using System;

using System.Collections.Generic;

using System.Linq;

namespace KutuphaneSistemi

{

class Kitap

{

public string Ad { get; set; }

public string Yazar { get; set; }

public int YayınYılı { get; set; }

public int StokAdedi { get; set; }

}

class KiralamaBilgisi

{

public string KullaniciAdi { get; set; }

public string KitapAdi { get; set; }

public int KiralamaSuresi { get; set; }

public DateTime IadeTarihi { get; set; }

}

class Program

{

static List<Kitap> kitaplar = new List<Kitap>

{

new Kitap { Ad = "1984", Yazar = "George Orwell", YayınYılı = 1949, StokAdedi = 5 },

new Kitap { Ad = "Suç ve ceza", Yazar = "fyodor dostoyevski", YayınYılı = 1866, StokAdedi = 10 },

new Kitap { Ad = "Simyacı", Yazar = "Paulo Coelho", YayınYılı = 1988, StokAdedi = 7 }

};

static List<KiralamaBilgisi> kiralamaBilgileri = new List<KiralamaBilgisi>

{

new KiralamaBilgisi

{

KullaniciAdi = "Görkem Ali Karabulut",

KitapAdi = "1984",

KiralamaSuresi = 7,

IadeTarihi = DateTime.Now.AddDays(3) // 3 gün içinde iade edilmesi gerekiyor

},

new KiralamaBilgisi

{

KullaniciAdi = "Jale Firuze Sinanoğulları", //isim uydurması zormuş :3

KitapAdi = "Suç ve ceza",

KiralamaSuresi = 10,

IadeTarihi = DateTime.Now.AddDays(5) // 5 gün içinde iadeetmesi gerekiyor

}

};

static void Main(string[] args)

{

while (true)

{

Console.Clear();

Console.WriteLine("Kütüphane Sistemi");

Console.WriteLine("1. Kitap Ekle");

Console.WriteLine("2. Kitap Kirala");

Console.WriteLine("3. Kitap İade");

Console.WriteLine("4. Kitap Arama");

Console.WriteLine("5. Raporlama");

Console.WriteLine("0. Çıkış");

Console.Write("Seçiminizi yapınız: ");

string secim = Console.ReadLine();

switch (secim)

{

case "1": KitapEkle(); break;

case "2": KitapKirala(); break;

case "3": KitapIade(); break;

case "4": KitapArama(); break;

case "5": Raporlama(); break;

case "0": return;

default: Console.WriteLine("Geçersiz seçim, tekrar deneyin :3"); break;

}

}

}

static void KitapEkle()

{

Console.Clear();

Console.WriteLine("Kitap Ekle");

Console.Write("Kitap Adı: ");

string ad = Console.ReadLine();

Console.Write("Yazar Adı: ");

string yazar = Console.ReadLine();

Console.Write("Yayın Yılı: ");

int yil = int.Parse(Console.ReadLine());

Console.Write("Adet: ");

int adet = int.Parse(Console.ReadLine());

var mevcut = kitaplar.FirstOrDefault(k => k.Ad == ad && k.Yazar == yazar);

if (mevcut != null)

{

mevcut.StokAdedi += adet;

Console.WriteLine("Kitap zaten var. Stok güncelleştirildi.");

}

else

{

kitaplar.Add(new Kitap { Ad = ad, Yazar = yazar, YayınYılı = yil, StokAdedi = adet });

Console.WriteLine("Kitap eklendi.");

}

Console.ReadLine();

}

static void KitapKirala()

{

Console.Clear();

Console.WriteLine("Kitap Kirala");

if (!kitaplar.Any())

{

Console.WriteLine("Kütüphanede kitap kalmadı...");

Console.ReadLine();

return;

}

for (int i = 0; i < kitaplar.Count; i++)

Console.WriteLine($"{i + 1}. {kitaplar[i].Ad} - {kitaplar[i].Yazar} (Stok: {kitaplar[i].StokAdedi})");

