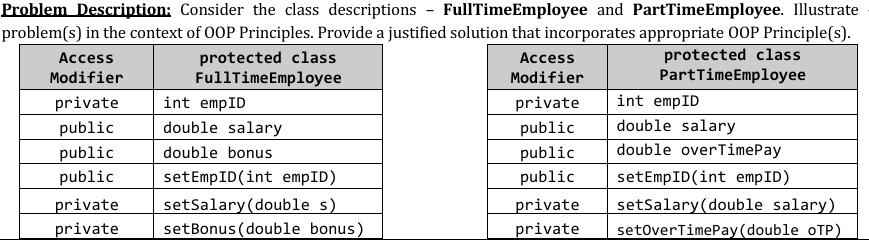
1. Complete the **Vehicle** and **Car** classes while maintaining the required OOP principles. After that, write a Main class, create **two sample objects of the Car** class inside the main method, and demonstrate the use of all the constructors and methods.

**Class Details:**

|  |  |
| --- | --- |
| **Vehicle** | **Car** |
| int vehicleId  String vehicleType  double price  int availableUnits | String brand  int warrantyYears |
| Vehicle()  Vehicle(int vehicleId, String vehicleType, double price, int availableUnits) | Car()  Car(int vehicleId, String vehicleType, double price, int availableUnits, String brand, int warrantyYears) |
| void setVehicleId(int id)  int getVehicleId()  void setVehicleType(String type)  String getVehicleType()  void setPrice(double price)  double getPrice()  void setAvailableUnits(int units)  int getAvailableUnits()  void displayVehicle() | void setBrand(String brand)  String getBrand()  void setWarrantyYears(int years)  int getWarrantyYears()  void totalPrice(int purchaseQuantity)  void totalPriceWithDiscount(int purchaseQuantity)  void displayCar() |

|  |
| --- |
|  |

1. Write the Vehicle and Car classes based on the above requirements. In the Main class, create two sample objects of the Car class, and perform the following operations. **If one buys 3 or more than 3 cars then he will get a 20% discount.** Display all details of the objects.
2. Calculate the total price of buying 2 units of the first Car object (no discount).
3. Calculate the total price of buying 4 units of the second Car object (with discount).

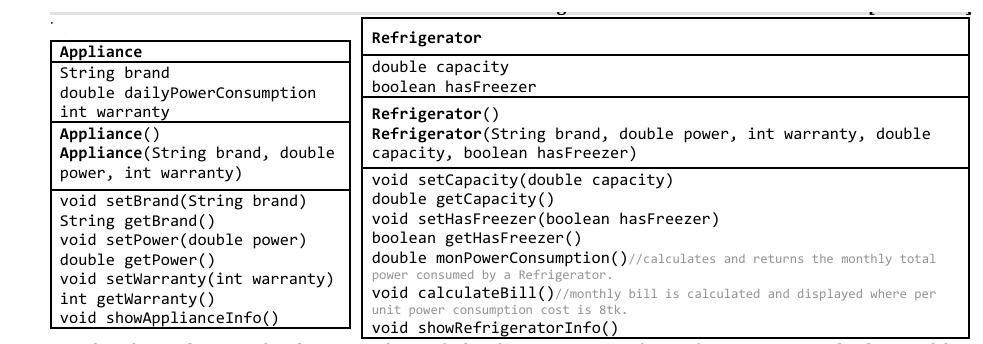
2. 

3.Look at the following class notation carefully and develop the class.

|  |
| --- |
| **Food** |
| String name  String ingredients []  double price  float calories |
| Food()  Food (String name, String[] ingredients, double price, float calories)  void setName(String name)  void setIngredients (String[] ingredients)  void setPrice(double price)  void setCalories (float calories)  String getName ( )  String[] getIngredients ( )  double getPrice ( )  float getCalories ( )  void removeIngredient(String ingredient)  void addingredient(String ingredient)  abstract void showDetails( ) |

|  |
| --- |
| **Start** |
| The ***Start*** class contains the main method. Inside the main method create two objects of the ***food*** class and demonstrate all the constructors and methods. |

4.

