# ÖDEV DOKÜMANINI DETAYLI BİR ŞEKİLDE İNCELEMEDEN ÖDEVE BAŞLAMAYIN

PROGRAMLAMAYA GİRİŞ DERSİ

PROJE ÖDEVİ

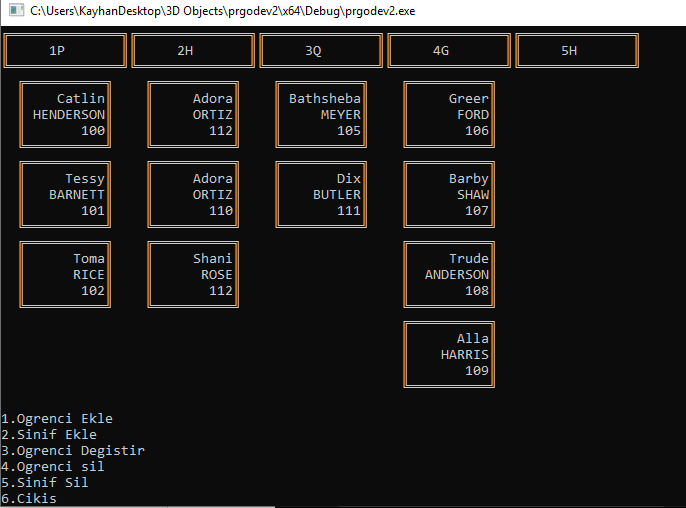
## Amaç

Örnek çalıştırılabilir dosyası ve ekran çıktısı aşağıda verilmiş olan C++ uygulamasını geliştiriniz.

**Örnek Çalıştırılabilir Program**

<https://drive.google.com/open?id=1VJ_YLBv-oQgtBUNUBxsfXy0dbhun9UwO>

**Örnek Ekran Çıktısı**



Kullanıcı Menüsü

4G sınıfına ait öğrenciler

Sınıflar

## Ogrenci Sınıfı

* Ogrenci sınıfı isim, soy isim ve numara değerini barındırmalıdır.
* Sınıfın sahip olduğu değişkenlerin dışarıdan direk erişime kapalı olası gerekir.
* Değişkenlere erişim üye fonksiyonlar aracılığıyla yapılabilir.
* Ogrenci ismi, “**isimler.txt”** dosyasında bulunan isimlerden rastgele seçilmelidir.
* Ogrenci soy isimi **“soyisimler.txt”** dosyasında bulunan soy isimlerden rastgele seçilmelidir.
* Kendi istediğiniz fonksiyon ve değişkenleri de ekleyebilirsiniz.

## DosyaYonetim Sınıfı

* DosyaYonetim sınıfı proje içerisinde dosya okuma ve yazma işlemleri ile ilgilenmektedir.
* Sınıf, çağrıldığında **isimler.txt** dosyasını okuyup buradan rastgele bir isim getirecek bir fonksiyona sahip olmalıdır.
* Sınıf, çağrıldığında **soyisimler.txt** dosyasını okuyup buradaki soy isimlerden rastgele birisini getirecek bir fonksiyona sahip olmalıdır.
* Sınıf, kendisine verilen Ogrenci nesnesini **kayitlar.txt** dosyasına kaydedebilecek bir fonksiyona sahip olmalıdır.
* Sınıf, kendisine verilen Sinif nesnesini **kayitlar.txt** dosyasına kaydedebilecek bir fonksiyona sahip olmalıdır.
* Sinif, **kayitlar.txt** dosyasındaki verileri okuyup bu verilere göre bir Okul nesnesi oluşturup getirecek bir fonksiyona sahip olmalıdır. ?

## Sinif Sınıfı

* Sınıf içerisinde **sinifIsmi** adında **string** bir değişken olmalı.
* Yeni bir sınıf oluşturulduğunda **sinifIsmi** değişkeni toplam sınıf sayısı değeri ile başlamalı ardından A-Z arasında rastgele bir değer eklenmelidir. Örneğin aşağıda daha önceden eklenmiş sınıf isimleri bulunmaktadır. Eğer yeni bir sınıf ekleyecek olsak sınıfın ismi 6 ile başlayacak ve ikinci karakteri A ile Z arasında rastgele bir değer alacaktır. Örneğin 6R gibi.



