**NAME :** K.T.JENEFER

**BATCH :** 2 JAVA

**PACKAGE ENTITY**

**CLASS USER**

|  |
| --- |
| package entity;  public class user {  private int userId;  private String username;  private String password;  private String role;  public user() {}  public user(int userId, String username, String password, String role) {  this.userId = userId;  this.username = username;  this.password = password;  this.role = role;  }  public int getUserId() {  return userId;  }  public void setUserId(int userId) {  this.userId = userId;  }  public String getUsername() {  return username;  }  public void setUsername(String username) {  this.username = username;  }  public String getPassword() {  return password;  }  public void setPassword(String password) {  this.password = password;  }  public String getRole() {  return role;  }  public void setRole(String role) {  this.role = role;  }  public String toString() {  return "User [userId=" + userId + ", username=" + username + ", role=" + role + "]";  }  } |

**CLASS CLIENT**

|  |
| --- |
| package entity;  public class client {  private int clientId;  private String clientName;  private String contactInfo;  private String policy;    public client() {}  public client(int clientId, String clientName, String contactInfo, String policy) {  this.clientId = clientId;  this.clientName = clientName;  this.contactInfo = contactInfo;  this.policy = policy;  }  public int getClientId() {  return clientId;  }  public void setClientId(int clientId) {  this.clientId = clientId;  }  public String getClientName() {  return clientName;  }  public void setClientName(String clientName) {  this.clientName = clientName;  }  public String getContactInfo() {  return contactInfo;  }  public void setContactInfo(String contactInfo) {  this.contactInfo = contactInfo;  }  public String getPolicy() {  return policy;  }  public void setPolicy(String policy) {  this.policy = policy;  }  public String toString() {  return "Client [clientId=" + clientId + ", clientName=" + clientName + ", contactInfo=" + contactInfo + "]";  }  } |

**CLASS CLAIM**

|  |
| --- |
| package entity;  import java.util.Date;  public class Claim {  private int claimId;  private String claimNumber;  private Date dateFiled;  private double claimAmount;  private String status;  private String policy;  private client client;    public Claim() {}  public Claim(int claimId, String claimNumber, Date dateFiled, double claimAmount, String status, String policy, client client) {  this.claimId = claimId;  this.claimNumber = claimNumber;  this.dateFiled = dateFiled;  this.claimAmount = claimAmount;  this.status = status;  this.policy = policy;  this.client = client;  }  public int getClaimId() {  return claimId;  }  public void setClaimId(int claimId) {  this.claimId = claimId;  }  public String getClaimNumber() {  return claimNumber;  }  public void setClaimNumber(String claimNumber) {  this.claimNumber = claimNumber;  }  public Date getDateFiled() {  return dateFiled;  }  public void setDateFiled(Date dateFiled) {  this.dateFiled = dateFiled;  }  public double getClaimAmount() {  return claimAmount;  }  public void setClaimAmount(double claimAmount) {  this.claimAmount = claimAmount;  }  public String getStatus() {  return status;  }  public void setStatus(String status) {  this.status = status;  }  public String getPolicy() {  return policy;  }  public void setPolicy(String policy) {  this.policy = policy;  }  public client getClient() {  return client;  }  public void setClient(client client) {  this.client = client;  }  public String toString() {  return "Claim [claimId=" + claimId + ", claimNumber=" + claimNumber + ", dateFiled=" + dateFiled + ", claimAmount=" + claimAmount +  ", status=" + status + ", policy=" + policy + ", client=" + client + "]";  }  } |

**CLASS PAYMENT**

|  |
| --- |
| package entity;  import java.util.Date;  public class Payment {  private int paymentId;  private Date paymentDate;  private double paymentAmount;  private client client;  public Payment() {}  public Payment(int paymentId, Date paymentDate, double paymentAmount, client client) {  this.paymentId = paymentId;  this.paymentDate = paymentDate;  this.paymentAmount = paymentAmount;  this.client = client;  }  public int getPaymentId() {  return paymentId;  }  public void setPaymentId(int paymentId) {  this.paymentId = paymentId;  }  public Date getPaymentDate() {  return paymentDate;  }  public void setPaymentDate(Date paymentDate) {  this.paymentDate = paymentDate;  }  public double getPaymentAmount() {  return paymentAmount;  }  public void setPaymentAmount(double paymentAmount) {  this.paymentAmount = paymentAmount;  }  public client getClient() {  return client;  }  public void setClient(client client) {  this.client = client;  }  public String toString() {  return "Payment [paymentId=" + paymentId + ", paymentDate=" + paymentDate + ", paymentAmount=" + paymentAmount + ", client=" + client + "]";  }  } |