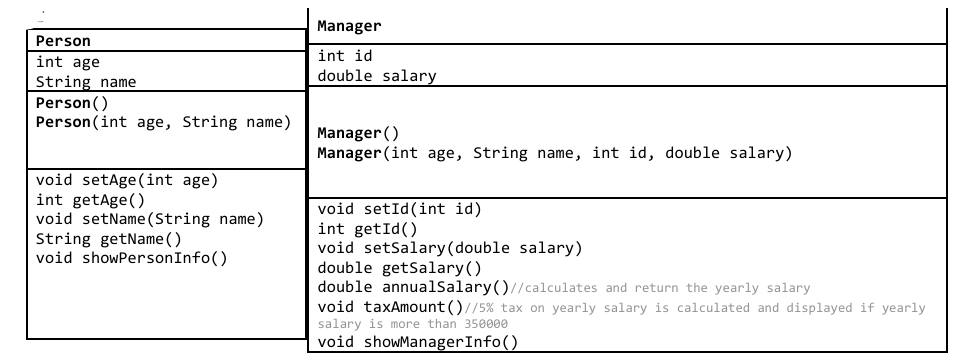


5. Look at the following class notations carefully and develop the program.

|  |
| --- |
| **Drinks** |
| String Type  int sizeInLitter |
| Drinks()  Drinks (String name, String[] ingredients, double price, float calories, String type, int sizeInLitter)  void setType (String type)  void setSize(int sizeInLitter)  String getType ()  int getSize ()  void removeIngredient(String ingredient)  void addingredient(String ingredient)  void showDetails( ) |

|  |
| --- |
| **Burger** |
| int numberOfPatty  static String addOn =”...”; |
| Burger()  Burger( String name, String[] ingredients, double price, float calories, int numberOfPatty)  void setNumberOfPatty (int numberOfPatty)  int getNumberOfPatty ()  void removeIngredient(String ingredient)  void addingredient(String ingredient)  static void showDetails () |

6.



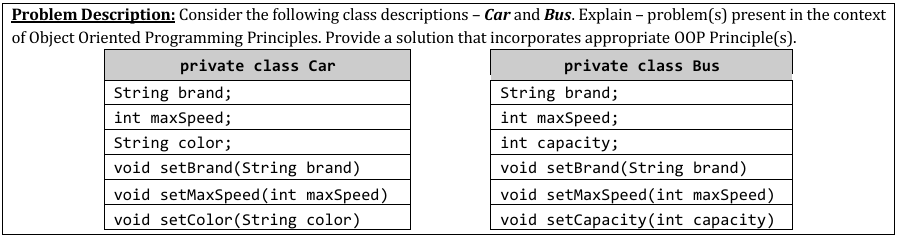
7.

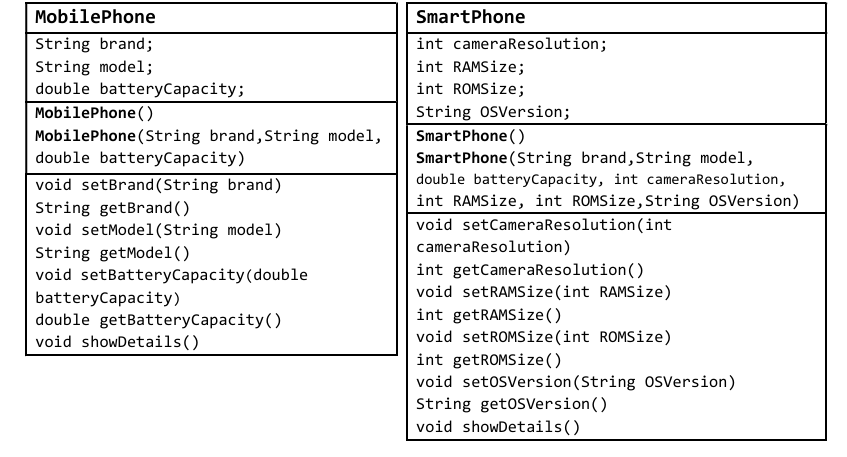
|  |
| --- |
| **Food** |
| String name  String ingredients []  double price  float calories |
| Food()  Food (String name, String[] ingredients, double price, float calories)  void setName(String name)  void setIngredients (String[] ingredients)  void setPrice(double price)  void setCalories (float calories)  String getName ( )  String[] getIngredients ( )  double getPrice ( )  float getCalories ( )  void removeIngredient(String ingredient)  void addingredient(String ingredient)  abstract void showDetails( ) |

