JAVA PROJECT
MODULE 5

Project Overview

Packages and Structure

- **com.insurance**:
  The root package containing the main application class.
- **com.insurance.models**:
  Contains the data models (CustomerDetails, Payment, Policy, Claim) that represent the entities in the insurance system.
- **com.insurance.intf**:
  Defines interfaces that declare methods for CRUD (Create, Read, Update, Delete) operations for each entity.
- **com.insurance.impl**:
  Contains the implementation classes that provide the actual business logic for the methods defined in the interfaces.
- **com.insurance.service**:
  Provides service classes that manage operations on the entities, making use of the implementation classes.
- **com.insurance.repository**:
  Provides sample data to populate the entities for testing and demonstration purposes.

Entities and Models

- **CustomerDetails**:
  Represents a customer in the insurance system, including fields like customerId, name, and email.
- **Payment**:
  Represents a payment made by a customer, including fields like paymentId, customerId, and amount.
- **Policy**:
  Represents an insurance policy, including fields like policyId, customerId, and policyType.
- **Claim**:
  Represents an insurance claim made against a policy, including fields like claimId, policyId, and status.

Interfaces and Implementations

- **Interfaces**:
  Each entity has an associated interface (e.g., CustomerDetailsIntf, PaymentIntf, etc.) that defines the methods for CRUD operations.

- **Implementations**:
  The implementation classes (e.g., CustomerDetailsImpl, PaymentImpl, etc.) provide the actual code to perform these operations using collections like List and Map.
- **Repositories**
  The repository classes (e.g., CustomerDetailsRepository, PaymentRepository, etc.) are used to simulate a data source by providing sample data that can be used by the application.
- **Services**
  The service classes (e.g., CustomerDetailsService, PaymentService, etc.) are responsible for managing operations on the entities. They interact with the implementation classes to perform CRUD operations and display the results.

Main Application

- **Insurance Application**:

  The main class that ties everything together. It creates instances of the service classes and calls their methods to perform operations on the entities.

Functionality

- **Create Operations**:
  Adding new records for customers, payments, policies, and claims.
- **Read Operations**:
  Fetching and displaying details of specific customers, payments, policies, and claims.
- **Update Operations**:
  Modifying existing records for customers, payments, policies, and claims.
- **Delete Operations**:

  Removing records for customers, payments, policies, and claims.

