

# DERS SONU UYGULAMA

## HAFTA #4

10 EKİM 2025 CUMA

Bir üniversite kütüphanesinde, kitapların ve üyelerin yönetimini kolaylaştırmak amacıyla bir **Kütüphane Yönetim Sistemi** geliştirilmek istenmektedir. Bu sistem, kitapların ödünç alınması, iade edilmesi, üyelerin kaydedilmesi ve kütüphane envanterinin görüntülenmesi işlemlerini gerçekleştirecektir.

Sizden, soyut sınıf (**abstract class**), arayüz (**interface**) ve sınıflar arası ilişkiler kullanarak nesne yönelimli prensiplere uygun bir **Java uygulaması geliştirmeniz** beklenmektedir. Aşağıda gereksinimler ayrıntılı olarak verilmiştir:

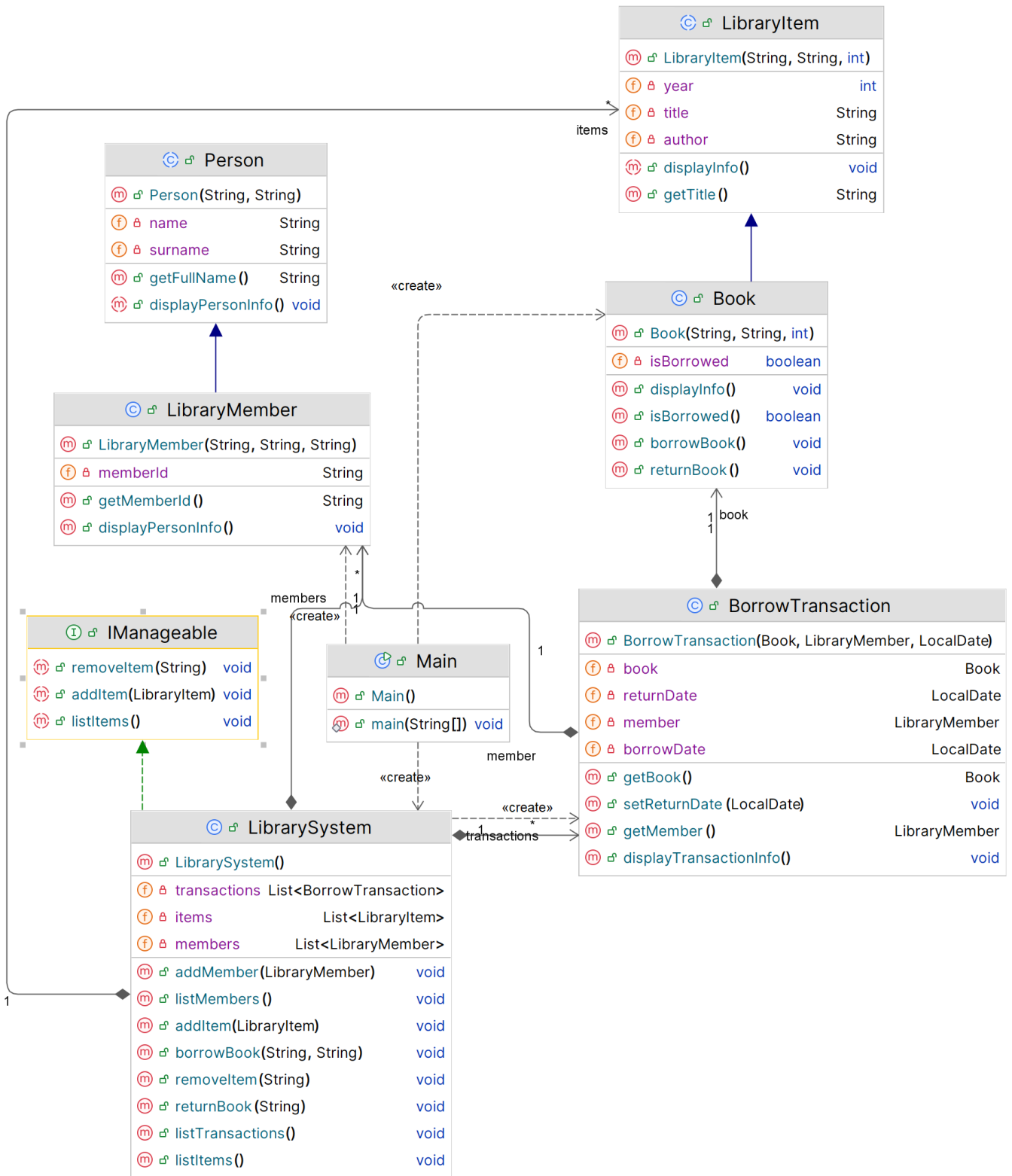
### GEREKSİNİMLER

1. **IManageable** adlı bir arayüz (**interface**) oluşturun.
  - a. Aşağıdaki metotları içermelidir:
    - i. **addItem(LibraryItem item)**
    - ii. **removeItem(String title)**
    - iii. **listItems()**
  - b. Bu arayüz, kütüphaneye yeni materyal ekleme, çıkarma ve listeleme işlemlerinin tanımını içermelidir.
2. **LibraryItem** adlı bir soyut sınıf (**abstract class**) oluşturun.
  - a. Kitap veya başka kütüphane materyallerinin ortak özelliklerini temsil eder.
  - b. Alanlar:
    - i. **title (String)**
    - ii. **author (String)**
    - iii. **year (int)**
  - c. Metotlar:
    - i. **getTitle()**
    - ii. **displayInfo()** (soyut metod)
3. **Book** sınıfını **LibraryItem** sınıfından türetin.
  - a. Ek alan: **isBorrowed (boolean)**
  - b. Metotlar:
    - i. **borrowBook()** → kitabın ödünç alındığını işaretler.
    - ii. **returnBook()** → kitabın iade edildiğini işaretler.
    - iii. **displayInfo()** → kitabın adını ve mevcut durumunu ekrana yazdırır.
4. **Person** adlı bir soyut sınıf (**abstract class**) oluşturun.
  - a. Ortak kişi bilgilerini içermelidir.
  - b. Alanlar:
    - i. **name, surname**
  - c. Metotlar:
    - i. **getFullName()**
    - ii. **displayPersonInfo()** (soyut metod)
5. **LibraryMember** adlı sınıfı **Person** sınıfından türetin.
  - a. Ek alan:
    - i. **memberId (String)**
  - b. Metotlar:
    - i. **displayPersonInfo()** → üye bilgilerini ekrana yazdırır.
6. **BorrowTransaction** adlı bir sınıf oluşturun.
  - a. Ödünç alma işlemlerini temsil eder.
  - b. Alanlar:
    - i. **book (Book)**
    - ii. **member (LibraryMember)**
    - iii. **borrowDate (LocalDate)**
    - iv. **returnDate (LocalDate)**
  - c. Metotlar:
    - i. **setReturnDate()** → kitabın iade tarihini ayarlar.
    - ii. **displayTransactionInfo()** → işlem detaylarını ekrana yazdırır.
7. **LibrarySystem** adlı bir sınıf oluşturun.

- a. **IManageable** arayüzünü uygulasin.
  - b. Kitap, üye ve işlem yönetiminden sorumludur.
  - c. Alanlar:
