

RAPOR

PART 1 :

Ders Class'ı

Ders adında bir Class oluşturuldu. Yapıcı fonksiyon içerisinde String **dersAdı** String **dersKodu** ve int tipinde **somestrNo** adında somestr numarası parametreleri tanımlandı.

Linked liste her yeni ders eklendiğinde otomatik olarak artan **ID** atanması için int **count** değeri oluşturuldu. **ID** değişkeninin **set** ve **get** metodları tanımlandı count değeri yeni nesnede bir arttırıldı.

Bir sonraki dersi tutması için **Ders** tipinde **next** değişkeni oluşturuldu.

Ders nesnesi çağrıldığında düzgün bilgi vermesi için **toString** metodu oluşturuldu.

DersListesi Linked List Class'ı

Linkin ilk ve son değerlerini tutması için **Ders** classına ait **baş** ve **son** fieldları oluşturuldu.

Metodlar:

basaEkle() **indekseEkle()** ve **sonaEkle()** metodlarıyla linkedliste eleman ekleme özelliği eklendi.

dersSilByDersKodu() metoduyla ders kodu verilen ilk dersi listeden siler.

Next() metodu parametre olarak verilen id numaralı dersi siler.

doluMu() metodu listeye eleman ekleme durumu listenin dolu veya boş olmasına göre değişiklik göstereceği için bu sorgu bir metod haline getirildi.

listeyiYazdir() metodu linkedlistdeki tüm dersleri yazdırır.

Size() metodu listede kaç eleman olduğunu gösterir.

listSemesterCourses(int) parametre olarak verilen somestr numarasına ait tüm dersleri yazdırır.

getByCode() parametre olarak verilen ders koduna ait tüm dersleri yazdırır.

nextInSomestr() ders kodu verilen ders ile aynı somestrdeki bir sonraki dersi yazdırır.

dersSilByDersID() id'si verilen dersi siler.

getByRange(int,int) parametre olarak verilen iki indeks değeri arasındaki tüm dersleri yazdırır.

PART 2 :

Listedeki bir elemanı disabled ve enabled duruma getirebilmek için DisabledList adına ikinci bir linked list classı oluşturuldu.

Ana listemizden bir ders disabled duruma geleceği zaman dersin tüm özelliklerine DisabledList sınıfına aktarılıyor.

Tekrar enable edildiğinde aynı konumuna gelebilmesi için disabled edilirken hangi indekste olduğu disable metodu içerisinde belirleniyor.

Enable edileceği zaman Part1 de yarattığımız indkxseEkle metoduyla() silindiğinde hangi indeksteyse tekrar o indekse ekliyoruz.

DisabledList classı ;

Ders adı, ders kodu, sömestr numarası, disabled edildiği konumdaki indeksini tutan lastIndeks ve ders ID'lerinden oluşan bir sınıf yaratıldı

Her sınıf üyesine otomatik olarak ID ataması yapılması için static bir sayaç eklendi.

ID için get ve set metodları oluşturuldu.

Metodlar:

listeyiYazdir() : showDisabled dediğimizde sınıfta bulunan dersleri yazdırabilmemiz için oluşturuldu

basaEkle() : bir ders disable edildiğinde DisabledList listemize eklenmesi için oluşturuldu.

dersSilByDersId() : bir ders tekrar enable edildiğinde DisabledList'ten dersi siler.

dersBul(): bir ders tekrar enable edildiğinde Disabled listen silinecek dersin id'sini bulur. dersSilByDersId() metodu içerisinde kullanılır.

DersListesi classı ;

Part1 den farklı olarak burada **disable()** **enable()** ve **showDisabled()** metodları oluşturuldu.

disable() :

1. Tekrar enable edileceğinde aynı yere gelebilmesi için bulunduğu indeksi kaydeden bir döngü oluşturuldu
2. İndeks bulunduktan sonra disabledList sınıfına eklendi
3. disabledList sınıfına eklendikten sonra mevcut listeden silindi

enable() :

1. parametre olarak verilen id yi dersBul metoduyla arar.
2. Bulursa ders sınıfına ait tüm özellikleri içeren yeni bir nesne oluşturur.
3. İndekse ekle metoduyla tekrar eski indeksine eklenir
4. disabledList listesinden silinir.

showDisabled():

1. disabledList sınıfının içerisinde oluşturduğumuz listeyiYazdir() metodunu çalıştırır.

PART 3 :

Part 1 den farklı olarak **Ders** sınıfına **nextSomestr** fieldı eklendi. Bu field kendisiyle aynı somestrda bulunan bir sonraki dersi işaret eder. Eğer son elemansa ilk elemanı işaret eder.

Yine part 1 den farklı olarak DersListesi sınıfına aynı sömestrdaki dersleri birbirine bağlayacak **linkSomestr()** metodu eklendi

linkSomestr(int somestrNo) :

1. verilen numara 1 ile 8 numaralarında mı kontrol eder
2. Doğruysa 3 tane Ders sınıfına ait işaretçi oluşturulur
 - a. **isaretcı2** : bağlanacak ilk dersi tutar
 - b. **isaretcı** : bağlanılacak sıradaki dersi tutar
 - c. **isaretcıBas** : bağlanılacak bir sonraki ders kalmadığında son isaretcıBasa bağlanır