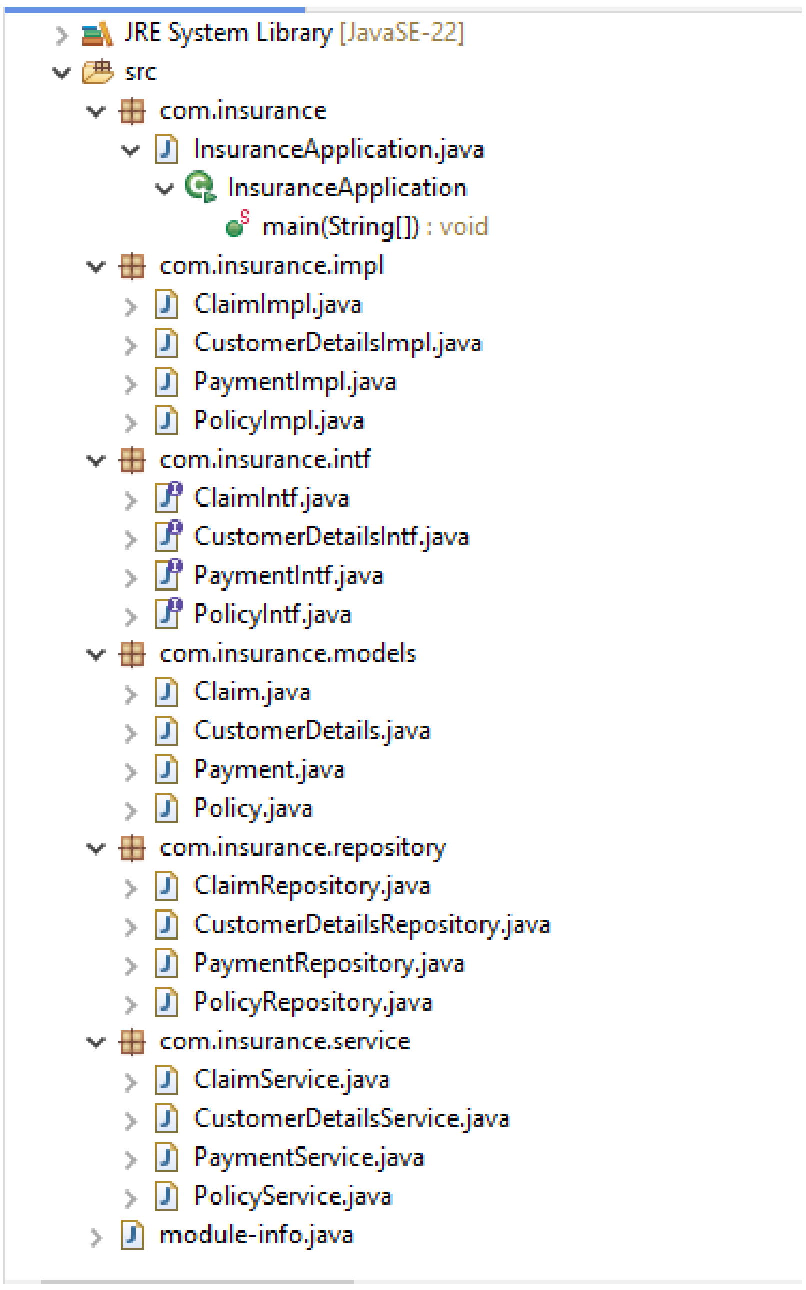
Example of How the System Works

- **Customer Management**
  Create customer records using sample data from the repository.
  Read and display a customer's details.
  Update the customer's information.
  Delete a customer record and display the remaining customers.
- **Payment Management**
  Similar operations are performed for payments, managing payment records for customers.
- **Policy Management**
  Policies are managed similarly, with the ability to add, update, read, and delete policy records.

- **Claim Management**

  Claims are managed with operations that allow adding, updating, reading, and deleting claim records associated with specific policies.

PROJECT STRUCTURE



```
> JRE System Library [JavaSE-22]
v src
  v com.insurance
    v InsuranceApplication.java
      v InsuranceApplication
          main(String[]) : void
  v com.insurance.impl
    > ClaimImpl.java
    > CustomerDetailsImpl.java
    > PaymentImpl.java
    > PolicyImpl.java
  v com.insurance.intf
    > ClaimIntf.java
    > CustomerDetailsIntf.java
    > PaymentIntf.java
    > PolicyIntf.java
  v com.insurance.models
    > Claim.java
    > CustomerDetails.java
    > Payment.java
    > Policy.java
  v com.insurance.repository
    > ClaimRepository.java
    > CustomerDetailsRepository.java
    > PaymentRepository.java
    > PolicyRepository.java
  v com.insurance.service
    > ClaimService.java
    > CustomerDetailsService.java
    > PaymentService.java
    > PolicyService.java
  > module-info.java
```

CustomerDetails.java

```
package com.insurance.models;

public class CustomerDetails {
    private String customerId;
    private String name;
    private String address;
    private String contactNumber;

    public CustomerDetails(String customerId, String name, String address, String
contactNumber) {
        this.customerId = customerId;
        this.name = name;
        this.address = address;
        this.contactNumber = contactNumber;
    }

    // Getters and Setters
    public String getCustomerId() {
        return customerId;
    }

    public void setCustomerId(String customerId) {
        this.customerId = customerId;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public String getContactNumber() {
        return contactNumber;
    }

    public void setContactNumber(String contactNumber) {
        this.contactNumber = contactNumber;
    }
}
```

