

NYP LAB 3

Kütüphane Analitiği (Menü Tabanlı)

Senaryo

Kitapları yöneten küçük bir uygulama yaz. Verileri **diziler** (gerekirse ArrayList/Vector köprüsü) ile tut, işlemleri **sınıflar/metotlar** ile yap ve programı **switch-case menü** üzerinden kullan.

Çıktı Olarak Beklenenler

- Paketli proje (ör. org.boston.libraries)
- En az 3 sınıf: Book, Library, CirculationStats
- Konsolda menü ile çalışan uygulama (switch-case)

Sınıflar ve Sorumluluklar

1) Book

- **Alanlar:** başlık, yazar, yıl, **puanlar** (1 boyutlu int[]), (opsiyonel) etiketler.
- **Kurucular:**
 - Tam parametreli (tüm alanları alır)
 - Parametresiz (makul varsayılanlar atar)
- **Metotlar:**
 - Ortalama puanı hesapla
 - Etiket ekle/çıkar (eklediysen)
 - **İçerik eşitliği:** "puanlar" dizisini eleman-eleman karşılaştırarak iki Book nesnesinin puan içeriği eşit mi?
- **Statik yardımcılar:**
 - Güvenli kopya üret (elle kopya ve clone yaklaşımını ayrı ayrı uygulat)
 - İki puan dizisinin **İçerik eşitliğini** test eden araç

2) Library

- **Alan:** Book[] envanteri (dizi).
- **Metotlar:**
 - **Arama (overloading):**
 1. Sadece başlığa göre
 2. Başlık + yıl kombinasyonuna göre
 3. Bir koşula göre (ör. ortalama puanı eşikten büyük olanları döndür)
 - **Sıralama:** kitapları ortalama puana göre **selection sort** ile sırala
 - **Kopya/karşılaştırma yardımcıları:**
 - Derin kopya (icerikleri yeni dizeye geçir)
 - İçerik eşitliği (erken çıkış mantığıyla)

3) CirculationStats

- **Alan:** yıllara ve çeyreklerde kullanım sayıları için **2 boyutlu** dizi
 - Standart (dikdörtgen) yapı veya **ragged** (satır uzunlukları farklı) yapı desteklenebilir
- **Metotlar:**
 - Belirli yılın toplam kullanımını
 - En yoğun çeyrek
 - Yıllık özet (her yılın toplamı)

Program Giriş: Switch-Case Menü

Program sürekli dönen bir döngü içinde menüyü gösterir, kullanıcıdan seçim alır, switch ile ilgili işlemi çalıştırır. "0 – Çıkış" seçilene kadar devam eder.

Ekranda gözükecek örnek menü metni (örnek; biçimini sana ait):

- | |
|---|
| 1) Kitapları Yükle |
| 2) Kitapları Listele |
| 3) Ara (Başlık) |
| 4) Ara (Başlık + Yıl) |
| 5) Ara (Koşula Göre) |
| 6) Sırala (Ortalama Puan Artan) |
| 7) Puan Dizisi: Kopya Al (Elle / clone) |
| 8) Puan Dizisi: İçerik Eşitliği Testi |
| 9) İstatistik: Yıl/Çeyrek Toplamları (2D) |
| 10) ArrayList'e Aktar / Geri Dön |
| 11) Vector Kapasite Gözlemi |
| 0) Çıkış |

Menü maddelerinde beklenen davranış

- **1) Kitapları Yükle:**
 - Dizi oluştur ve kitap nesnelerini yükle (ister sabit örnek veri, ister dosyadan).
 - Dizi boyutu, başlatma ve **indeks sınırlarına** dikkat.
- **2) Kitapları Listele:**
 - 0..length-1 aralığında dolaş, her kitabın temel bilgilerini yaz.
- **3/4/5) Arama:**
 - Üç farklı **overload** gerçekten farklı imzalarla çalışmalı (tür/sıra).
 - Bulunan kitap(lar)ı göster; bulunamazsa uygun mesaj ver.
- **6) Sırala (Selection Sort):**
 - Ortalama puana göre artan sırala.
 - Swap gereksinimini gözet; sıralama sonrası baş/orta/son öğelerden küçük bir özet bastır.
- **7) Kopya Al (Elle/clone):**
 - Bir kitap için puan dizisinin **elle kopyası** ve **clone** ile kopyası alınmış hallerini üret, alias olmadığını kanıtlayan kısa çıktı ver.
- **8) İçerik Eşitliği:**
 - İki puan dizisini eleman-eleman karşılaştır; eşit/eşit değil sonucu ve ilk farkın indeksi (varsı) yazdır.
- **9) İstatistik (2D):**
 - Seçilen yılın toplamını, en yoğun çeyreği ve (varsı) ragged satır uzunluklarını düzgün raporla.
- **10) ArrayList'e Aktar / Geri Dön:**
 - Book[] ↔ ArrayList<Book> dönüşümünü yap.
 - **size** ile **capacity** (gözlemleneyebildiğin kadariyla) farkına dair not düş; ArrayList'in yalnız **nesne** tuttuğunu hatırlat.
- **11) Vector Kapasite Gözlemi (opsiyonel):**
 - Başlangıç kapasitesi ver, eklemeler sonrası **size()** ve **capacity()** değerlerini ekrana yaz; **ensureCapacity** kullanımını göster.
- **0) Çıkış:**
 - Program sonlanır.

Girdi doğrulama: Geçersiz seçimde anlamlı uyarı ver ve menüye geri dön. Boş/eksik girişleri hataya düşürme, nazik mesaj ver.