    - i. **List<LibraryItem> items**
    - ii. **List<LibraryMember> members**
    - iii. **List<BorrowTransaction> transactions**
  - d. Metotlar:
    - i. **addItem()** → kitap ekler.
    - ii. **removeItem()** → kitap siler.
    - iii. **listItems()** → mevcut kitapları listeler.
    - iv. **addMember()** → yeni üye ekler.
    - v. **listMembers()** → üyeleri listeler.
    - vi. **borrowBook(title, memberId)** → kitabı üyeye ödünç verir.
    - vii. **returnBook(title)** → kitabın iadesini gerçekleştirir.
    - viii. **listTransactions()** → tüm ödünç işlemlerini listeler.
8. **Main** adlı sınıfta **main()** metodu oluşturun.
- a. Uygulamanın çalıştırıldığı ana sınıf olmalıdır.
  - b. Aşağıdaki işlemleri sırasıyla gerçekleştirin:
    - i. **LibrarySystem** sınıfından bir nesne oluşturun.
    - ii. En az **üç kitap** ve **iki üye** ekleyin.
    - iii. Kitap ve üye listelerini görüntüleyin.
    - iv. Bazı kitapları ödünç verin.
    - v. En az bir kitabı iade edin.
    - vi. Tüm işlem geçmişini ekrana yazdırın.
9. Tüm sınıfların yapısını **Class diyagramı** ile gösteriniz.

#### İLİŞKİLER:

- **LibrarySystem** sınıfı **IManageable** arayüzünü uygular.
- **Book** sınıfı **LibraryItem** soyut sınıfından türetilir.
- **LibraryMember** sınıfı **Person** soyut sınıfından türetilir.
- **BorrowTransaction** sınıfı **Book** ve **LibraryMember** sınıflarıyla ilişkilidir (composition).
- **Main** sınıfı sistemi çalıştıran ana sınıftır.



"C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDE

- ✅ Yapay Zeka eklendi.
- ✅ Derin Öğrenme eklendi.
- ✅ Makine Öğrenmesi eklendi.
- 👤 Üye eklendi: Ayşe Yılmaz
- 👤 Üye eklendi: Ali Can

#### 📚 Kütüphane Envanteri:

- 📖 Kitap: Yapay Zeka | Durum: Mevcut
- 📖 Kitap: Derin Öğrenme | Durum: Mevcut
- 📖 Kitap: Makine Öğrenmesi | Durum: Mevcut

#### 👤 Üye Listesi:

- 👤 Üye: Ayşe Yılmaz | ID: M001
- 👤 Üye: Ali Can | ID: M002
- 📖 Yapay Zeka kitabı Ayşe Yılmaz tarafından ödünç alındı.
- 📖 Derin Öğrenme kitabı Ali Can tarafından ödünç alındı.

#### 📅 Ödünç İşlemleri:

- 🔪 Ödünç: Ayşe Yılmaz | Kitap: Yapay Zeka | Alış Tarihi: 2025-10-10 | Henüz iade edilmedi
- 🔪 Ödünç: Ali Can | Kitap: Derin Öğrenme | Alış Tarihi: 2025-10-10 | Henüz iade edilmedi
- ✅ Yapay Zeka iade edildi.

#### 📅 Ödünç İşlemleri:

- 🔪 Ödünç: Ayşe Yılmaz | Kitap: Yapay Zeka | Alış Tarihi: 2025-10-10 | İade Tarihi: 2025-10-10
- 🔪 Ödünç: Ali Can | Kitap: Derin Öğrenme | Alış Tarihi: 2025-10-10 | Henüz iade edilmedi

#### 📚 Kütüphane Envanteri:

- 📖 Kitap: Yapay Zeka | Durum: Mevcut
- 📖 Kitap: Derin Öğrenme | Durum: Ödünç alındı
- 📖 Kitap: Makine Öğrenmesi | Durum: Mevcut

Process finished with exit code 0