```
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Scanner;

class BankAccount {
    private String accountNumber;
    private String customerName;
    private double balance;
    private double interestRate;

    public BankAccount(String accountNumber, String customerName, double balance, double
interestRate) {
        this.accountNumber = accountNumber;
        this.customerName = customerName;
        this.balance = balance;
        this.interestRate = interestRate;
    }

    public void deposit(double amount) {
        balance += amount;
    }

    public void withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
        } else {
            System.out.println("Insufficient balance.");
        }
    }

    public void calculateInterest() {
```

```
        double interest = balance * (interestRate / 12);
        balance += interest;
    }

    public String getAccountNumber() {
        return accountNumber;
    }

    public String getCustomerName() {
        return customerName;
    }

    public double getBalance() {
        return balance;
    }
}

class BankManagementSystem {
    private static final int MAX_CASH_INFLOW = 10000;
    private static final double INTEREST_RATE = 0.05;

    private HashMap<String, BankAccount> customerAccounts;
    private ArrayList<Employee> employees;
    private Admin admin;

    public BankManagementSystem() {
        customerAccounts = new HashMap<>();
        employees = new ArrayList<>();
        admin = new Admin();
    }
```

```
    public void registerCustomer(String accountNumber, String customerName, double initialDeposit) {
        if (customerAccounts.containsKey(accountNumber)) {
            System.out.println("Account number already exists.");
            return;
        }

        if (initialDeposit > MAX_CASH_INFLOW) {
            System.out.println("Maximum cash inflow limit exceeded.");
            return;
        }

        BankAccount account = new BankAccount(accountNumber, customerName, initialDeposit,
INTEREST_RATE);
        customerAccounts.put(accountNumber, account);
        System.out.println("Customer registered successfully.");
    }

    public void customerLogin(String accountNumber, String password) {
        if (!customerAccounts.containsKey(accountNumber)) {
            System.out.println("Invalid account number.");
            return;
        }

        BankAccount account = customerAccounts.get(accountNumber);
        System.out.println("Welcome, " + account.getCustomerName() + "!");
        System.out.println("Your current balance is: $" + account.getBalance());

        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter amount to deposit: ");
        double depositAmount = scanner.nextDouble();
        if (depositAmount > MAX_CASH_INFLOW) {
```

```
            System.out.println("Maximum cash inflow limit exceeded.");
            return;
        }
        account.deposit(depositAmount);
        System.out.println("Deposit successful. New balance: $" + account.getBalance());

        System.out.print("Enter amount to withdraw: ");
        double withdrawalAmount = scanner.nextDouble();
        account.withdraw(withdrawalAmount);
        System.out.println("Withdrawal successful. New balance: $" + account.getBalance());
    }

    public void employeeLogin(String employeeId, String password) {
        for (Employee employee : employees) {
            if (employee.getEmployeeId().equals(employeeId) &&
employee.getPassword().equals(password)) {
                System.out.println("Welcome, " + employee.getName() + "!");
                System.out.println("Your position: " + employee.getPosition());
                return;
            }
        }
        System.out.println("Invalid employee ID or password.");
    }

    public void adminLogin(String username, String password) {
        if (admin.getUsername().equals(username) && admin.getPassword().equals(password)) {
            System.out.println("Welcome, Admin!");
            System.out.println("Customer Accounts:");
            for (BankAccount account : customerAccounts.values()) {
                System.out.println("Account Number: " + account.getAccountNumber() + ", Customer
Name: " + account.getCustomerName() + ", Balance: $" + account.getBalance());
            }
```

```
            System.out.println("Employees:");
            for (Employee employee : employees) {
                System.out.println("Employee ID: " + employee.getEmployeeId() + ", Name: " +
employee.getName() + ", Position: " + employee.getPosition());
            }
        } else {
            System.out.println("Invalid admin username or password.");
        }
    }
}

class Employee {
    private String employeeId;
    private String name;
    private String position;
    private String password;

    public Employee(String employeeId, String name, String position, String password) {
        this.employeeId = employeeId;
        this.name = name;
        this.position = position;
        this.password = password;
    }

    public String getEmployeeId() {
        return employeeId;
    }

    public String getName() {
        return name;
    }
```

```
    public String getPosition() {
        return position;
    }

    public String getPassword() {
        return password;
    }
}

class Admin {
    private final String username = "admin";
    private final String password = "password";

    public String getUsername() {
        return username;
    }

    public String getPassword() {
        return password;
    }
}

public class BankManagementSystemApp {
    public static void main(String[] args) {
        BankManagementSystem system = new BankManagementSystem();

        // Register customers
        system.registerCustomer("123456789", "John Doe", 5000);
        system.registerCustomer("987654321", "Jane Smith", 8000);
```

```
        // Employee login
        system.employeeLogin("E001", "password1");

        // Customer login
        system.customerLogin("123456789", "password");

        // Admin login
        system.adminLogin("admin", "password");
    }
}
```