Courier Management System

Coding Task 5

Object Oriented Programming

Scope : Entity classes/Models/POJO, Abstraction/Encapsulation

Create the following model/entity classes within package entities with variables declared private, constructors(default and parametrized,getters,setters and toString())

1. User Class:

Variables: userID , userName , email , password , contactNumber , address

2. Courier Class

Variables: courierID , senderName , senderAddress , receiverName , receiverAddress , weight , status, trackingNumber , deliveryDate ,userId

3. Employee Class:

Variables : employeeID , employeeName , email , contactNumber , role String, salary

4. Location Class

Variables : LocationID , LocationName , Address

5. CourierCompany Class

Variables : companyName , courierDetails -collection of Courier Objects, employeeDetails-collection of Employee Objects, locationDetails - collection of Location Objects.

6. Payment Class:

Variables: PaymentID long, CourierID long, Amount double, PaymentDate Date

---

This task focuses on implementing core Object-Oriented Programming (OOP) principles such as encapsulation and abstraction through entity/model classes (POJO - Plain Old Java Object). Each class represents a real-world entity with private variables, constructors, getters, setters, and a toString() method.

The following classes will be created under the entities package:

1. **User Class** - Represents users interacting with the courier system.
2. **Courier Class** - Represents the details of a courier package.
3. **Employee Class** - Represents employees managing courier operations.
4. **Location Class** - Represents locations where couriers are processed.
5. **CourierCompany Class** - Manages couriers, employees, and locations.
6. **Payment Class** - Manages payment transactions related to courier services

---

Task 5.1 - User Class

The **User** class represents a registered user in the system. It contains personal details like userID, userName, email, password, contactNumber, and address. The class includes both default and parameterized constructors for creating user instances. Encapsulation is maintained by making variables private and providing public getter and setter methods for controlled access.

```java
package entity;

public class User {
    private int userID;  // Change userID type to long
    private String userName;
    private String email;
    private String password;
    private String contactNumber;
    private String address;

    // Default constructor
    public User() {}

    // Parametrized constructor
    public User(int userID, String userName, String email, String password, String contactNumber, String address) {
        this.userID = userID;
        this.userName = userName;
        this.email = email;
        this.password = password;
        this.contactNumber = contactNumber;
        this.address = address;
    }

    // Getter and Setter methods
    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 getEmail() { return email; }

    public void setEmail(String email) { this.email = email; }

    public String getPassword() { return password; }

    public void setPassword(String password) { this.password = password; }

    public String getContactNumber() { return contactNumber; }

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

    public String getAddress() { return address; }

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

    // toString method to display user details
    @Override
    public String toString() {
        return "userID : " + userID +
                ", userName : " + userName + '\'' +
                ", email : " + email + '\'' +
                ", password : " + password + '\'' +
                ", contactNumber :" + contactNumber + '\'' +
                ", address : " + address ;
    }
}
```

