction_type,
            "amount": amount,
            "target": target_account,
            "account_number": self.account_number
        }
        self.transactions.append(transaction)
        save_transaction(transaction)

    def get_details(self):
        return f"Account Holder: {self.name}\nAccount Number:
{self.account_number}\nAccount Type: {self.account_type}\nBalance: ₹
{self.balance}\n"

# File Handling Functions

def save_accounts(accounts):
    with open(ACCOUNTS_FILE, "wb") as f:
        pickle.dump(accounts, f)

def load_accounts():
    if os.path.exists(ACCOUNTS_FILE):
        with open(ACCOUNTS_FILE, "rb") as f:
            return pickle.load(f)
    return {}

def save_transaction(transaction):
    transactions = load_transactions()
    transactions.append(transaction)
    with open(TRANSACTIONS_FILE, "wb") as f:
        pickle.dump(transactions, f)

def load_transactions():
    if os.path.exists(TRANSACTIONS_FILE):
        with open(TRANSACTIONS_FILE, "rb") as f:
            data = pickle.load(f)
            return data if isinstance(data, list) else []  # Ensure
it's always a list.
    return []

def transaction_history(account_number):
    transactions = load_transactions()
    account_transactions = [t for t in transactions if
t['account_number'] == account_number]
    if account_transactions:
        for t in account_transactions:
            print(f"Date: {t['date']}, Type: {t['type']}, Amount: ₹
{t['amount']}, Target: {t.get('target', 'N/A')}")
    else:
```

```
        print("No transactions found.")

def generate_report(account):
    transactions = [t for t in load_transactions() if
t['account_number'] == account.account_number]
    deposits = [t['amount'] for t in transactions if t['type'] ==
"Deposit"]
    withdrawals = [t['amount'] for t in transactions if t['type'] ==
"Withdrawal"]

    total_deposits = np.sum(deposits) if deposits else 0
    total_withdrawals = np.sum(withdrawals) if withdrawals else 0
    avg_transaction = np.mean(deposits + withdrawals) if deposits or
withdrawals else 0

    print(f"Total Deposits: ₹{total_deposits}")
    print(f"Total Withdrawals: ₹{total_withdrawals}")
    print(f"Average Transaction Amount: ₹{avg_transaction:.2f}")

# Main Menu

def main():
    accounts = load_accounts()
    while True:
        print("\nBank Account Management System")
        print("1. Open New Account")
        print("2. View Account Details")
        print("3. Deposit Money")
        print("4. Withdraw Money")
        print("5. Transfer Money")
        print("6. View Transaction History")
        print("7. Generate Account Report")
        print("8. Exit")

        choice = input("Enter your choice: ")

        if choice == "1":
            name = input("Enter Account Holder's Name: ")
            acc_type = input("Enter Account Type (Savings/Current): ")
            initial_balance = float(input("Enter Initial Deposit
Amount: "))
            new_acc = BankAccount(name, acc_type, initial_balance)
            accounts[new_acc.account_number] = new_acc
            save_accounts(accounts)
            print(f"Account Created Successfully! Account Number:
{new_acc.account_number}")

        elif choice == "2":
            acc_num = input("Enter Account Number: ")
            if acc_num in accounts:
```

```
                print(accounts[acc_num].get_details())
            else:
                print("Account not found.")

        elif choice == "3":
            acc_num = input("Enter Account Number: ")
            if acc_num in accounts:
                amount = float(input("Enter Deposit Amount: "))
                print(accounts[acc_num].deposit(amount))
                save_accounts(accounts)
            else:
                print("Account not found.")

        elif choice == "4":
            acc_num = input("Enter Account Number: ")
            if acc_num in accounts:
                amount = float(input("Enter Withdrawal Amount: "))
                print(accounts[acc_num].withdraw(amount))
                save_accounts(accounts)
            else:
                print("Account not found.")

        elif choice == "5":
            from_acc = input("Enter Your Account Number: ")
            to_acc = input("Enter Recipient Account Number: ")
            if from_acc in accounts and to_acc in accounts:
                amount = float(input("Enter Transfer Amount: "))
                print(accounts[from_acc].transfer(accounts[to_acc],
amount))
                save_accounts(accounts)
            else:
                print("One or both accounts not found.")

        elif choice == "6":
            acc_num = input("Enter Account Number: ")
            transaction_history(acc_num)

        elif choice == "7":
            acc_num = input("Enter Account Number: ")
            if acc_num in accounts:
                generate_report(accounts[acc_num])
            else:
                print("Account not found.")

        elif choice == "8":
            print("Thank you for using the system!")
            break
        else:
            print("Invalid choice. Try again.")
```

```
if __name__ == "__main__":
    main()


Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  1
Enter Account Holder's Name:  Disha Sindhi
Enter Account Type (Savings/Current):  Savings
Enter Initial Deposit Amount:  50000

Account Created Successfully! Account Number: 100013

Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  2
Enter Account Number:  100013

Account Holder: Disha Sindhi
Account Number: 100013
Account Type: Savings
Balance: ₹50000.0


Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit
```

```
Enter your choice:  2
Enter Account Number:  100013

Account Holder: Disha Sindhi
Account Number: 100013
Account Type: Savings
Balance: ₹50000.0


Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  3
Enter Account Number:  100013
Enter Deposit Amount:  4000

Deposited ₹4000.0. New balance: ₹54000.0

Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  4
Enter Account Number:  100013
Enter Withdrawal Amount:  3000

Withdrawn ₹3000.0. New balance: ₹51000.0

Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit
```

```
Enter your choice:  5
Enter Your Account Number:  100013
Enter Recipient Account Number:  100014

One or both accounts not found.

Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  5
Enter Your Account Number:  100013
Enter Recipient Account Number:  100013
Enter Transfer Amount:  2000

Transferred ₹2000.0 to Account 100013. New balance: ₹51000.0

Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  6
Enter Account Number:  100013

Date: 2025-02-16 16:18:51.090579, Type: Deposit, Amount: ₹4000.0,
Target: None
Date: 2025-02-16 16:19:46.409799, Type: Withdrawal, Amount: ₹3000.0,
Target: None
Date: 2025-02-16 16:20:57.965207, Type: Transfer Out, Amount: ₹2000.0,
Target: 100013
Date: 2025-02-16 16:20:57.979813, Type: Transfer In, Amount: ₹2000.0,
Target: 100013

Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
```

```
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  7
Enter Account Number:  100013

Total Deposits: ₹4000.0
Total Withdrawals: ₹3000.0
Average Transaction Amount: ₹3500.00

Bank Account Management System
1. Open New Account
2. View Account Details
3. Deposit Money
4. Withdraw Money
5. Transfer Money
6. View Transaction History
7. Generate Account Report
8. Exit

Enter your choice:  8

Thank you for using the system!
```