

#### GRUP-A:

Bu soruda yurt yönetim sistemi modülü kodlanacaktır. Yurt yönetimi en fazla N sayıda öğrenci için bir liste saklamaktadır. Her bir öğrencinin id, ad, soyad ve borç bilgisi bulunmaktadır. Ayrıca her öğrencinin id, ad, soyad, borç bilgileri toString() metodu ile geri döndürülmelidir. Öğrencinin borç kontrolü için öğrenci sınıfında borç kontrol metodu olmalıdır. Yurt sisteminde ID değeri verilen öğrenciyi bulmak için ogrenciBul isimli metod yazılmadadır.

**ÖĞRENCİ EKLE:** Yurt sisteminde yeni öğrenci eklenebilmelidir, öğrenci eklenirken:

Yurt kapasitesinin dolması durumunda istisna oluşmalı. **“Yurt kapasitesi doldu” mesajını yazdıracak ile ilgili istisna sınıfı** yazılmalıdır.

Aynı id'li öğrenci tekrar sisteme eklenmeye çalışıldığı durumda istisna oluşmalı ve ilgili istisna sınıfı yazılmalıdır. Herhangi bir aykırı durum oluşmıyorsa ekleme işlemi yapılmalı ve kullanıcıya mesaj verilmelidir.

**ÖĞRENCİ SİL:** Yurt sisteminde öğrenci çıkarılabilmelidir, çıkarma yaparken:

ID değeri verilen bir öğrenci yurttan ayrılabilir, ancak;

Eğer öğrencinin borcu varsa istisna oluşturulmalı ve ilgili istisna sınıfı **“Borcu Olan Öğrenci yurttan ayrılamaz” mesajını yazdırmalıdır.**

**BORÇ GÜNCELLEME İŞLEMLERİ:** Öğrencilerin borç bilgisi borçEkle ve borçOde isminde iki farklı metodla güncellenebilmelidir.

Bu işlem yurtYonetimi tarafından aşağıdaki şekillerde yapılabilir.

1- Tüm öğrencilerin borç bilgisi verilen miktar kadar arttırılır.

2- id numarası verilen bir öğrencinin borç bilgisi verilen miktar kadar azaltılabilmelidir. OgrenciBul() metodu kullanılmalıdır.

**YURT ÖĞRENCİ LİSTELE:** Yurtta bulunan tüm öğrencilere ait bilgiler aşağıdaki şekilde listelenir. Bunun için Iterator/foreach ile öğrenci sınıfındaki toString metodu kullanılmalıdır.

OGRENCİ LISTESİ

[ID: 3, AD: Ayşe, SOYAD: TEZ, BORÇ: 3000.0 ]

[ID: 5, AD: Mehmet, SOYAD: ÖZ, BORÇ: 1000.0 ]

## GRUP-B:

Bu soruda market sepeti modülü kodlanması gerekmektedir. Market sepetindeki ürünlere aşağıdaki şartlarda listeleme, adet güncelleme, ekleme ve çıkarma yapılabilmektedir. Her bir ürünün id, ad ve adet bilgisi olmalıdır. Ayrıca her ürünün id, ad ve adet bilgileri toString() metodu ile geri döndürülmelidir. Market sepeti içerisinde ürünler Arraylist ile saklanabilmelidir. Sepet içerisinde id değeri verilen bir ürünün bulunmasını sağlayan metod olmalıdır[mevcutUrunBul(id)].

Sepet üzerinde ekleme, silme, ürün adet güncelleme ve listeleme işlemleri aşağıdaki şartlara göre yapılabilmelidir.

### ÜRÜN EKLE:

Aynı ID'li ürün sepete ikinci kez eklenemez. Bu durumda **"x id ile urun kaydı var"** şeklinde bir istisna oluşmalıdır. (AynıldUrunIstisna sınıfına mesaj göndermelidir.)

Bir üründen sepete en fazla 5 tane eklenebilir, **aksi durumda istisna oluşmalı ve ürün eklenmemelidir.** (UrunAdetIstisna sınıfına mesaj göndermelidir.)

Diğer durumlarda ürün nesnesi sepet listesine eklenebilmelidir.

### ÜRÜN ÇIKAR:

ID değeri verilen bir ürün eğer listede varsa çıkarılmalı, listede yoksa ürün bulunamadı mesajı yazdırılmalıdır. Herhangi bir istisna yazmanıza gerek yoktur.

