

Bu soruda bir markete ait ürün-sepet ve market kartı kullanımını içeren basit yapı kodlanacaktır. Sepetlerde market kartı kullanımı olması durumunda bu karta puan yüklemesi yapılması beklenmektedir.

Metotlar ve görevleri şu şekilde özetlenebilir:

toString metodu, Item sınıfında tanımlanarak ürünle ilgili bilgileri String olarak döndürmesi sağlanmalıdır.

addItem metodu, parametre olarak aldığı Item nesnesini ilgili Basket'e atmalıdır. Bir sepette en fazla 2 ürün olabilmektedir.

basketInfo metodu, sepette toplam kaç ürün olduğunu ve toplam ücret tutarını konsola yazdırmalı; ayrıca, yapılan bu alışverişte eğer market kartı kullanılmışsa, bu karta sepet tutarının %10'u kadar puan yüklemesi yapmalıdır.

addPoints metodu, parametre olarak aldığı puan miktarını karta atmalıdır.

Main metot ve çıktısının şu şekilde olması beklenmektedir:

```

public static void main(String[] args) {

    MarketCard mc = new MarketCard(110110);

    Basket b1 = new Basket(101);
    Basket b2 = new Basket(102);

    Item item1 = new Item(1000011, "milk", 30);
    Item item2 = new Item(1000012, "cheese", 100);
    Item item3 = new Item(1000013, "bread", 8);

    System.out.println(item1);

    b1.addItem(item1);
    b1.addItem(item2);
    b1.addItem(item3);
    b1.setCard(mc);
    b1.basketInfo();

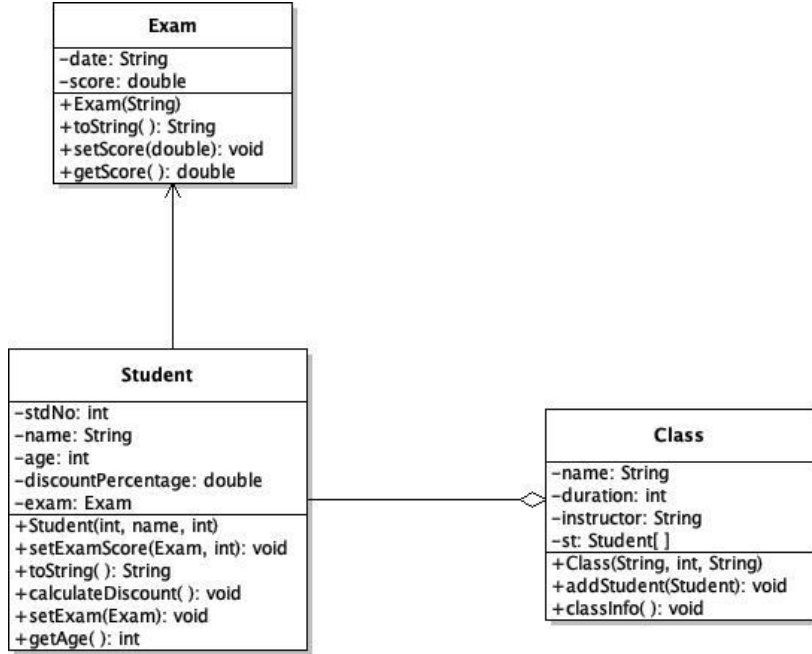
    b2.addItem(item3);
    b2.setCard(mc);
    b2.basketInfo();

    System.out.println(mc.getPoints());

}
  
```

```

<terminated> Main (48) [Java Application] /Users/elcin/.p2/pool/plugins/org.eclipse.justj
Item [barcodeNo=1000011, name=milk, price=30.0]
You've reached the maximum amount of items in your basket
Basket ID:101   Number of items:2   Total price:130.0
Basket ID:102   Number of items:1   Total price:8.0
13.8
  
```



```

public static void main(String[] args) {

    Exam ex = new Exam("01-07-2023");

    Class c1 = new Class("Painting", 3, "Mehmet Er");

    Student s1 = new Student(101, "Ali Demir", 34);
    Student s2 = new Student(102, "Veli Sur", 56);
    Student s3 = new Student(103, "Mert Yurt", 25);

    s1.setExamScore(ex, 65);

    System.out.println(s1);

    c1.addStudent(s1);
    c1.addStudent(s2);
    c1.addStudent(s3);
    c1.classInfo();

    s1.calculateDiscount();
    System.out.println(s1.getDiscountPercentage());

}
  
```

Bir kurs merkezinde açılan sınıflar ve kayıt olan öğrencileri modelleyen yapı kodlanacaktır. Öğrencilerin kayıt olmadan önce girdikleri yetenek sınavı puanları kurs kayıt ücretlerindeki indirimi belirleyecektir.

Metotlar ve görevleri şu şekilde özetlenebilir:

setExamScore metodu, bir öğrencinin girdiği sınav ve aldığı notu parametre olarak alıp gerekli alanların atamasını yapmalıdır.

toString metodu Student ve Exam sınıflarında tanımlanarak öğrenci ve girdiği sınava ait bilgileri String olarak döndürmelidir.

addStudent metodu, parametre olarak aldığı Student nesnesini ilgili Class'a atamalıdır. Bir sınıfta en fazla 2 öğrenci olabilmektedir.

classInfo metodu, sınıf adını, öğrenci sayısını ve sınıftaki öğrencilerin yaş ortalamasını konsola yazdırmalıdır.

calculateDiscount metodu, öğrencinin sınav sonucuna bakarak 70 ve üzeri almış ise %20, 50-70 arası ise %10 indirim oranı ataması yapmalıdır. Bunun dışındaki durumlar için indirim oranı hesaplanmamaktadır.

Main metot ve çıktısının şu şekilde olması beklenmektedir:

```

Problems  @ Javadoc  Declaration  Console X
<terminated> Main (49) [Java Application] /Users/elcin/.p2/pool/plugins/org.eclipse.justj.oper
Student [stdNo=101, name=Ali Demir, age=34, exam=Exam [score=65.0]]
Course capacity is full
Class name:Painting  Number of students:2  Average age:45.0
10.0
  
```