**CLASS POLICY**

|  |
| --- |
| package entity;  public class Policy {  private int policyId;  private String policyName;  private String policyType;  private double coverageAmount;  private double premium;    public Policy() {}  public Policy(int policyId, String policyName, String policyType, double coverageAmount, double premium) {  this.policyId = policyId;  this.policyName = policyName;  this.policyType = policyType;  this.coverageAmount = coverageAmount;  this.premium = premium;  }  public int getPolicyId() {  return policyId;  }  public void setPolicyId(int policyId) {  this.policyId = policyId;  }  public String getPolicyName() {  return policyName;  }  public void setPolicyName(String policyName) {  this.policyName = policyName;  }  public String getPolicyType() {  return policyType;  }  public void setPolicyType(String policyType) {  this.policyType = policyType;  }  public double getCoverageAmount() {  return coverageAmount;  }  public void setCoverageAmount(double coverageAmount) {  this.coverageAmount = coverageAmount;  }  public double getPremium() {  return premium;  }  public void setPremium(double premium) {  this.premium = premium;  }  public String toString() {  return "Policy [policyId=" + policyId + ", policyName=" + policyName + ", policyType=" + policyType +  ", coverageAmount=" + coverageAmount + ", premium=" + premium + "]";  }  } |

**PACKAGE DAO**

**CLASS INSURANCESERVICEIMPL**

|  |
| --- |
| package dao;  import entity.Policy;  import java.util.ArrayList;  import java.util.List;  public abstract class InsuranceServiceImpl implements IPolicyService {  protected List<Policy> policies = new ArrayList<>();  public boolean createPolicy(Policy policy) {  return policies.add(policy);  }  public abstract Policy getPolicy(int policyId);  public abstract List<Policy> getAllPolicies();  public boolean updatePolicy(Policy policy) {  int index = -1;  for (int i = 0; i < policies.size(); i++) {  if (policies.get(i).getPolicyId() == policy.getPolicyId()) {  index = i;  break;  }  }  if (index != -1) {  policies.set(index, policy);  return true;  }  return false;  }  public boolean deletePolicy(int policyId) {  return policies.removeIf(policy -> policy.getPolicyId() == policyId);  }  } |

**CLASS IPOLICYSERVICE**

|  |
| --- |
| package dao;  import entity.Policy;  import java.util.List;  public interface IPolicyService {  boolean createPolicy(Policy policy);  Policy getPolicy(int policyId);  List<Policy> getAllPolicies();  boolean updatePolicy(Policy policy);  boolean deletePolicy(int policyId);  } |

**CLASS CONCRETEINSURANCESERVICE**

|  |
| --- |
| package dao;  import entity.Policy;  import java.util.List;  public class ConcreteInsuranceService extends InsuranceServiceImpl {  public Policy getPolicy(int policyId) {  return policies.stream()  .filter(policy -> policy.getPolicyId() == policyId)  .findFirst()  .orElse(null);  }  public List<Policy> getAllPolicies() {  return policies;  }  } |

**PACKAGE EXCEPTION**

**CLASS POLICYNOTFOUNDEXCEPTION**

|  |
| --- |
| package exception;  public class PolicyNotFoundException extends Exception {  public PolicyNotFoundException(String message) {  super(message);  }  } |

**PACKAGE UTIL**

**CLASS DBCONNUTIL**

|  |
| --- |
| package util;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.SQLException;  public class DBConnUtil {  private static Connection connection = null;  public static Connection getConnection() {  if (connection == null) {  try {  String url = PropertyUtil.getPropertyString("db.properties");  connection = DriverManager.getConnection(url);  } catch (SQLException e) {  e.printStackTrace();  }  }  return connection;  }  } |