Console.Write("Kiralamak istediğiniz kitabın numarasını yazınız: ");

int no = int.Parse(Console.ReadLine()) - 1;

if (no < 0 || no >= kitaplar.Count)

{

Console.WriteLine("Geçersiz seçim.");

Console.ReadLine();

return;

}

Kitap secilenKitap = kitaplar[no];

if (secilenKitap.StokAdedi == 0)

{

Console.WriteLine("Bu kitap stokta bulunamadı.");

Console.ReadLine();

return;

}

Console.Write("Kiralamak istediğiniz gün sayısını yazınız: ");

int gun = int.Parse(Console.ReadLine());

int ucret = gun \* 5;

Console.Write("Bütçeniz ne kadar?: ");

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

if (butce < ucret)

{

Console.WriteLine("Bütçe yetmiyor."); //TT

Console.ReadLine();

return;

}

Console.Write("Adınız: ");

string ad = Console.ReadLine();

secilenKitap.StokAdedi--;

kiralamaBilgileri.Add(new KiralamaBilgisi

{

KullaniciAdi = ad,

KitapAdi = secilenKitap.Ad,

KiralamaSuresi = gun,

IadeTarihi = DateTime.Now.AddDays(gun)

});

Console.WriteLine($"Kiralama başarılı. İade tarihi: {DateTime.Now.AddDays(gun):d}");

Console.ReadLine();

}

static void KitapIade()

{

Console.Clear();

Console.WriteLine("Kitap İade");

if (!kiralamaBilgileri.Any())

{

Console.WriteLine("Kiralanmış kitap yok.");

Console.ReadLine();

return;

}

for (int i = 0; i < kiralamaBilgileri.Count; i++)

Console.WriteLine($"{i + 1}. {kiralamaBilgileri[i].KitapAdi} - {kiralamaBilgileri[i].KullaniciAdi}");

Console.Write("İade edeceğiniz kitabın numarası: ");

int no = int.Parse(Console.ReadLine()) - 1;

if (no < 0 || no >= kiralamaBilgileri.Count)

{

Console.WriteLine("Geçersiz seçim.");

Console.ReadLine();

return;

}

KiralamaBilgisi iadeEdilen = kiralamaBilgileri[no];

var kitap = kitaplar.First(k => k.Ad == iadeEdilen.KitapAdi);

kitap.StokAdedi++;

kiralamaBilgileri.RemoveAt(no);

Console.WriteLine("İade başarılı.");

Console.ReadLine();

}

static void KitapArama()

{

Console.Clear();

Console.WriteLine("Kitap Arama");

Console.Write("Kitap adı veya yazar adı girin: ");

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

var sonuc = kitaplar.Where(k => k.Ad.ToLower().Contains(arama) || k.Yazar.ToLower().Contains(arama));

foreach (var kitap in sonuc)

Console.WriteLine($"{kitap.Ad} - {kitap.Yazar} ({kitap.YayınYılı}) - Stok: {kitap.StokAdedi}");

if (!sonuc.Any())

Console.WriteLine("Sonuç bulunamadı.");

Console.ReadLine();

}

static void Raporlama()

{

Console.Clear();

Console.WriteLine("Raporlama");

Console.WriteLine("1. Tüm Kitaplar");

Console.WriteLine("2. Kiralanan Kitaplar");

string secim = Console.ReadLine();

if (secim == "1")

{

foreach (var kitap in kitaplar)

Console.WriteLine($"{kitap.Ad} - {kitap.Yazar} ({kitap.YayınYılı}) - Stok: {kitap.StokAdedi}");

}

else if (secim == "2")

{

foreach (var kiralama in kiralamaBilgileri)

Console.WriteLine($"{kiralama.KitapAdi} - {kiralama.KullaniciAdi} (İade: {kiralama.IadeTarihi:d})");

}

Console.ReadLine();

}

}

}