**ToyDemo program:**

**package** com;

//toy demo class

**public** **class** ToyDemo {

String getLeastPrice(Toy t, Toy t1, Toy t2, Toy t3, String category) {

**if** (category.equals(t.getCategory()) || category.equals(t1.getCategory()) || category.equals(t2.getCategory())

|| category.equals(t3.getCategory())) {

**if** ((category.equals(t.getCategory())) && (category.equals(t1.getCategory()))) {

**double** r = t.getPrice() - (t.getPrice() \* (t.getDiscount() / 100));

**double** r1 = t1.getPrice() - (t1.getPrice() \* (t1.getDiscount() / 100));

**if** (r > r1) {

**return** t1.getName();

} **else**

**return** t.getName();

}

**else** **if** ((category.equals(t2.getCategory())) && (category.equals(t3.getCategory()))) {

**double** r = t2.getPrice() - (t2.getPrice() \* (t2.getDiscount() / 100));

**double** r1 = t3.getPrice() - (t3.getPrice() \* (t3.getDiscount() / 100));

**if** (r > r1) {

**return** t3.getName();

} **else**

**return** t2.getName();

} **else**

**return** " ";

} **else**

**return** "no category found";

}

//main method

**public** **static** **void** main(String[] args) {

Toy t = **new** Toy("Apple", "fruits", 100.78, 10);

Toy t1 = **new** Toy("Grapes", "fruits", 90.67, 5);

Toy t2 = **new** Toy("Lion", "Animal", 60.56, 6);

Toy t3 = **new** Toy("Tiger", "Animal", 70.89, 10);

ToyDemo td = **new** ToyDemo();

System.***out***.println(td.getLeastPrice(t, t1, t2, t3, "Animal"));

}

}

//Toy class

**class** Toy {

**private** String name;

**private** String category;

**private** **double** price;

**private** **double** discount;

Toy(String name, String category, **double** price, **double** discount) {

**this**.name = name;

**this**.category = category;

**this**.price = price;

**this**.discount = discount;

}

**public** **double** getPrice() {

**return** price;

}

**public** **void** setPrice(**double** price) {

**this**.price = price;

}

**public** **double** getDiscount() {

**return** discount;

}

**public** **void** setDiscount(**double** discount) {

**this**.discount = discount;

}

**public** String getName() {

**return** name;

}

**public** String getCategory() {

**return** category;

}

}

**Output:**

Lion

**CarDemo program:**

**package** com;

//cardemo class

**public** **class** CarDemo {

//main method

**public** **static** **void** main(String[] args) {

**char** p, c;

Car c1 = **new** Car("Hyundai", "elantra", 4, 400000);

Car c2 = **new** Car("Tayota", "fortuner", 9, 13000);

Car c3 = **new** Car("maruti", "800ac", 5, 8000);

Car c4 = **new** Car("Suzuki", "dzire", 6, 5000000);

System.***out***.println(*bestCar*(c1, c2, c3, c4, 'p'));

}

**private** **static** String bestCar(Car c1, Car c2, Car c3, Car c4, **char** c) {

String res = **null**;

**if** (c == 'p') {

**if** (c1.getOnRoadPrice() > c2.getOnRoadPrice() && c1.getOnRoadPrice() > c3.getOnRoadPrice()

&& c1.getOnRoadPrice() > c4.getOnRoadPrice()) {

res = c1.make + " " + c1.model;

} **else** **if** (c2.getOnRoadPrice() > c1.getOnRoadPrice() && c2.getOnRoadPrice() > c3.getOnRoadPrice()

&& c2.getOnRoadPrice() > c4.getOnRoadPrice()) {

res = c2.make + " " + c2.model;

} **else** **if** (c3.getOnRoadPrice() > c1.getOnRoadPrice() && c3.getOnRoadPrice() > c2.getOnRoadPrice()

&& c3.getOnRoadPrice() > c4.getOnRoadPrice()) {

res = c3.make + " " + c3.model;

} **else** **if** (c4.getOnRoadPrice() > c1.getOnRoadPrice() && c4.getOnRoadPrice() > c2.getOnRoadPrice()

&& c4.getOnRoadPrice() > c3.getOnRoadPrice()) {

res = c4.make + " " + c4.model;

}

}

**else** **if** (c == 'c') {

**if** (c1.getpassengerCapacity() > c2.getpassengerCapacity()

&& c1.getpassengerCapacity() > c3.getpassengerCapacity()

&& c1.getpassengerCapacity() > c4.getpassengerCapacity()) {

res = c1.make + " " + c1.model;

} **else** **if** (c2.getpassengerCapacity() > c1.getpassengerCapacity()

&& c2.getpassengerCapacity() > c3.getpassengerCapacity()