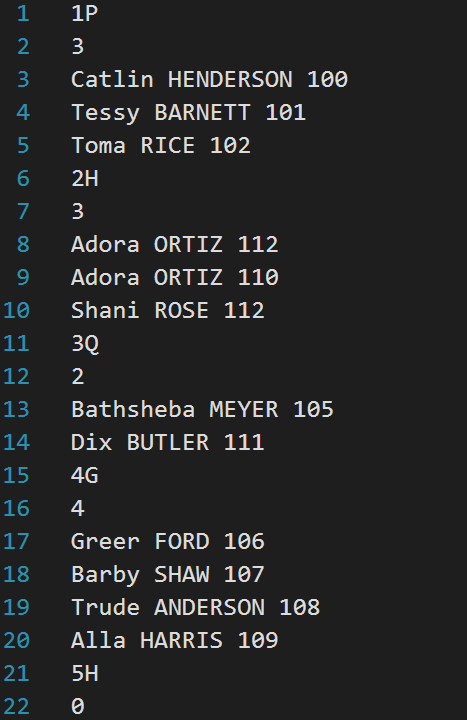
* Sınıf, içerisinde sahip olduğu öğrenci nesnelerini tutacak bir dizi olmalıdır.(Max 100 öğrenci olacağı düşünülecek.)
* Sınıf içerisinde **ogrenciEkle** adında bir fonksiyon olacak. Bu fonksiyon parametre olarak öğrenci nesnesi alacak ve aldığı nesneyi öğrenci dizisine ekleyecektir.
* Sınıf içerisinde **ogrenciSil** adında bir fonksiyon bulunacaktır. Bu fonksiyon numarası verilen öğrenciyi öğrenci dizisinden çıkartacaktır.
* Sınıf içerisine gerek duyduğunuz fonksiyon ve elemanları ekleyebilirsiniz.

## Okul Sınıfı

* Okul sınıfı kurucu fonksiyonu **“kayitlar.txt”** dosyasını okuyup, dosya içerisindeki sınıf ve öğrenci bilgilerini okuyup bunları kendi içerisinde bir sınıf dizisi içerisinde saklamalıdır. Dosya yönetim sınıfının dosya yükleme işlemi sırasında kullanılması tavsiye edilir.

Sınıf ismi

Sınıftaki öğrenci sayısı

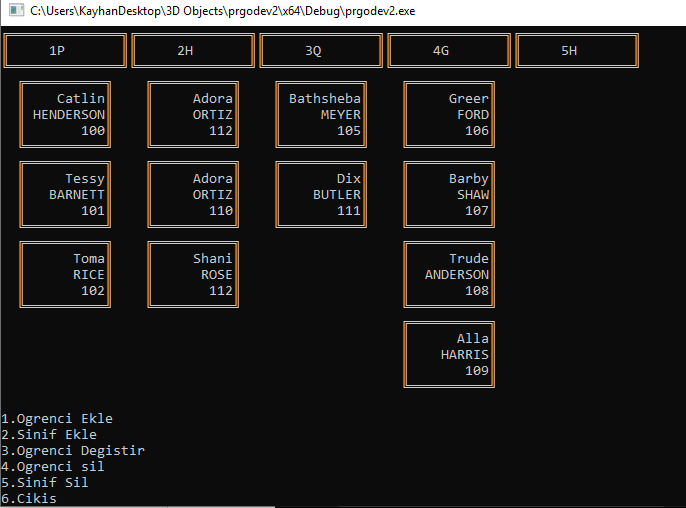


Öğrenci bilgisi sırasıyla

İsim soyisim numara

Değerine sahiptir.

* Bütün sınıfları ve öğrencileri aşağıdaki biçimde ekrana çıkartacak bir fonksiyona sahip olmalıdır.



* Okul içerisindeki bütün sınıfları ve sınıflara ait olan öğrencileri **kayitlar.txt** dosyasına yazdıran bir fonksiyon barındırmalıdır. **Kayitlar.txt** dosyasının formatı daha önce verilmiştir.

## Program Sınıfı

* Program sınıfı içerisinde bir adet okul nesnesi barındırmalıdır.
* Sınıf içerisinde **calistir** isimli bir fonksiyon bulunmalıdır. Bu fonksiyonun görevi okul nesnesinin yazdır metodunu kullanarak sınıf ve öğrencileri ekrana çıkartmalı. Ardından kullanıcı menüsü ekrana çıkartılmalı ve kullanıcıdan seçim istenmelidir.
* Menü kontrol işlemlerinin tümü bu sınıf içerisinde yapılmalıdır.