CustomerDetailsIntf.java

```
package com.insurance.intf;

import com.insurance.models.CustomerDetails;

import java.util.List;

public interface CustomerDetailsIntf {
   void addCustomer(CustomerDetails customer);
   void addAllCustomers(List<CustomerDetails> customers);
   CustomerDetails getCustomer(String customerId);
   void updateCustomer(String customerId, CustomerDetails customer);
   void deleteCustomer(String customerId);
}
```

CustomerDetailsImpl.java

```
package com.insurance.impl;

import com.insurance.intf.CustomerDetailsIntf;
import com.insurance.models.CustomerDetails;

import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class CustomerDetailsImpl implements CustomerDetailsIntf {
   private Map<String, CustomerDetails> customers = new HashMap<>();

   @Override
   public void addCustomer(CustomerDetails customer) {
      customers.put(customer.getCustomerId(), customer);
   }

   @Override
   public void addAllCustomers(List<CustomerDetails> customersList) {
      for (CustomerDetails customer : customersList) {
         customers.put(customer.getCustomerId(), customer);
      }
   }

   @Override
   public CustomerDetails getCustomer(String customerId) {
      return customers.get(customerId);
   }

   @Override
   public void updateCustomer(String customerId, CustomerDetails customer) {
      customers.put(customerId, customer);
```

```
    }

    @Override
    public void deleteCustomer(String customerId) {
        customers.remove(customerId);
    }
}
```

CustomerDetailsRepository.java

```
package com.insurance.repository;

import com.insurance.models.CustomerDetails;

import java.util.ArrayList;
import java.util.List;

public class CustomerDetailsRepository {
    public List<CustomerDetails> getSampleCustomers() {
        List<CustomerDetails> customers = new ArrayList<>();
        customers.add(new CustomerDetails("C001", "Alice Brown", "123 Main St", "555-
1234"));
        customers.add(new CustomerDetails("C002", "Bob White", "456 Oak St", "555-
5678"));
        return customers;
    }
}
```

CustomerDetailsService.java

```
package com.insurance.service;

import com.insurance.impl.CustomerDetailsImpl;
import com.insurance.models.CustomerDetails;
import com.insurance.repository.CustomerDetailsRepository;

import java.util.List;

public class CustomerDetailsService {
    public void manageCustomers() {
        CustomerDetailsImpl customerImpl = new CustomerDetailsImpl();
        CustomerDetailsRepository customerRepository = new CustomerDetailsRepository();

        // Add all customers
        List<CustomerDetails> customers = customerRepository.getSampleCustomers();
        customerImpl.addAllCustomers(customers);

        // Read a customer
        CustomerDetails customer = customerImpl.getCustomer("C001");
        System.out.println("Customer Details: ");
        System.out.println("Name: " + customer.getName() + ", Address: " +
customer.getAddress());
```

```
        // Update a customer
        customer.setContactNumber("555-9999");
        customerImpl.updateCustomer("C001", customer);
        System.out.println("Updated Customer Details: ");
        System.out.println("Name: " + customer.getName() + ", Contact Number: " +
customer.getContactNumber());

        // Delete a customer
        customerImpl.deleteCustomer("C002");

        // Display all remaining customers
        System.out.println("Remaining Customers: ");
        for (CustomerDetails remainingCustomer : customerImpl.customers.values()) {
            System.out.println("Customer ID: " + remainingCustomer.getCustomerId() + ",
Name: " + remainingCustomer.getName());
        }
    }
}
```

