VIT - Vellore

Name: RONIT MEXSON .
Email: ronit.mexson2024@vitstudent.ac.in
Roll no: 24BAI0036
Phone: 9999999999
Branch: ARUMUGA ARUN R_OOPS
Department: admin
Batch: VL2024250502365
Degree: admin

Scan to verify results

BCSE102P_Structured and Object Oriented Programming Lab_VL2024250502365

VIT V_Structured and OOP_Lab 6_COD_Hard_Multi-level inheritance

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Harry is designing a program for vehicle parameter calculation of his friend Emily's car.

Help him write the program to calculate the maximum range, payload capacity and fully loaded weight of Emily's car.

The program consists of three classes that exhibit multi-level inheritance:

class Vehicle - contains fuel efficiency and maximum capacity as attributes.class Car - derived from Vehicle class and calculates the maximum range for Emily's car.class EmilyCar - derived from Car class and computes the payload capacity and fully loaded weight of Emily's car.

Formulas

Maximum range = Fuel efficiency * Maximum capacityPayload capacity = 80% of the Maximum capacityTotal fully loaded weight = Payload capacity + (Fuel efficiency * Maximum capacity)

Answer

```
// You are using GCC
#include<iostream>
#include<iomanip>
using namespace std;
class Vehicle{
   protected:
   double fuelEfficiency,max_capacity;
   Vehicle(double fuelEfficiency, double max_capacity){
      this -> fuelEfficiency = fuelEfficiency;
      this -> max_capacity = max_capacity;
   }
};
class Car : public Vehicle{
   public:
   Car(double fuelEfficiency, double max_capacity) :
Vehicle(fuelEfficiency,max_capacity){}
   double calculateMaxrange() {
      return fuelEfficiency * max_capacity;
   }
};
class EmilyCar : public Car{
   public:
   EmilyCar(double fuelEfficiency, double max_capacity) :
Car(fuelEfficiency,max_capacity) {}
   double calculatePayloadcapacity(){
      return 0.8 * max_capacity;
   }
   double calculateFullyLoadedWeight(){
      return calculatePayloadcapacity() + (fuelEfficiency * max_capacity);
   }
};

int main(){
   double fuelEfficiency,max_capacity;
   cin >> fuelEfficiency >> max_capacity;
```

```
    EmilyCar emilyCar(fuelEfficiency,max_capacity);

    cout << fixed << setprecision(2);
    cout << "Maximum Range: "<<emilyCar.calculateMaxrange() << " km" << endl;
    cout << "Payload Capacity: "<<emilyCar.calculatePayloadcapacity()<<" kg"<<
endl;
    cout << "Fully loaded Weight: "<<emilyCar.calculateFullyLoadedWeight()<<"
kg"<< endl;
    return 0;
}
```

Status : Correct **Marks : 10/10**