**CLASS PROPERTYUTIL**

|  |
| --- |
| package util;  import java.io.FileInputStream;  import java.io.IOException;  import java.util.Properties;  public class PropertyUtil {  public static String getPropertyString(String fileName) {  Properties properties = new Properties();  try (FileInputStream fis = new FileInputStream(fileName)) {  properties.load(fis);  String host = properties.getProperty("db.host");  String dbname = properties.getProperty("db.name");  String user = properties.getProperty("db.user");  String password = properties.getProperty("db.password");  String port = properties.getProperty("db.port");  return "jdbc:mysql://" + host + ":" + port + "/" + dbname + "?user=" + user + "&password=" + password;  } catch (IOException e) {  e.printStackTrace();  }  return null;  }  } |

**PACKAGE MAIN**

**CLASS MAINMODULE**

|  |
| --- |
| package main;  import dao.ConcreteInsuranceService;  import dao.IPolicyService;  import entity.Policy;  import exception.PolicyNotFoundException;  import java.util.List;  import java.util.Scanner;  public class MainModule {  public static void main(String[] args) {  IPolicyService policyService = new ConcreteInsuranceService();  Scanner scanner = new Scanner(System.in);  boolean exit = false;  while (!exit) {  System.out.println("\nInsurance Management System:");  System.out.println("1. Create Policy");  System.out.println("2. View Policy");  System.out.println("3. View All Policies");  System.out.println("4. Update Policy");  System.out.println("5. Delete Policy");  System.out.println("6. Exit");  System.out.print("Enter your choice: ");  int choice = scanner.nextInt();  switch (choice) {  case 1:  System.out.print("Enter Policy ID: ");  int policyId = scanner.nextInt();  System.out.print("Enter Policy Name: ");  String policyName = scanner.next();  System.out.print("Enter Policy Type: ");  String policyType = scanner.next();  System.out.print("Enter Coverage Amount: ");  double coverageAmount = scanner.nextDouble();  System.out.print("Enter Premium: ");  double premium = scanner.nextDouble();  Policy newPolicy = new Policy(policyId, policyName, policyType, coverageAmount, premium);  if (policyService.createPolicy(newPolicy)) {  System.out.println("Policy created successfully.");  } else {  System.out.println("Error creating policy.");  }  break;  case 2:  // View Policy  System.out.print("Enter Policy ID to View: ");  int viewId = scanner.nextInt();  try {  Policy policy = policyService.getPolicy(viewId);  if (policy != null) {  System.out.println(policy);  } else {  throw new PolicyNotFoundException("Policy not found!");  }  } catch (PolicyNotFoundException e) {  System.out.println(e.getMessage());  }  break;  case 3:  System.out.println("All Policies:");  List<Policy> policies = policyService.getAllPolicies();  if (policies.isEmpty()) {  System.out.println("No policies available.");  } else {  for (Policy p : policies) {  System.out.println(p);  }  }  break;  case 4:  System.out.print("Enter Policy ID to Update: ");  int updateId = scanner.nextInt();  System.out.print("Enter New Policy Name: ");  String newName = scanner.next();  System.out.print("Enter New Policy Type: ");  String newType = scanner.next();  System.out.print("Enter New Coverage Amount: ");  double newCoverageAmount = scanner.nextDouble();  System.out.print("Enter New Premium: ");  double newPremium = scanner.nextDouble();  Policy updatedPolicy = new Policy(updateId, newName, newType, newCoverageAmount, newPremium);  if (policyService.updatePolicy(updatedPolicy)) {  System.out.println("Policy updated successfully.");  } else {  System.out.println("Error updating policy.");  }  break;  case 5:  System.out.print("Enter Policy ID to Delete: ");  int deleteId = scanner.nextInt();  if (policyService.deletePolicy(deleteId)) {  System.out.println("Policy deleted successfully.");  } else {  System.out.println("Error deleting policy.");  }  break;  case 6:  exit = true;  System.out.println("Exiting the Insurance Management System.");  break;  default:  System.out.println("Invalid choice. Please try again.");  break;  }  }  scanner.close();  }} |

|  |
| --- |
| **github link:** https://github.com/jenefer-kt/InsuranceManagementSystem |