

Nesneye Yönelik Programlama

3.Laboratuvarı - 23.05.2021

Süre: 90 dk.

main metot içeren Test sınıfı verilmiştir.

Başarılar.

- Öğrencilere kitap ödünç verildiği bir sistem kurulacaktır.
- Main metodun içerisinde öğrenci ve kitap nesneleri oluşturulmuştur. Öğrencileri **ArrayList** içerisinde, kitapları **LinkedList** yapısında tutmanız istenmektedir.
- **MyInfoClass** isimli bir sınıf oluşturup parametre olarak herhangi bir türde nesne alan **getObjectInfo** ve parametre olarak liste alan **getListInfo** metotlarını tanımlamanız gerekmektedir. Bu metotlar, gönderilen parametrenin içeriğini String olarak yazdırmalıdır. (Student ve Book sınıflarında toString metotlarını override etmeyi unutmayınız.)
- Oluşturacağınız Library sınıfına ait **borrowBook** metodu öğrenci ve kitap listelerini parametre olarak alıp çalıştığında, var olan kitapları, ilk sıradaki öğrenciden başlayarak sırasıyla bir öğrenciye bir kitap verilecek şekilde eşleştirmelidir. Bu eşleştirme işleminde **HashMap** yapısı kullanmanız istenmektedir.
- **mapPrint** metodu, oluşturduğunuz hashmap'in içeriğini outputta gördüğünüz bilgileri içerek şekilde ekrana yazdırmalıdır.
- **returnBook** metodu, parametre olarak aldığı öğrenci kitabı getirdiğinde, bu öğrenciyi hashmap'ten silmeli ve iade edilen kitabı sırada bekleyen öğrenciye vermelidir.
- Kütüphaneye yeni kitap girişinin yapıldığı **newBook** metodu, konsoldan girilen kitap ve yazar adı bilgilerini alır ve kitapların olduğu listede aynı kitaptan olup olmadığının kontrolünü yapar. Eğer böyle bir kitap yoksa listeye ekler.
Bu kitaptan varsa "**Library has this book!**" hata mesajı içeren, sizin oluşturacağınız BookException sınıfından bir exception fırlatır. BookException sınıfı IOException'dan kalıtımla oluşturulmalıdır. Checked bir exception olması sebebiyle, main metot içerisinde newBook metodunu çağırdığınızda gerekli düzenlemeleri yapmanız gerektiğini unutmayınız (try-catch).

OUTPUTS:

```
***** info *****
Student [id=102, name=Mike Garret]
Book [name=1st Book, author=1st Author]
Student [id=101, name=Ellie Brown]
Student [id=102, name=Mike Garret]
Student [id=103, name=John Doens]
Student [id=104, name=Zoey Capp]
Student [id=105, name=Eric Palm]

***** current match *****
101 numbered student borrowed -> 1st Book
103 numbered student borrowed -> 3rd Book
102 numbered student borrowed -> 2nd Book

***** after s2 returned the book *****
101 numbered student borrowed -> 1st Book
104 numbered student borrowed -> 2nd Book
103 numbered student borrowed -> 3rd Book

***** after s1 returned the book *****
104 numbered student borrowed -> 2nd Book
105 numbered student borrowed -> 1st Book
103 numbered student borrowed -> 3rd Book

***** wrong student *****
Ellie Brown has already returned the book.

***** after s4 returned the book *****
105 numbered student borrowed -> 1st Book
103 numbered student borrowed -> 3rd Book

***** after s3 returned the book *****
105 numbered student borrowed -> 1st Book

***** after s5 returned the book *****
No students to borrow book

***** new book to library *****
Enter book name:
1st Book
Enter author name:
1st Author
lab3_2021.BookException: Library has this book!
    at lab3_2021.Library.newBook(Library.java:87)
    at lab3_2021.Test.main(Test.java:78)
```