Banking System Simulator (Python OOP Project)

Description:

A command-line banking system simulator using object-oriented programming. It allows creating bank accounts, performing deposits and withdrawals, and maintaining a transaction history using classes and inheritance.

OOP Concepts Used:

- Classes

- Inheritance

- Polymorphism

- File I/O

Features:

- Create bank accounts

- Deposit and withdraw money

- Transaction history

- Basic admin/user flow

Code:

```
class Account:
    def __init__(self, owner, balance=0):
        self.owner = owner
        self.balance = balance
        self.transactions = []

    def deposit(self, amount):
        self.balance += amount
        self.transactions.append(f"Deposited: ${amount}")
        print(f"Deposited ${amount}. New balance: ${self.balance}")

    def withdraw(self, amount):
        if amount > self.balance:
            print("Insufficient funds!")
            return
        self.balance -= amount
        self.transactions.append(f"Withdrew: ${amount}")
        print(f"Withdrew ${amount}. New balance: ${self.balance}")

    def display_balance(self):
        print(f"{self.owner}'s Balance: ${self.balance}")

    def show_transactions(self):
        print(f"Transaction History for {self.owner}:")
        for t in self.transactions:
            print("-", t)

class Bank:
    def __init__(self):
        self.accounts = {}

    def create_account(self, owner):
        if owner in self.accounts:
            print("Account already exists.")
            return
        self.accounts[owner] = Account(owner)
        print(f"Account created for {owner}.")

    def get_account(self, owner):
        return self.accounts.get(owner, None)

def main():
    bank = Bank()
    while True:
        print("\n--- Bank Menu ---")
        print("1. Create Account")
        print("2. Deposit")
        print("3. Withdraw")
        print("4. Show Balance")
        print("5. Show Transactions")
        print("6. Exit")
        choice = input("Enter choice: ")

        if choice == "1":
            name = input("Enter account holder name: ")
            bank.create_account(name)
        elif choice == "2":
            name = input("Enter name: ")
            acc = bank.get_account(name)
            if acc:
                amount = float(input("Enter amount: "))
                acc.deposit(amount)
            else:
                print("Account not found.")
        elif choice == "3":
            name = input("Enter name: ")
            acc = bank.get_account(name)
            if acc:
                amount = float(input("Enter amount: "))
                acc.withdraw(amount)
            else:
                print("Account not found.")
        elif choice == "4":
            name = input("Enter name: ")
            acc = bank.get_account(name)
            if acc:
                acc.display_balance()
            else:
                print("Account not found.")
        elif choice == "5":
            name = input("Enter name: ")
            acc = bank.get_account(name)
            if acc:
                acc.show_transactions()
            else:
                print("Account not found.")
        elif choice == "6":
            print("Exiting Bank System.")
            break
        else:
            print("Invalid choice.")

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