Task 5.2 - Courier Class

The **Courier** class models a package that is being sent through the courier system. It includes details like courierID, senderName, senderAddress, receiverName, receiverAddress, weight, status, trackingNumber, deliveryDate, and userId. This class helps in tracking shipments and linking them to users. It ensures data security by keeping fields private and providing controlled access through getters and setters.

```java
package entity;

import java.util.*;

public class Courier {
    private long courierID;
    private String senderName;
    private String senderAddress;
    private String receiverName;
    private String receiverAddress;
    private double weight;
    private String status;
    private String trackingNumber;
    private Date deliveryDate;
    private int userId;
    private int employeeId;
    private int locationId;
    private String courierCompany;
    private static int trackingNumberCounter = 1000;


    public Courier(String senderName, String senderAddress, String receiverName, String receiverAddress,
                   double weight, int userId) {
        this.courierID = trackingNumberCounter++;
        this.senderName = senderName;
        this.senderAddress = senderAddress;
        this.receiverName = receiverName;
        this.receiverAddress = receiverAddress;
        this.weight = weight;
        this.status = "Yet to Transit";
        this.trackingNumber = generateTrackingNumber();
        this.deliveryDate = null;
        this.userId = userId;
        this.employeeId = -1;
        this.locationId = -1;
        this.courierCompany = "";
    }

    // Getters and Setters
    public long getCourierID() { return courierID; }
    public void setCourierID(int courierID) { this.courierID = courierID; }
    public String getSenderName() { return senderName; }
    public void setSenderName(String senderName) { this.senderName = senderName; }
    public String getSenderAddress() { return senderAddress; }
    public void setSenderAddress(String senderAddress) { this.senderAddress = senderAddress; }
    public String getReceiverName() { return receiverName; }
    public void setReceiverName(String receiverName) { this.receiverName = receiverName; }
    public String getReceiverAddress() { return receiverAddress; }
    public void setReceiverAddress(String receiverAddress) { this.receiverAddress = receiverAddress; }
    public double getWeight() { return weight; }
    public void setWeight(double weight) { this.weight = weight; }
    public String getStatus() { return status; }
    public void setStatus(String status) { this.status = status; }
    public String getTrackingNumber() { return trackingNumber; }
    public Date getDeliveryDate() { return deliveryDate; }
    public void setDeliveryDate(Date deliveryDate) { this.deliveryDate = deliveryDate; }
    public long getUserId() { return userId; }
    public void setUserId(int userId) { this.userId = userId; }
```

```java
    public int getEmployeeId() { return employeeId; }
    public void setEmployeeId(int employeeId) { this.employeeId = employeeId; }

    public int getLocationId() { return locationId; }
    public void setLocationId(int locationId) { this.locationId = locationId; }

    public String getCourierCompanyId() { return courierCompany; }
    public void setCourierCompanyId(String courierCompanyId) { this.courierCompany = courierCompanyId; }

    @Override
    public String toString() {
        return "courierID=" + courierID +
                ", senderName='" + senderName +
                "', receiverName='" + receiverName +
                "', status='" + status +
                "', trackingNumber='" + trackingNumber +
                "', employeeId=" + (employeeId == -1 ? "Not Assigned" : employeeId) +
                ", locationId=" + (locationId == -1 ? "Not Assigned" : locationId) +
                ", courierCompany=" + (courierCompany == "" ? "Not Assigned" : courierCompany);
    }

    private String generateTrackingNumber() { return "TN" + this.courierID; }
}
```

Task 5.3 - Employee Class

The **Employee** class represents the employees working in the courier company. It includes attributes such as employeeID, employeeName, email, contactNumber, role, and salary. This class helps manage employee records and allows easy retrieval of details. By keeping fields private, it ensures data integrity and follows encapsulation principles.

```java
package entity;

public class Employee {
    private int employeeID;
    private String employeeName;
    private String email;
    private String contactNumber;
    private String role;
    private double salary;

    public Employee(int employeeID, String employeeName, String email, String contactNumber, String role, double salary) {
        this.employeeID = employeeID;
        this.employeeName = employeeName;
        this.email = email;
        this.contactNumber = contactNumber;
        this.role = role;
        this.salary = salary;
    }

    // Getters and Setters
    public int getEmployeeID() { return employeeID; }
    public void setEmployeeID(int employeeID) { this.employeeID = employeeID; }
    public String getEmployeeName() { return employeeName; }
    public void setEmployeeName(String employeeName) { this.employeeName = employeeName; }
    public String getEmail() { return email; }
    public void setEmail(String email) { this.email = email; }
    public String getContactNumber() { return contactNumber; }
    public void setContactNumber(String contactNumber) { this.contactNumber = contactNumber; }
    public String getRole() { return role; }
    public void setRole(String role) { this.role = role; }
    public double getSalary() { return salary; }
```

```
    public double getSalary() { return salary; }
    public void setSalary(double salary) { this.salary = salary; }

    @Override
    public String toString() {
        return "Employee ID: " + employeeID + ", Name: " + employeeName + ", Email: " + email + ", Role: " + role + ", Salary: " +
    }
}
```

