



Aşağıdaki gereksinimleri karşılayan bir **kütüphane yönetim sistemini** Java ile yazınız:

### Book Sınıfı

`isbn` ve `title` bilgilerini tutar.

`Identifiable` arayüzünü uygular; `getId()` metodu ISBN döndürür.

`Comparable<Book>` arayüzünü uygular; kitaplar **başlığa göre** sıralanır.

`toString()` metodu anlamlı çıktı vermelidir.

### ItemNotFoundException

Bir ID ile aranan nesne bulunamadığında fırlatılan özel bir Exception olmalıdır.

### Repository<T extends Identifiable>

Nesneleri ID'ye göre saklayan generic bir sınıf olmalıdır.

`add(T item)`, `findById(String id)`, `getAll()` metotlarını içerir.

`findById` metodu, nesne yoksa `ItemNotFoundException` fırlatmalıdır.

### User Sınıfı

`Identifiable` arayüzünü uygular (`id`, `name`).

Kullanıcının ödünç aldığı kitapları `ArrayList<Book>` içinde tutar.

`borrow(Book b)`, `returnBook(String isbn)` ve `printBorrowed()` metotlarına sahip olmalıdır.

### Main Sınıfı

En az 3 kitap oluşturup `Repository<Book>` içine ekleyiniz.

Bir kullanıcı oluşturup bazı kitapları ödünç aldırınız.

Kullanıcının ödünç aldığı kitapları yazdırınız.

`findById` ile bir kitap bulunuz; olmayan bir ID için istisna yakalayınız.

Tüm kitapları başlığa göre sıralayıp ekrana yazdırınız.

Kullanıcı bir kitabı iade etsin ve güncel listeyi yazdırınız.