

```
using System;
using System.Collections.Generic;
using System.Linq;

class Program
{
    static List<kitap> library = new List<kitap>();
    static List<kiralama> kiralamalar = new List<kiralama>();

    static void Main(string[] args)
    {
        bool programdevam = true;

        while (programdevam)
        {
            Console.Clear();
            Console.WriteLine("Kütüphane Yönetim Sistemi");
            Console.WriteLine("1. Kitap Ekle");
            Console.WriteLine("2. Kitap Kirala");
            Console.WriteLine("3. Kitap İade Et");
            Console.WriteLine("4. Kitap Arama");
            Console.WriteLine("5. Raporlama");
            Console.WriteLine("6. Çıkış");
            Console.Write("Bir seçenek girin: ");
            string seçim = Console.ReadLine();

            switch (seçim)
            {
                case "1":
                    kitapekle();
```

```
        break;
    case "2":
        kitapkiralala();
        break;
    case "3":
        kitapiadeet();
        break;
    case "4":
        kitapparama();
        break;
    case "5":
        raporlama();
        break;
    case "6":
        programdevam = false;
        break;
    default:
        Console.WriteLine("Geçersiz seçim! Tekrar dene.");
        break;
}

if (programdevam)
{
    Console.WriteLine("Devam etmek için bir tuşa basın...");
    Console.ReadKey();
}
}
```

```
static void kitapekle()
```

```
{  
    Console.Clear();  
    Console.WriteLine("Yeni Kitap Ekle");  
  
    Console.Write("Kitap Adı: ");  
    string kitapadı = Console.ReadLine();  
  
    Console.Write("Yazar Adı: ");  
    string yazaradı = Console.ReadLine();  
  
    Console.Write("Yayın Yılı: ");  
    int yayinyili = int.Parse(Console.ReadLine());  
  
    Console.Write("Adet: ");  
    int stokadet = int.Parse(Console.ReadLine());  
  
    kitap mevcutkitap = library.FirstOrDefault(k => k.adi == kitapadı && k.yazaradı == yazaradı);  
    if (mevcutkitap != null)  
    {  
        mevcutkitap.stok += stok;  
        Console.WriteLine("Kitap stoğu başarıyla artırıldı.");  
    }  
    else  
    {  
        library.Add(new kitap { kitapadı = kitapadı, yazaradı = yazaradı, yayinyili = yayinyili, stokadet =  
stokadet });  
        Console.WriteLine("Kitap başarıyla eklendi.");  
    }  
}
```

```
static void kitapkiralala()
{
    Console.Clear();
    Console.WriteLine("Kitap Kiralama");

    if (library.Count == 0)
    {
        Console.WriteLine("Kütüphanede hiç kitap yok.");
        return;
    }

    Console.WriteLine("Mevcut Kitaplar:");
    for (int i = 0; i < library.Count; i++)
    {
        Console.WriteLine("{i + 1}. {library[i].kitapadi} - {library[i].yazaradi} ({library[i].Yayinyili}) - Stok: {library[i].stokadet}");
    }

    Console.Write("Kiralamak istediğiniz kitabı seçin (1 - {0}): ", library.Count);
    int secim= int.Parse(Console.ReadLine()) - 1;

    if (secim < 0 || secim >= library.Count || library[secim].stokadet <= 0)
    {
        Console.WriteLine("Geçersiz kitap veya stokta yeterli kitap yok.");
        return;
    }

    Console.Write("Kiralamak istediğin gün sayısı: ");
    int gün = int.Parse(Console.ReadLine());
```

```
Console.Write("Bütçen: ");

decimal butce = decimal.Parse(Console.ReadLine());

decimal kirabedeli = days * 5;

if (butce >= kirabedeli)
{

    library[secim].stokadet--;

    Console.Write("Kullanıcı Adı: ");

    string kullanicadi = Console.ReadLine();

    kirabedeli.Add(new kirabedeli)
    {
        kullanicadi = kullanicadi,
        kitapadi = library[secim].kitapadi,
        kiralamasuresi = gun,
        iadetarifi = DateTime.Now.AddDays(gun)
    });

    Console.WriteLine("Kitap başarıyla kiralandı. İade tarihi: " +
        {DateTime.Now.AddDays(days).ToShortDateString()}");
}

else
{
    Console.WriteLine("Bütçeniz yeterli değil.");
}
}
```

```
static void kitapiadeet()
{
    Console.Clear();
    Console.WriteLine("Kitap İade");

    if (kiralama.Count == 0)
    {
        Console.WriteLine("Kiralanmış kitap bulunmamaktadır.");
        return;
    }

    Console.WriteLine("Kiralanan Kitaplar:");
    for (int i = 0; i < kiralama.Count; i++)
    {
        Console.WriteLine("{i + 1}. {kiralama[i].kitapadi} - {kiralama[i].kullaniciadi} - İade Tarihi: {kiralama[i].iadetarihi.ToShortDateString()}");
    }

    Console.WriteLine("İade etmek istediğiniz kitabı seçin (1 - {0}): ", kiralama.Count);
    int rentalIndex = int.Parse(Console.ReadLine()) - 1;

    if (rentalIndex < 0 || rentalIndex >= kiralama.Count)
    {
        Console.WriteLine("Geçersiz seçim.");
        return;
    }

    string kitapadi = rentalRecords[rentalIndex].kitapadi;
    library.FirstOrDefault(b => k.adi == kitapadi).stokadet++;
```

```
kiralama.RemoveAt(rentalIndex);
```

```
Console.WriteLine("Kitap başarıyla iade edildi.");
```

```
}
```

```
static void kitappara()
```

```
{
```

```
    Console.Clear();
```

```
    Console.WriteLine("Kitap Arama");
```

```
    Console.WriteLine("1. Kitap Adına Göre");
```

```
    Console.WriteLine("2. Yazar Adına Göre");
```

```
    Console.Write("Bir seçenek girin: ");
```

```
    string arama = Console.ReadLine();
```

```
    Console.Write("Aramak istediğiniz kelimeyi girin: ");
```

```
    string arama = Console.ReadLine();
```

```
    IEnumerable<kitap> aramasonuclari = null;
```

```
    if (aramasecenegi == "1")
```

```
    {
```

```
        aramasonuclari = library.Where(k => k.adi.StartsWith(arama,
StringComparison.OrdinalIgnoreCase));
```

```
    }
```

```
    else if (aramasecenegi == "2")
```

```
    {
```

```
    }
```

```
if (aramasonuclari != null && aramasonuclari.Any())
{
    Console.WriteLine("Arama Sonuçları:");
    foreach (var kitap in aramasonuclari)
    {
        Console.WriteLine("{kitap.adi} - {kitap.yazaradi} ({kitap.yayinyili}) - Stok: {kitap.stokadet}");
    }
}
else
{
    Console.WriteLine("Sonuç bulunamadı.");
}
}
```

```
static void raporlama()
{
    Console.Clear();
    Console.WriteLine("Raporlama");

    Console.WriteLine("1. Tüm Kitaplar");
    Console.WriteLine("2. Belirli Bir Yazara Ait Kitaplar");
    Console.WriteLine("3. Belirli Bir Yıla Ait Kitaplar");
    Console.WriteLine("4. Kirada Olan Kitaplar");
    Console.Write("Bir seçenek girin: ");
    string raporsecenegi = Console.ReadLine();

    if (raporsecenegi == "1")
    {
        Console.WriteLine("Tüm Kitaplar:");
        foreach (var kitap in library)
```



```

    {
        Console.WriteLine("{kitap.adi} - {kitap.yazaradi} ({kitap.Yayinyili}) - Stok: {kitap.stokadet}");
    }
}

else if (raporsecenegi == "2")
{
    Console.Write("Yazar Adı girin: ");
    string yazaradi = Console.ReadLine();

    var yazarkitapları = library.Where(k => k.k.yazaradi.StartsWith(yazaradi,
StringComparison.OrdinalIgnoreCase)).ToList();

    if (yazarkitapları.Any())
    {
        Console.WriteLine("\n\"{yazaradi}\" Yazarına Ait Kitaplar:");
        foreach (var kitap in yazarkitapları)
        {
            Console.WriteLine("{kitap.adi} - {kitap.Yayinyili} - Stok: {kitap.stokadet}");
        }
    }
    else
    {
        Console.WriteLine("Yazarın kitapları bulunamadı.");
    }
}

else if (raporsecenegi == "3")
{
    Console.Write("Yıl girin: ");
    int yil = int.Parse(Console.ReadLine());

    var yayinyilikitaplari = library.Where(k => k.yayinyili == yayinyili).ToList();

    if (yayinyili.Any())