## main fonksiyonu

* **main** fonksiyonu içerisinde sadece iki ifade bulunmalıdır. İlk ifade bir adet program nesnesi oluşturmalıdır. İkinci ifade bu nesnenin **calistir** metodu çağrılmalıdır. main fonksiyonu içerisinde başka bir kod bulunmamalıdır. (Aşağıdaki gibi sadece iki satır ifade bulunmalıdır)

Program prog;

prog.calistir();

## Geliştirdiğiniz uygulamanın örnek çalıştırılabilir program içerisindeki her işlemi gerçekleştirmesi gerekmektedir.

## Çizim için yardımcı fonksiyonlar

Aşağıdaki karakter ve fonksiyonlar aracılığıyla projede istenilen çizimleri yapabilirsiniz. Kodları direk kopyalayıp yapıştırmak probleme yol açabilir. Bu fonksiyonları Okul sınıfı içerisine yerleştirebilirsiniz.

ustYazdir fonksiyonu üst kısmı çizdirir. ╔═══════╗

altYazdir fonksiyonu alt kısmı çizdirir ╚═══════╝

char solUstKose = 201;

char duz = 205;

char sagUstKose = 187;

char solAltKose = 200;

char sagAltKose = 188;

char dikey = 186;

void ustYazdir(int elemanSayisi)

{

cout << solUstKose;

for (int i = 0; i < elemanSayisi; i++)

{

cout << duz;

}

cout << sagUstKose;

}

void altYazdir(int elemanSayisi)

{

cout << solAltKose;

for (int i = 0; i < elemanSayisi; i++)

{

cout << duz;

}

cout << sagAltKose;

}

UYULMASI GEREKEN KOD DÜZENİ

Her C++ dosyasının başında aşağıdaki yorum bloğu bulunacaktır. Yorum bulunmayan her C++ dosyası için 10 puan kırılacaktır. (**pdf üzerinden kopyalayıp yapıştırmanız problem çıkartabilir**)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*

\*\* ÖĞRENCİ ADI...........................:

\*\* ÖĞRENCİ NUMARASI.............:

\*\* DERS GRUBU…………………:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Değişken isimleri anlamlı olmalıdır. Örneğin tek sayı adedini tutacak bir değişken için

|  |  |  |  |
| --- | --- | --- | --- |
| YANLIŞ | yerine | DOĞRU | yazılmalıdır. |
| int a; | int tekAdedi; |

Her bir küme parantezinin altındaki kodları yazmadan önce **tab** tuşuna basılarak boşluk bırakılmalıdır.

|  |  |  |
| --- | --- | --- |
| YANLIŞ |  | DOĞRU |
| if(true)  {  int a =6;  cout<<"Merhaba";  if(false)  cout<<"Nasilsin";  } |  | if(true)  {  int a =6;  cout<<"Merhaba";  if(false)  cout<<"Nasilsin";  } |

Her döngü ve koşul işlemlerinden önce yapılan işlem hakkında yorum yazılmalıdır

|  |  |  |
| --- | --- | --- |
| YANLIŞ |  | DOĞRU |
| int t = 0;  for(int i=0;i<100;i++)  t+=i; |  | int toplam = 0;  //1 den 100 e kadar olan sayılar toplanıyor  for(int i=0;i<100;i++)  toplam+=i; |

ÖDEV TESLİM KURALLARI

# Proje Ödevi son ders haftasında dersin öğretim üyesi tarafından belirlenen ders saatlerinde sunulacaktır. Sunum sırasında öğrenci bilgisayarını getirmeli ve programın çalışmasına dair soruları yanıtlaması gerekmektedir. Sunulmayan ödevler teslim edilmemiş sayılacaktır.

# Sabise yüklenecek dosyalar

* Sisteme sadece projedeki C++ kod dosyası yüklenecektir.
* Kod dosyasının dosya ismi aşağıdaki gibi olmalıdır

Dosya isimleri: **OgrenciNumarasi.cpp**

Örneğin;

**b000110002.cpp** (b000110002 numaralı bir öğrenci)

DİKKAT!!!!

Gönderdiğiniz dosyanın kod dosyası olduğuna emin olmadan yollamayın. Dosyanızı Notepad programı ile açarak kontrol edin. Yanlış dosyanın gönderilmesi ödevinizin değerlendirme dışı kalmasına sebep olur.

# DERLEYİCİ VE TESLİM TARİHİ

Derleme için **Visual Studio 2019**  C++ derleyicisi kullanılacaktır.

**Dosya isimlerinde yapılan hatalar, 20 puan kırılmasına sebep olacaktır.**

**Zamanında teslim edilmeyen ödevler değerlendirilmeyecektir.**

**( En ufak bir gecikme ödevin kabul edilmemesi ile sonuçlanır)**

KOD PAYLAŞMAK YASAKTIR

Ödevler bireyseldir ve verilen bütün ödevler karşılaştırılacaktır. Birbirine çok benzeyen ödevler **kopya** muamelesi görecektir. Öğretim üyesi kopya durumunda ödevi değerlendirmez veya gerekli soruşturma ve ceza işlemlerini başlatabilir.