&& c2.getpassengerCapacity() > c4.getpassengerCapacity()) {

res = c2.make + " " + c2.model;

} **else** **if** (c3.getpassengerCapacity() > c1.getpassengerCapacity()

&& c3.getpassengerCapacity() > c2.getpassengerCapacity()

&& c3.getpassengerCapacity() > c4.getpassengerCapacity()) {

res = c3.make + " " + c3.model;

} **else** **if** (c4.getpassengerCapacity() > c1.getpassengerCapacity()

&& c4.getpassengerCapacity() > c2.getpassengerCapacity()

&& c4.getpassengerCapacity() > c3.getpassengerCapacity()) {

res = c4.make + " " + c4.model;

}

}

**return** res;

}

}

//car class

**class** Car {

String make;

String model;

**int** passengerCapacity;

**double** onRoadPrice;

Car(String make, String model, **int** passengerCapacity, **double** onRoadPrice) {

**this**.make = make;

**this**.model = model;

**this**.passengerCapacity = passengerCapacity;

**this**.onRoadPrice = onRoadPrice;

}

**int** getpassengerCapacity() {

**return** passengerCapacity;

}

**public** **void** setpassengerCapacity(**int** passengerCapacity) {

**this**.passengerCapacity = passengerCapacity;

}

**double** getOnRoadPrice() {

**return** onRoadPrice;

}

**public** **void** setonRoadPrice(**double** onRoadPrice) {

**this**.onRoadPrice = onRoadPrice;

}

**public** String getmake() {

**return** make;

}

**public** String getmodel() {

**return** model;

}

}

**Output:**

Suzuki dzire

Credit Card program:

**package** com;

**public** **class** CreditCardDemo {

//Display function

**void** display(Customer c, CreditCardCompany c1) {

System.***out***.println("customer id =" + c.getCustId());

System.***out***.println("customer account id =" + c.getAccId());

System.***out***.println("customer credit card charges =" + c.getCreditCardCharges());

System.***out***.println("customer pay back amount=" + c1.getPaybackAmount(c));

}

**public** **static** **void** main(String[] args) {

CreditCardDemo c = **new** CreditCardDemo();

CreditCardCompany cc = **new** CreditCardCompany();

Customer c1 = **new** Customer(1, 1, 400);

Customer c2 = **new** Customer(2, 2, 1400);

Customer c3 = **new** Customer(3, 3, 1800);

Customer c4 = **new** Customer(4, 4, 3000);

c.display(c1, cc);

c.display(c2, cc);

c.display(c3, cc);

c.display(c4, cc);

}

}

**class** Customer {

**private** **int** custId;

**private** **int** accId;

**double** creditCardCharges;

//Using Constructor

**public** Customer(**int** c, **int** a, **int** cc) {

custId = c;

accId = a;

creditCardCharges = cc;

}

**public** **double** getCreditCardCharges() {

**return** creditCardCharges;

}

**public** **void** setCreditCardCharges(**double** creditCardCharges) {

**this**.creditCardCharges = creditCardCharges;

}

**public** **int** getAccId() {

**return** accId;

}

**public** **int** getCustId() {

**return** custId;

}

}

**class** CreditCardCompany {

//getPaybackAmount function definition

**public** **double** getPaybackAmount(Customer c) {

**double** payBack = 0;

**double** pay = c.creditCardCharges;

**if** (pay > 2500) {

payBack = payBack + (500 \* (0.01) \* 0.25);

pay = pay - 500;

payBack = payBack + (1000 \* (0.01) \* 0.50);

pay = pay - 1000;

payBack = payBack + (1000 \* (0.01) \* 0.75);

pay = pay - 1000;

payBack = payBack + (pay \* (0.01) \* 1);

} **else** **if** (pay <= 2500 && pay > 1500) {

payBack = payBack + (500 \* (0.01) \* 0.25);

pay = pay - 500;

payBack = payBack + (1000 \* (0.01) \* 0.50);

pay = pay - 1000;

payBack = payBack + (pay \* (0.01) \* 0.75);

} **else** **if** (pay <= 1500 && pay > 500) {

payBack = payBack + (500 \* (0.01) \* 0.25);

pay = pay - 500;

payBack = payBack + (pay \* 0.01 \* 0.50);

} **else** {

payBack = (pay \* 0.01 \* 0.25);

}

**return** payBack;

}

}

**Output:**

customer id =1

customer account id =1

customer credit card charges =400.0

customer pay back amount=1.0

customer id =2

customer account id =2

customer credit card charges =1400.0

customer pay back amount=5.75

customer id =3

customer account id =3

customer credit card charges =1800.0

customer pay back amount=8.5

customer id =4

customer account id =4

customer credit card charges =3000.0

customer pay back amount=18.75