### ÜRÜN ADET GÜNCELLE

ID ve ekleme/çıkarma yapılacak adet bilgisi verilen bir ürünün adet değerinde aşağıdaki durumlara göre güncelleme yapılmalıdır.

Verilen ID değerine ve kullanıcıdan alınan +X veya -X değerine göre, adet güncelleme yapılmalıdır.

Eğer güncel adet değeri 0 veya altına inmişse bir istisna oluşur. **Urun Adeti En Az 1 Olmalı, "+id+" ID'li Urun Sepetten Çıkarıldı mesajı yazılır, ürün listeden silinir.**

Eğer güncel adet değeri 5 veya daha fazla değere ulaşmışsa, **Urun Adeti En Fazla 5 Olmalı, Urun Sayısı 5 Yapıldı mesajı verilerek ürün adet değeri 5 olarak değiştirilir.**

Diğer durumlar için; **herhangi bir istisna veya uyarı olmadan ürün adedi güncellenir.**

### SEPET LİSTELE:

Sepette bulunan tüm ürünlere ait bilgiler aşağıdaki şekilde listelenir. Bunun için Iterator/foreach ile ürün sınıfındaki toString metodu kullanılmalıdır.

### URUN LISTESİ

[ID: 1, AD: Kalem, ADET:1 ]

[ID: 5, AD: Cetvel, ADET:5 ]

## GRUP-C:

Bu soruda bellek yönetim modülü kodlanması gerekmektedir. Bellekte bulunan processlere aşağıdaki şartlarda listeleme, process işleme, process yükleme ve process çıkarma yapılabilmektedir. Her bir process için id, ad ve koşma süresi bilgisi olmalıdır. Ayrıca her processin id, ad ve adet bilgileri toString() metodu ile geri döndürülmelidir. Bellek içerisinde processler ArrayList ile saklanabilmelidir, ayrıca bellekte en fazla N sayıda process saklanabilmelidir. Bellek içerisinde id değeri verilen bir processin bulunmasını sağlayan metod olmalıdır[pocessBul(id)]. Bellek üzerinde aşağıdaki şartlara göre işlemler yapılabilmelidir.

**PROCESS YÜKLE:** Process nesnesinin belleğe eklenmesi işlemleridir.

Bellekte en fazla N sayıda process bulunabilir, process eklenirken N değerinin aşılması durumunda, **“Bellek Dolu,**

**Process Yükleneemedi”** mesajını veren **BellekDolu istisnası oluşmalıdır.**

Aynı ID’li process belleğe ikinci kez eklenemez. Bu durumda **“X ID’li process beklekte zaten var”** şeklinde bir **istisna oluşmalıdır.**

Diğer durumlarda process belleğe eklenmelidir.

**PROCESS ÇIKAR:** ID değeri verilen bir process’in belleğe yüklenmesi işlemleridir.

ID değeri verilen bir process bellekte varsa, process koşma süresine bakılacak ve bu süre değeri 0 değerindeyse(yani koşacak süresi kalmamışsa) bellekten process çıkarılmalıdır. Aksi durumda **“X ID’li process henüz tamamlanmadı”** şeklinde bir **istisna oluşmalıdır.**

Belirtilen ID ile bir process bellekte yoksa, “Process Bulunamadı” mesajı istisna oluşturulmadan yazdırılmalıdır.

**PROCESS İŞLE:**ID ve süre bilgisi verilen bir processin koşma süresi değerinde güncelleme yapılması işlemleridir.

Bellekte ID değeri verilen process varsa, verilen süre bilgisi process’in koşma süresinden çıkarılır. Bu işlem sonucunda koşma süresi 0 veya altına düşmüşse process bellekten çıkarılır, diğer durumlarda process bellekte kalmaya devam eder/etmesi gerekir.

## BELLEK YAZDIR:

Bellekte bulunan tüm processlere ait bilgiler aşağıdaki şekilde listelenir. Bunun için Iterator/foreach ile ürün sınıfındaki toString metodu kullanılmalıdır.

## PROCESS LISTESİ

```
[ID: 1, AD: Word, KOSMA SÜRESİ:1 ]  
[ID: 2, AD: Excel, KOSMA SÜRESİ:4 ]  
[ID: 4, AD: Flash, KOSMA SÜRESİ:8 ]  
[ID: 5, AD: PyCharm, KOSMA SÜRESİ:5 ]
```