Task 5.4 - Location Class

The **Location** class represents different locations where the courier service operates. It contains attributes like locationID, locationName, and address. This class helps in organizing branch or office locations efficiently. Encapsulation is used to maintain data security while allowing controlled access through methods.

```
package entity;

public class Location {
    private int locationID;
    private String locationName;
    private String address;

    public Location(int locationID, String locationName, String address) {
        this.locationID = locationID;
        this.locationName = locationName;
        this.address = address;
    }

    public int getLocationID() { return locationID; }
    public void setLocationID(int locationID) { this.locationID = locationID; }
    public String getLocationName() { return locationName; }
    public void setLocationName(String locationName) { this.locationName = locationName; }
    public String getAddress() { return address; }
    public void setAddress(String address) { this.address = address; }

    @Override
    public String toString() {
        return "Location ID: " + locationID + ", Name: " + locationName + ", Address: " + address;
    }
}
```

Task 5.5 - CourierCompany Class

The **CourierCompany** class acts as a central entity that manages multiple couriers, employees, and locations. It includes attributes like companyName, courierDetails (a collection of Courier objects), employeeDetails (a collection of Employee objects), and locationDetails (a collection of Location objects). This class demonstrates **abstraction** by handling multiple entities while hiding the implementation details.

```
package entity;

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

public class CourierCompany {
    private String companyName;
    private List<Courier> courierDetails;
    private List<Employee> employeeDetails;
    private List<Location> locationDetails;
```

```
    public CourierCompany(String companyName) {
        this.companyName = companyName;
        this.courierDetails = new ArrayList<>();
        this.employeeDetails = new ArrayList<>();
        this.locationDetails = new ArrayList<>();
    }

    public String getCompanyName() { return companyName; }
    public void setCompanyName(String companyName) { this.companyName = companyName; }
    public List<Courier> getCourierDetails() { return courierDetails; }
    public List<Employee> getEmployeeDetails() { return employeeDetails; }
    public List<Location> getLocationDetails() { return locationDetails; }

    public void addCourier(Courier courier) { courierDetails.add(courier); }
    public void addEmployee(Employee employee) { employeeDetails.add(employee); }
    public void addLocation(Location location) { locationDetails.add(location); }

    @Override
    public String toString() {
        return "Company: " + companyName + ", Employees: " + employeeDetails.size() + ", Locations: " +
                locationDetails.size() + ", Couriers: " + courierDetails.size();
    }
}
```

Task 5.6 - Payment Class

The **Payment** class handles payment transactions related to courier services. It contains attributes such as paymentID, courierID, amount, and paymentDate. This class ensures secure transaction management within the courier system, following encapsulation by keeping payment details private and providing controlled access methods.

```
package entity;

import java.util.Date;

public class Payment {
    private long paymentID;
    private long courierID;
    private double amount;
    private Date paymentDate;

    public Payment(long paymentID, long courierID, double amount, Date paymentDate) {
        this.paymentID = paymentID;
        this.courierID = courierID;
        this.amount = amount;
        this.paymentDate = paymentDate;
    }
```

```java
    public long getPaymentID() { return paymentID; }
    public void setPaymentID(long paymentID) { this.paymentID = paymentID; }
    public long getCourierID() { return courierID; }
    public void setCourierID(long courierID) { this.courierID = courierID; }
    public double getAmount() { return amount; }
    public void setAmount(double amount) { this.amount = amount; }
    public Date getPaymentDate() { return paymentDate; }
    public void setPaymentDate(Date paymentDate) { this.paymentDate = paymentDate; }

    @Override
    public String toString() {
        return "Payment ID: " + paymentID + ", Courier ID: " + courierID + ", Amount: " + amount + ", Date: " + paymentDate;
    }
}
```

Conclusion:

In this task, we created entity/model classes for a courier management system, demonstrating fundamental OOP concepts such as encapsulation and abstraction. These classes will serve as the foundation for further system development, enabling structured and maintainable code.