Payment.java

```
package com.insurance.models;

public class Payment {
    private String paymentId;
    private String customerId;
    private double amount;
    private String date;

    public Payment(String paymentId, String customerId, double amount, String date) {
        this.paymentId = paymentId;
        this.customerId = customerId;
        this.amount = amount;
        this.date = date;
    }

    // Getters and Setters
    public String getPaymentId() {
        return paymentId;
    }

    public void setPaymentId(String paymentId) {
        this.paymentId = paymentId;
    }

    public String getCustomerId() {
        return customerId;
    }

    public void setCustomerId(String customerId) {
        this.customerId = customerId;
    }

    public double getAmount() {
        return amount;
    }

    public void setAmount(double amount) {
        this.amount = amount;
    }

    public String getDate() {
        return date;
    }

    public void setDate(String date) {
        this.date = date;
    }
}
```

PaymentIntf.java

```
package com.insurance.intf;

import com.insurance.models.Payment;

import java.util.List;

public interface PaymentIntf {
    void addPayment(Payment payment);
    void addAllPayments(List<Payment> payments);
    Payment getPayment(String paymentId);
    void updatePayment(String paymentId, Payment payment);
    void deletePayment(String paymentId);
}
```

PaymentImpl.java

```
package com.insurance.impl;

import com.insurance.intf.PaymentIntf;
import com.insurance.models.Payment;

import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class PaymentImpl implements PaymentIntf {
    private Map<String, Payment> payments = new HashMap<>();

    @Override
    public void addPayment(Payment payment) {
        payments.put(payment.getPaymentId(), payment);
    }

    @Override
    public void addAllPayments(List<Payment> paymentsList) {
        for (Payment payment : paymentsList) {
            payments.put(payment.getPaymentId(), payment);
        }
    }

    @Override
    public Payment getPayment(String paymentId) {
        return payments.get(paymentId);
    }

    @Override
    public void updatePayment(String paymentId, Payment payment) {
        payments.put(paymentId, payment);
    }

    @Override
    public void deletePayment(String paymentId) {
        payments.remove(paymentId);
    }
}
```

PaymentRepository.java

```
package com.insurance.repository;

import com.insurance.models.Payment;

import java.util.ArrayList;
import java.util.List;

public class PaymentRepository {
    public List<Payment> getSamplePayments() {
        List<Payment> payments = new ArrayList<>();
        payments.add(new Payment("P001", "C001", 200.00, "2024-08-01"));
        payments.add(new Payment("P002", "C002", 300.00, "2024-08-02"));
        return payments;
    }
}
```

PaymentService.java

```
package com.insurance.service;

import com.insurance.impl.PaymentImpl;
import com.insurance.models.Payment;
import com.insurance.repository.PaymentRepository;
import java.util.List;
```

```
public class PaymentService {
   public void managePayments() {
      PaymentImpl paymentImpl = new PaymentImpl();
      PaymentRepository paymentRepository = new PaymentRepository();

      // Add all payments
      List<Payment> payments = paymentRepository.getSamplePayments();
      paymentImpl.addAllPayments(payments);

      // Read a payment
      Payment payment = paymentImpl.getPayment("P001");
      System.out.println("Payment Details: ");
      System.out.println("Amount: " + payment.getAmount() + ", Customer ID: " +
payment.getCustomerId());

      // Update a payment
      payment.setAmount(550.00);
      paymentImpl.updatePayment("P001", payment);
      System.out.println("Updated Payment Details: ");
      System.out.println("Amount: " + payment.getAmount() + ", Customer ID: " +
payment.getCustomerId());

      // Delete a payment
      paymentImpl.deletePayment("P002");

      // Display all remaining payments
      System.out.println("Remaining Payments: ");
      for (Payment remainingPayment : paymentImpl.payments.values()) {
         System.out.println("Payment ID: " + remainingPayment.getPaymentId() + ",
Amount: " + remainingPayment.getAmount());
      }
   }
}
```