```

```

{
    Console.WriteLine("{yayinyili} Yılında Yayımlanan Kitaplar:");
    foreach (var kitap in yayinyilikitaplari)
    {
        Console.WriteLine("{kitap.adi} - {kitap.yazaradi} - Stok: {kitap.stokadet}");
    }
}
else
{
    Console.WriteLine("Bu yılda yayımlanan kitap bulunamadı.");
}
}
else if (raporsecenegi == "4")
{
    Console.WriteLine("Kirada Olan Kitaplar:");
    foreach (var kayit in kiralama)
    {
        Console.WriteLine("{kayit.kitapadi} - Kiralayan: {kayit.kullaniciadi} - İade Tarihi: {kayit.iadetarihi.ToShortDateString()}");
    }
}
}
}

```

```

class kitap
{
    public string kitapadi { get; set; }
    public string yazaradi { get; set; }
    public int Yayinyili { get; set; }
    public int stokadet { get; set; }
}

```

```
}
```

```
class kiralama
```

```
{
```

```
    public string kullaniciadi { get; set; }
```

```
    public string kitapadi { get; set; }
```

```
    public int kiralamasüresi { get; set; }
```

```
    public iadetarihi { get; set; }
```

```
}
```

Kodu kendim yapabildiğim kadar ilerletmeye çalıştım zorlandığım noktalarda örnek videolar izleyerek bilgilerimi destekledim. Hatalı yerleri chatgbtye sorarak anlamaya çalıştım. Aynı zamanda arkadaşlarımla da yorumlaştık.

<https://github.com/senasafa>