```
import java.util.*;

class BankAccount {
    private static int accountCounter = 1000; // Unique account number generator
    private int accountNumber;
    private String accountHolder;
    private double balance;
    private List<String> transactionHistory;

    public BankAccount(String accountHolder) {
        this.accountHolder = accountHolder;
        this.accountNumber = accountCounter++;
        this.balance = 0.0;
        this.transactionHistory = new ArrayList<>();
        transactionHistory.add("Account created with balance: $0.0");
    }

    public int getAccountNumber() {
        return accountNumber;
    }

    public String getAccountHolder() {
        return accountHolder;
    }

    public double getBalance() {
        return balance;
    }

    public void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            transactionHistory.add("Deposited: $" + amount);
            System.out.println("Deposit successful! New balance: $" + balance);
        } else {
            System.out.println("Invalid deposit amount.");
        }
    }

    public void withdraw(double amount) {
        if (amount > 0 && amount <= balance) {
            balance -= amount;
            transactionHistory.add("Withdrawn: $" + amount);
            System.out.println("Withdrawal successful! New balance: $" + balance);
        } else {
            System.out.println("Invalid withdrawal amount or insufficient
balance.");
        }
    }
```

```
    public void displayAccountDetails() {
        System.out.println("\nAccount Number: " + accountNumber);
        System.out.println("Account Holder: " + accountHolder);
        System.out.println("Current Balance: $" + balance);
        System.out.println("Transaction History:");
        for (String transaction : transactionHistory) {
            System.out.println(" - " + transaction);
        }
    }
}

public class SimpleBankingSystem {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<BankAccount> accounts = new ArrayList<>();

        while (true) {
            System.out.println("\n*** Simple Banking System ***");
            System.out.println("1. Create Account");
            System.out.println("2. Deposit Money");
            System.out.println("3. Withdraw Money");
            System.out.println("4. View Account Details");
            System.out.println("5. Exit");
            System.out.print("Choose an option: ");

            int choice = scanner.nextInt();
            scanner.nextLine(); // Consume newline

            switch (choice) {
                case 1:
                    System.out.print("Enter Account Holder Name: ");
                    String name = scanner.nextLine();
                    BankAccount newAccount = new BankAccount(name);
                    accounts.add(newAccount);
                    System.out.println("Account created successfully! Your Account
Number is: " + newAccount.getAccountNumber());
                    break;

                case 2:
                    System.out.print("Enter Account Number: ");
                    int accNumDep = scanner.nextInt();
                    BankAccount accountDep = findAccount(accounts, accNumDep);
                    if (accountDep != null) {
                        System.out.print("Enter deposit amount: $");
                        double depositAmount = scanner.nextDouble();
                        accountDep.deposit(depositAmount);
                    } else {
                        System.out.println("Account not found.");
                    }
                    break;
```

```
                case 3:
                    System.out.print("Enter Account Number: ");
                    int accNumWith = scanner.nextInt();
                    BankAccount accountWith = findAccount(accounts, accNumWith);
                    if (accountWith != null) {
                        System.out.print("Enter withdrawal amount: $");
                        double withdrawAmount = scanner.nextDouble();
                        accountWith.withdraw(withdrawAmount);
                    } else {
                        System.out.println("Account not found.");
                    }
                    break;

                case 4:
                    System.out.print("Enter Account Number: ");
                    int accNumView = scanner.nextInt();
                    BankAccount accountView = findAccount(accounts, accNumView);
                    if (accountView != null) {
                        accountView.displayAccountDetails();
                    } else {
                        System.out.println("Account not found.");
                    }
                    break;

                case 5:
                    System.out.println("Exiting... Thank you for using the banking
system.");
                    scanner.close();
                    System.exit(0);

                default:
                    System.out.println("Invalid choice! Please try again.");
            }
        }
    }

    private static BankAccount findAccount(List<BankAccount> accounts, int
accountNumber) {
        for (BankAccount acc : accounts) {
            if (acc.getAccountNumber() == accountNumber) {
                return acc;
            }
        }
        return null;
    }
}
```