Policy.java

```
package com.insurance.models;

public class Policy {
   private String policyId;
   private String customerId;
   private String policyType;

   public Policy(String policyId, String customerId, String policyType) {
      this.policyId = policyId;
      this.customerId = customerId;
      this.policyType = policyType;
   }

   // Getters and Setters
   public String getPolicyId() {
      return policyId;
```

```
    }

    public void setPolicyId(String policyId) {
        this.policyId = policyId;
    }

    public String getCustomerId() {
        return customerId;
    }

    public void setCustomerId(String customerId) {
        this.customerId = customerId;
    }

    public String getPolicyType() {
        return policyType;
    }

    public void setPolicyType(String policyType) {
        this.policyType = policyType;
    }
}
```

PolicyIntf.java

```
package com.insurance.intf;

import com.insurance.models.Policy;
import java.util.List;

public interface PolicyIntf {
    void addPolicy(Policy policy);
    void addAllPolicies(List<Policy> policies);
    Policy getPolicy(String policyId);
    void updatePolicy(String policyId, Policy policy);
    void deletePolicy(String policyId);
}
```

PolicyImpl.java

```
package com.insurance.impl;

import com.insurance.intf.PolicyIntf;
import com.insurance.models.Policy;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class PolicyImpl implements PolicyIntf {
    private Map<String, Policy> policies = new HashMap<>();

    @Override
    public void addPolicy(Policy policy) {
        policies.put(policy.getPolicyId(), policy);
    }

    @Override
    public void addAllPolicies(List<Policy> policiesList) {
        for (Policy policy : policiesList) {
            policies.put(policy.getPolicyId(), policy);
        }
    }

    @Override
    public Policy getPolicy(String policyId) {
        return policies.get(policyId);
    }

    @Override
    public void updatePolicy(String policyId, Policy policy) {
        policies.put(policyId, policy);
    }

    @Override
    public void deletePolicy(String policyId) {
        policies.remove(policyId);
    }
}
```

PolicyRepository.java

```
package com.insurance.repository;

import com.insurance.models.Policy;
import java.util.ArrayList;
import java.util.List;

public class PolicyRepository {
    public List<Policy> getSamplePolicies() {
        List<Policy> policies = new ArrayList<>();
        policies.add(new Policy("PL001", "C001", "Health"));
        policies.add(new Policy("PL002", "C002", "Auto"));
        return policies;
    }
}
```

PolicyService.java

```
package com.insurance.service;

import com.insurance.impl.PolicyImpl;
import com.insurance.models.Policy;
import com.insurance.repository.PolicyRepository;
import java.util.List;
```

```
public class PolicyService {
    public void managePolicies() {
        PolicyImpl policyImpl = new PolicyImpl();
        PolicyRepository policyRepository = new PolicyRepository();

        // Add all policies
        List<Policy> policies = policyRepository.getSamplePolicies();
        policyImpl.addAllPolicies(policies);

        // Read a policy
        Policy policy = policyImpl.getPolicy("PL001");
        System.out.println("Policy Details: ");
        System.out.println("Policy Type: " + policy.getPolicyType() + ", Customer ID: " +
policy.getCustomerId());

        // Update a policy
        policy.setPolicyType("Life");
        policyImpl.updatePolicy("PL001", policy);
        System.out.println("Updated Policy Details: ");
        System.out.println("Policy Type: " + policy.getPolicyType() + ", Customer ID: " +
policy.getCustomerId());

        // Delete a policy
        policyImpl.deletePolicy("PL002");

        // Display all remaining policies
        System.out.println("Remaining Policies: ");
        for (Policy remainingPolicy : policyImpl.policies.values()) {
            System.out.println("Policy ID: " + remainingPolicy.getPolicyId() + ", Policy Type: "
+ remainingPolicy.getPolicyType());
        }
    }
}
```

Claim.java

```
package com.insurance.models;

public class Claim {
    private String claimId;
    private String policyId;
    private String status;

    public Claim(String claimId, String policyId, String status) {
        this.claimId = claimId;
        this.policyId = policyId;
        this.status = status;
    }

    // Getters and Setters
    public String getClaimId() {
        return claimId;
    }

    public void setClaimId(String claimId) {
        this.claimId = claimId;
    }

    public String getPolicyId() {
        return policyId;
    }

    public void setPolicyId(String policyId) {
        this.policyId = policyId;
    }

    public String getStatus() {
        return status;
    }

    public void setStatus(String status) {
        this.status = status;
    }
}
```

