



## Yıldız Yazılımcı Yetiştirme Programı

*Türkiye'nin en yenilikçi yazılımcı yetiştirme programı*

# OKUL YÖNETİM SİSTEMİ (OYS)



Bir okul yönetim sistemi kurulmasına gereksinim var

- 1) Okulda farklı iş tanımları (memur, hizmetli, öğretmen) olan çalışanlar bulunmakta.

Her çalışanın sicil numarası (M/H/Ö-nnnn şeklinde: *M-023, H-452 gibi*) ismi, cinsiyeti, evlilik durumu, doğum tarihi, işe başlama ve bitirme tarihleri, iş başlangıcında aldıkları maaş ve en fazla iki adet telefonu (cep, ev) kayıt altında tutulmak zorunda. İsim (önisim, göbek adı ve soyismi) evlilik dışında değiştiremeyecek bir çalışan özelliği. Her çalışanın göbek adı olması zorunluluğu yok, ama varsa bir daha değiştirilemez

*Not: Sicil numarasının nümerik kısmı öğrencilerin okul numarasına benzer şekilde üretilmeli*



2) Okulda ayrıca öğrenciler bulunmakta.

Her öğrencinin kayıt numarası (YYYY-nnn şeklinde: *2020-001, 2020-002 gibi*) ismi, cinsiyeti, doğum tarihi, okula başlama ve bitirme tarihleri, hangi sınıfta olduğu ve en fazla üç adet telefonu (cep, ev, veli) kayıt altında tutulmak zorunda. Öğrencilerin evlenme imkanları yok, evlenmek isterlerse uygulama bir istisna ile durumu yakalamalı

*Not: Kayıt numarasının nümerik (nnn) kısmı ardışık olarak numaralandırılmış şekilde olmalı. Yani ilk öğrenci YYYY-001 ile başlamalı ikinci öğrenci YYYY-002 gibi ilerlemeli 999'a varıldığında tekrar 001'den başlamalı*



- 3) Okulda, açılan derslere bir asıl bir de yedek öğretmen atanması gerekiyor. Her dersin bir ismi, hangi sene verildiği ve derse katılan öğrencilerin listesi kayıt altında tutulmak zorunda. Ayrıca dersin kaç saatlik olduğu ve haftanın hangi günü hangi saatler arasında verildiği de bilinmeli

Dersleri sadece okulda çalışan «memur» statüsündeki çalışanlar açabilir ve öğretmen ve öğrenci atamalarını yapabilirler. Bir derse öğrenci olarak bir «öğretmen» veya «hizmetli» atanamayacağı gibi bir «öğrenci» veya «hizmetli» de öğretmenlik yapamaz. «Memur»lar derslere hiçbir şekilde katılamazlar



## 4) OYS'de aşağıda sıralanan işler yapılabilinmeli

### «Ana Menü»

Tanımlanan işlevleri içeren tek kademeli bir menü

Uygulama özellikle istenmediği sürece bu menü içinde çalışacak. Bitirilme isteğinde kullanıcıya gerçekten bunu isteyip istemediği tekrar sorulacak

### I. «Yeni Çalışan»

Çalışan yaratma (sadece (1)'de tanımlanan iş tanımlarına uyan çalışanlar yaratılabilir

### II. «Çalışan Listesi»

Çalışan listesi gösterme

### III. «Eski Çalışanlar»

Eski çalışanları işe giriş, çıkış tarihleri ve kaç yıl ve ay çalışmış oldukları bilgisi ile listeleme



## IV. «Yeni Sınıf»

Yeni sınıf açma (sınıfın öğretmenleri ve kaç öğrencilik bir sınıf bilgisi ile)  
(öğretmenler sadece daha önce tanımlanmış öğretmenlerden olabilirler)

## V. «Öğrenci Atama»

Sınıfa öğrenci atama. Eğer öğrenci daha önceden yaratılmış ise öğrenci numarası sınıfa kayd edilir. (bu sınavda öğrencileri tek tek kullanıcı arayüzü üzerinden tanımlamanıza gerek yok. Uygulamanın bir yerinde öğrencileri constructorlar ile tanımlayabilirsiniz)

## VI. «Sınıf – Listesi»

Sınıf bilgisi gösterme

(sınıf adı verilince o sınıfın öğretmenleri ve öğrencileri listelenir)



## VII. «Öğrencinin katıldığı sınıflar»

Bir öğrenci numarası verildiğinde o öğrencinin girdiği derslerin listesi gösterilir

## VIII. «Hediye Listesi»

Okul yönetimi çalışanlarının işe başlama tarihlerinde bir ufak hediye gönderiyor. Aynı şekilde sadece kadın çalışanlara da doğum günlerinde bir hediye gönderiliyor. İçinde bulunulan ayda kimlerin işe başlama kimlerin doğum günü olduğunun bir listesinin her an çıkartılabilinmesi gerekli

## IX. «Maaş – Listesi»

Çalışanların maaşları şu şekilde hesaplanıyor:

- İşe başladıkları tarihten sonra geçen her altı ayda bir öğretmenlere %10, memurlara %9 ve hizmetlilere %8.5 zam yapılıyor
- İstenilen bir tarihte güncel maaşların listesinin çıkartılabilinmesi gerekli

## X. «Program bitirme»



- Ana menü doğru implemente edildi ise: 5 Puan
- (1), (2), (3), (8), (9) sorular: 10ar Puan
- (10) implemente edildi ise: 5 Puan
- Menüü hem Türkçe hem İngilizce göstermek (program argument ile) 10 Puan (5 + 5)
- (4), (5), (6), (7) numaralı sorular 30 Puan (10, 5, 10, 5)
- Her seferinde veri girişlerinden kurtulmak isterseniz nasıl bir çözüm uygularsınız?
  - a. Sadece sözlü olarak anlatım fazladan 10 Puan
  - b. İmplementasyon fazladan 20 Puan  
(Sözlü anlatıp bir de implemente ederseniz yine de 20 puan, yani 30 puan almak imkansız)





- Alınabilecek en yüksek puan 100'dür
- Ancak istenilen (4), (5), (6), (7) dışındaki işlevler gerçekleştirildi ise ekstra puanlar değerlendirilmeye eklenir
- CodingStandart'lara uyum, isimlendirme kurallarına bağlılık, nesne tabanlı yazılım geliştirme yaklaşımına yakınlık daha iyi değerlendirilmenize yardım edecektir
- Uygulamanıcı JAR halinde email olarak gönderebilirsiniz (gönderim saati 18:00'i geçemez)



Bir iki tüyo:

- Aynı bilgileri gereksinimleriniz doğrultusunda birden fazla veri tipinde saklayabilirsiniz
- Önyüzden veri girişi yapmanıza gerek yok. Yapıcı metotlarla üretmeniz yeterli (ikişer nesne üretmeniz yeterli olacaktır)
- toString() yöntemlerini unutmayın
- Mümkün olduğunca BAUtils metotlarını kullanın
- Yapabildiklerinizi ve hızınızı karşılaştırabilmek için aşağıdaki sıra ile implementasyonu yapın:
  1. Öncelikle ana menü ve (1), (2), (3), (8), (9), (10) numaralı menüleri bitirin
  2. Ardından (4) ve (5) numaraları yapmaya çalışın
  3. Vaktiniz yeterse (6) ve (7)'yi implemente edin



# BilgeAdam

boost

## TEŞEKKÜRLER