Q 1 wap to demonstarte one to one assocation using class car and Engine

class Engine {

private String model;

public Engine(String model) {

this.model = model;

}

public String getModel() {

return model;

}

}

class Car {

private String make;

private Engine engine;

public Car(String make, Engine engine) {

this.make = make;

this.engine = engine;

}

public String getMake() {

return make;

}

public Engine getEngine() {

return engine;

}

}

public class OneToOneDemo {

public static void main(String[] args) {

Engine e1 = new Engine("V8");

Car c1 = new Car("Ford Mustang", e1);

System.out.println("Car Make: " + c1.getMake());

System.out.println("Engine Model: " + c1.getEngine().getModel());

}

}

Q 2 write a program to demonstrate one to many association using Employee class and Department class

hint class Employee { .....}

class Department {

List<Employee> l;

.....

import java.util.ArrayList;

import java.util.List;

class Employee {

private String name;

public Employee(String name) {

this.name = name;

}

public String getName() {

return name;

}

}

class Department {

private String name;

private List<Employee> employees;

public Department(String name) {

this.name = name;

this.employees = new ArrayList<>();

}

public String getName() {

return name;

}

public void addEmployee(Employee employee) {

employees.add(employee);

}

public List<Employee> getEmployees() {

return employees;

}

}

public class OneToManyDemo {

public static void main(String[] args) {

Department dept = new Department("IT");

dept.addEmployee(new Employee("John Doe"));

dept.addEmployee(new Employee("Jane Smith"));

System.out.println("Department: " + dept.getName());

for (Employee employee : dept.getEmployees()) {

System.out.println("Employee: " + employee.getName());

}

}

}

Q 3 write a program to demonstrate many to many association using Student and course class

class student

{

List<Course> l;

...}

class Course

{

List< Student> s;

....

}

import java.util.ArrayList;

import java.util.List;

class Student {

private String name;

private List<Course> courses;

public Student(String name) {

this.name = name;

this.courses = new ArrayList<>();

}

public String getName() {

return name;

}

public void addCourse(Course course) {

courses.add(course);

}

public List<Course> getCourses() {

return courses;

}

}

class Course {

private String name;

private List<Student> students;

public Course(String name) {

this.name = name;

this.students = new ArrayList<>();

}

public String getName() {

return name;

}

public void addStudent(Student student) {

students.add(student);

}

public List<Student> getStudents() {

return students;

}

}

public class ManyToManyDemo {

public static void main(String[] args) {

Student s1 = new Student("Alice");

Student s2 = new Student("Bob");

Course c1 = new Course("Math");

Course c2 = new Course("Science");

s1.addCourse(c1);

s1.addCourse(c2);

s2.addCourse(c1);

c1.addStudent(s1);

c1.addStudent(s2);

c2.addStudent(s1);

// Print student courses

System.out.println("Alice's Courses:");

for (Course course : s1.getCourses()) {

System.out.println("- " + course.getName());

}

// Print course students

System.out.println("Math Students:");

for (Student student : c1.getStudents()) {

System.out.println("- " + student.getName());

}

}

}