ClaimIntf.java

```
package com.insurance.intf;

import com.insurance.models.Claim;
import java.util.List;

public interface ClaimIntf {
    void addClaim(Claim claim);
    void addAllClaims(List<Claim> claims);
    Claim getClaim(String claimId);
    void updateClaim(String claimId, Claim claim);
    void deleteClaim(String claimId);
}
```

ClaimImpl.java

```
package com.insurance.impl;

import com.insurance.intf.ClaimIntf;
import com.insurance.models.Claim;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class ClaimImpl implements ClaimIntf {
    private Map<String, Claim> claims = new HashMap<>();
    @Override
    public void addClaim(Claim claim) {
        claims.put(claim.getClaimId(), claim);
    }

    @Override
    public void addAllClaims(List<Claim> claimsList) {
        for (Claim claim : claimsList) {
            claims.put(claim.getClaimId(), claim);
        }
    }

    @Override
    public Claim getClaim(String claimId) {
        return claims.get(claimId);
    }

    @Override
    public void updateClaim(String claimId, Claim claim) {
        claims.put(claimId, claim);
    }

    @Override
    public void deleteClaim(String claimId) {
        claims.remove(claimId);
    }
}
```

ClaimRepository.java

```
package com.insurance.repository;

import com.insurance.models.Claim;
import java.util.ArrayList;
import java.util.List;

public class ClaimRepository {
    public List<Claim> getSampleClaims() {
        List<Claim> claims = new ArrayList<>();
        claims.add(new Claim("CL001", "PL001", "Pending"));
        claims.add(new Claim("CL002", "PL002", "Approved"));
        return claims;
    }
}
```

ClaimService.java

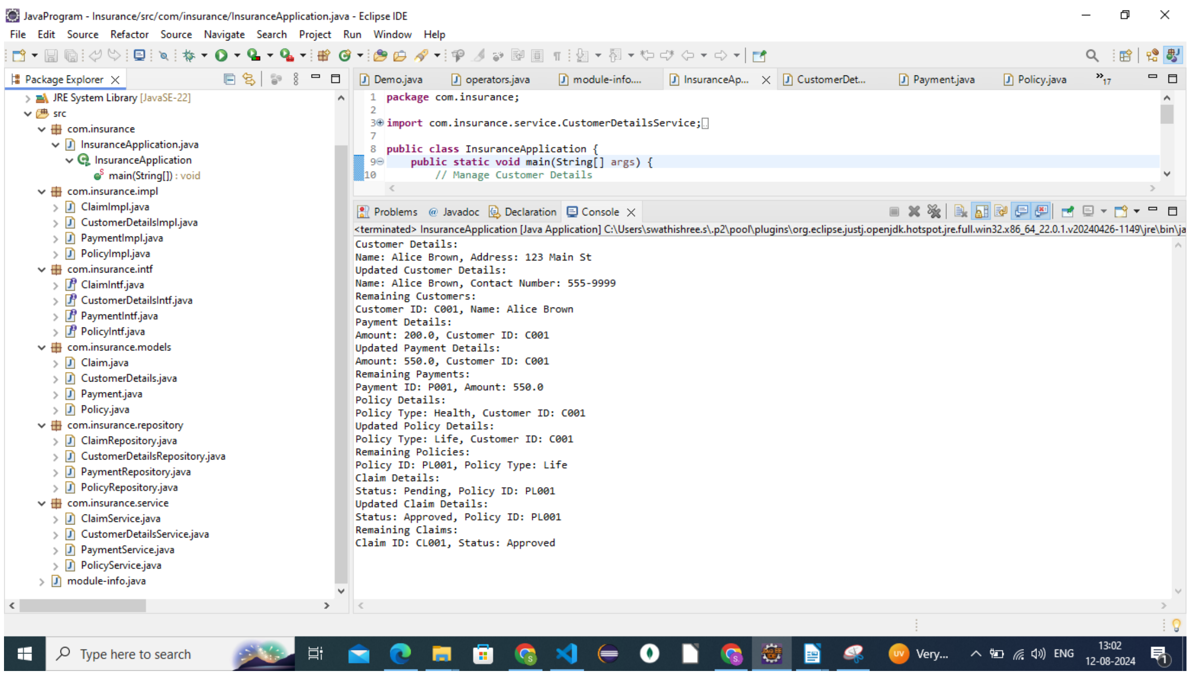
```
package com.insurance.service;

import com.insurance.impl.ClaimImpl;
import com.insurance.models.Claim;
import com.insurance.repository.ClaimRepository;
import java.util.List;

public class ClaimService {
    public void manageClaims() {
        ClaimImpl claimImpl = new ClaimImpl();
        ClaimRepository claimRepository = new ClaimRepository();

        // Add all claims
        List<Claim> claims = claimRepository.getSampleClaims();
        claimImpl.addAllClaims(claims);

        // Read a claim
        Claim claim = claimImpl.getClaim("CL001");
        System.out.println("Claim Details: ");
        System.out.println("Status: " + claim.getStatus() + ", Policy ID: " +
claim.getPolicyId());

        // Update a claim
        claim.setStatus("Approved");
        claimImpl.updateClaim("CL001", claim);
        System.out.println("Updated Claim Details: ");
        System.out.println("Status: " + claim.getStatus() + ", Policy ID: " +
claim.getPolicyId());

        // Delete a claim
        claimImpl.deleteClaim("CL002");

        // Display all remaining claims
        System.out.println("Remaining Claims: ");
        for (Claim remainingClaim : claimImpl.claims.values()) {
            System.out.println("Claim ID: " + remainingClaim.getClaimId() + ", Status: " +
remainingClaim.getStatus());
        }
    }
}
```