Look at the following class notation carefully and develop the class.

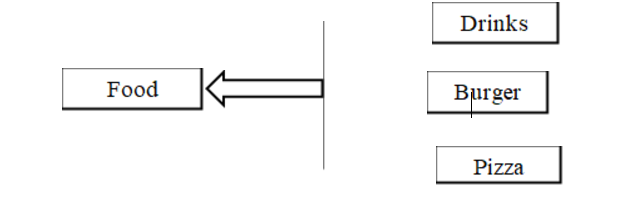
|  |
| --- |
| **Start** |
| The ***Start*** class contains the main method. Inside the main method create five objects of the ***food*** class. Create an array of food classes. Assign objects of the ***food*** class in the array and print the values of the objects from the array. |

8.





9.Look at the following class diagram and class notations carefully, and develop the program

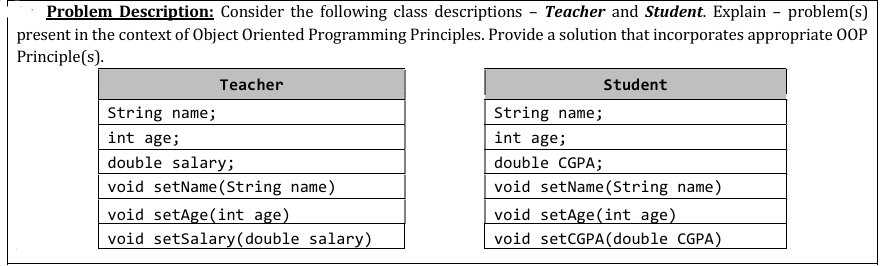


**Class Diagram**

|  |
| --- |
| **Food** |
| String name  String ingredients []  double price  float calories |
| Food()  Food (String name, String[] ingredients, double price, float calories)  void setName(String name)  void setIngredients (String[] ingredients)  void setPrice(double price)  void setCalories (float calories)  String getName ( )  String[] getIngredients ( )  double getPrice ( )  float getCalories ( )  void removeIngredient(String ingredient)  void addingredient(String ingredient)  abstract void showDetails( ) |

|  |
| --- |
| **Drinks** |
| String Type  int sizeInLitter |
| Drinks()  Drinks (String name, String[] ingredients, double price, float calories, String type, int sizeInLitter)  void setType (String type)  void setSize(int sizeInLitter)  String getType ()  int getSize () |

10 .



11.

Consider the following classes Employee and Start. Explain – problem(s) present in the context of Object-Oriented Programming Principles. Provide a solution that incorporates appropriate OOP Principle(s).

private class Employee{

private String empID;

private double empSalary;

public Employee() { }

public void setEmpID(String empID) {

this.empID = empID; }

public void setEmpSalary(double empSalary) {

this.empSalary = empSalary;}

public void showDetails() {

System.out.println(“ID: ”+empID);

System.out.println(“Salary: ”+empSalary); }

}

public class Start {

public static void main(String[] args){

Employee e1 = new Employee();

e1.setEmpID(“E-01”);

e1.setEmpSalary(-500);

e1.showDetails(); }}

12.