Main Method (InsuranceApplication.java)

```
package com.insurance;

import com.insurance.service.CustomerDetailsService;
import com.insurance.service.PaymentService;
import com.insurance.service.PolicyService;
import com.insurance.service.ClaimService;

public class InsuranceApplication {
    public static void main(String[] args) {
        // Manage Customer Details
        CustomerDetailsService customerService = new CustomerDetailsService();
        customerService.manageCustomers();

        // Manage Payments
```

```
            PaymentService paymentService = new PaymentService();
            paymentService.managePayments();

            // Manage Policies
            PolicyService policyService = new PolicyService();
            policyService.managePolicies();

            // Manage Claims
            ClaimService claimService = new ClaimService();
            claimService.manageClaims();
        }
    }
```

OUTPUT:



```
package com.insurance;

import com.insurance.service.CustomerDetailsService;

public class InsuranceApplication {
    public static void main(String[] args) {
        // Manage Customer Details
```

```
<terminated> InsuranceApplication [Java Application] C:\Users\swathishree.s\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-1149\jre\bin\ja
Customer Details:
Name: Alice Brown, Address: 123 Main St
Updated Customer Details:
Name: Alice Brown, Contact Number: 555-9999
Remaining Customers:
Customer ID: C001, Name: Alice Brown
Payment Details:
Amount: 200.0, Customer ID: C001
Updated Payment Details:
Amount: 550.0, Customer ID: C001
Remaining Payments:
Payment ID: P001, Amount: 550.0
Policy Details:
Policy Type: Health, Customer ID: C001
Updated Policy Details:
Policy Type: Life, Customer ID: C001
Remaining Policies:
Policy ID: PL001, Policy Type: Life
Claim Details:
Status: Pending, Policy ID: PL001
Updated Claim Details:
Status: Approved, Policy ID: PL001
Remaining Claims:
Claim ID: CL001, Status: Approved
```