**T.C.  
  
 MANİSA CELAL BAYAR ÜNVERSİTESİ  
  
KIRKAĞAÇ MESLEK YÜKSEKOKULU  
  
BİLGİSAYAR PROGRAMCILIĞI**

**Ders Programı Hazırlama Otomasyonu**

Hazırlayanlar

**231809087 ENES ERMİŞ**

**231809015 MEHMET POLAT ALEMDAR**

**231809003 HİLAL AŞGİT**

**231809084 HASAN AKBUDAK**

Danışman

**Öğr.Gör. LEVENT KARAKUZ**

İçindekiler Tablosu

[**Giriş** 3](#_Toc156219893)

[Uygulamayı kurma: 3](#_Toc156219894)

[IntelliJ IDEA Kurulumu: 3](#_Toc156219895)

[Uygulamayı Açma: 3](#_Toc156219896)

[Proje Açma: 3](#_Toc156219897)

[Bağımlılıkları Yükleme: 3](#_Toc156219898)

[Main Class'ı Başlatma: 3](#_Toc156219899)

[Mysql Ayarlama: 3](#_Toc156219900)

[Uygulama ne işe yarıyor ? 4](#_Toc156219901)

[Hızlı ve Verimli Program Oluşturma: 4](#_Toc156219902)

[Çakışmaları Önleme: 4](#_Toc156219903)

[En Uygun Atamaları Yapma: 4](#_Toc156219904)

[Esneklik ve Uyum: 4](#_Toc156219905)

[Teklif edilen Sistem 4](#_Toc156219906)

[1. Kısıtlı Yetkili Kullanıcı 4](#_Toc156219907)

[2. Tam Yetkili Kullanıcı 5](#_Toc156219908)

[Otomasyonumuz 6](#_Toc156219909)

[Kullanılabilirlik 6](#_Toc156219910)

[Güvenilirlik 6](#_Toc156219911)

[Performans 6](#_Toc156219912)

[Desteklenebilirlik 6](#_Toc156219913)

[İmplementasyon 6](#_Toc156219914)

[Arayüz 6](#_Toc156219915)

[Gizlilik Gereksinimi 6](#_Toc156219916)

[Aktörler 6](#_Toc156219917)

[Olaylar 7](#_Toc156219918)

[Seneryolar 7](#_Toc156219919)

[Veri Tabanı Diyagramı: 8](#_Toc156219920)

[Program Aşamaları Kodlarla 9](#_Toc156219921)

[Çalıştırma: 9](#_Toc156219922)

[Giriş Yapma: 10](#_Toc156219923)

[Menü Oluşturma : 12](#_Toc156219924)

[Field Oluşturma: 16](#_Toc156219925)

[ComboBox Oluşturma: 17](#_Toc156219926)

[Tablo Özelleştirme: 18](#_Toc156219927)

[Butonlar: 21](#_Toc156219928)

[Geri Git: 21](#_Toc156219929)

[Veriyi Sil: 23](#_Toc156219930)

[Veriyi Kayıt Et: 23](#_Toc156219931)

[Veriyi Güncelle: 29](#_Toc156219932)

[Cıktı Al: 32](#_Toc156219933)

[Özel Cıktı Al: 35](#_Toc156219934)

[Algoritma Aşamaları Kodlarla 39](#_Toc156219935)

[1. Bölümlerin Müsaitliği 40](#_Toc156219936)

[2. Dersliklerin Müsaitliği 41](#_Toc156219937)

[3. Öğretmen Müsaitliği 43](#_Toc156219938)

[4. Ders Bilgileri 45](#_Toc156219939)

[Programı Başlatmak 47](#_Toc156219940)

[Alogirtma Akış Diyagramı… 54](#_Toc156219941)

# **Giriş**

## Uygulamayı kurma:

### IntelliJ IDEA Kurulumu:

* Öncelikle IntelliJ IDEA'nın resmi web sitesinden uygun sürümü indirin ve kurulumu gerçekleştirin. ([İndirmek için tıkla](https://www.jetbrains.com/idea/download/download-thanks.html?platform=windows&code=IIC))
* Kurulum sihirbazını takip ederek temel ayarları seçip, IntelliJ IDEA'nın bilgisayarınıza başarılı bir şekilde kurulmasını sağlayın.

### Uygulamayı Açma:

* Kurulumun tamamlanmasının ardından IntelliJ IDEA uygulamasını başlatın.
* Sol üst köşede bulunan "File" menüsüne tıklayın. ([Github İçin Tıkla](https://github.com/UgurAyebe0/sistemAnalizi))

### Proje Açma:

* "File" menüsünden "Open" seçeneğine tıklayın ve bilgisayarınızda bulunan proje dosyanızı seçin.

### Bağımlılıkları Yükleme:

* Projenizi başarıyla açıktan sonra, bağımlılıkları yüklemek için “Ctrl + Shift + O” tuş kombinasyonunu kullanın ve IntelliJ IDEA'nın sağ alt köşesinde bulunan "Load Maven Changes" seçeneğine tıklayın.

### Main Class'ı Başlatma:

* Sağ alt köşede bulunan "Fun" menüsünden "Main Class" seçeneğine gidin.
* Ardından "Public Class Main" başlığını bulun ve başlatmak için uygun seçeneği seçin.

## Mysql Ayarlama:

Otomasyonu sorunsuz bir şekilde kullanmak ve veritabanı işlemlerini gerçekleştirmek için MySQL bağlantı bilgilerinizi doğru bir şekilde ayarlamak oldukça önemlidir. Aşağıda, bağlantı bilgilerini düzenlemek için adım adım bir rehber bulabilirsiniz:

Daha sonrasunda Fun -> Database -> Config dosyasında database ayarlarını ayarlamamız için ufak bir kod satırı mevcuttur.   
  
metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Burada {} parantezlerin içine yazdiğim yerler Mysql bilgilerinizi güncelleneniz gerekmektedir.

## Uygulama ne işe yarıyor ?

Modern eğitim ortamlarında yöneticilerin en büyük zorluklarından biri, öğretmen ve ders kaynaklarını etkili bir şekilde yöneterek, öğrencilere en iyi eğitimi sunmaktır. Bu noktada, ders programı oluşturmak sıklıkla zaman alıcı ve karmaşık bir süreç olabilir. Ancak, şimdi bu zorluğu ortadan kaldıran bir çözüm var: Akıllı Ders Programı Otomasyonu!

Bu benzersiz uygulama, yöneticilere öğretmen bilgileri, ders içerikleri, okul zamanı ve bölüm detayları gibi kritik verileri girdikten sonra, ders çakışmalarını önleyen ve en uygun şekilde bölümlere atama yapabilen bir akıllı sistem sunuyor. İşte bu uygulamanın sunduğu faydalar:

### Hızlı ve Verimli Program Oluşturma:

Geleneksel yöntemlerle saatler süren ders programı oluşturma sürecini, bu otomasyon sayesinde sadece birkaç tıklama ile tamamlayabilirsiniz. Bu, yöneticilere zaman kazandırarak daha önemli görevlere odaklanma imkanı sunar.

### Çakışmaları Önleme:

Uygulama, dersler arasındaki çakışmaları etkili bir şekilde engeller. Bu, öğrencilere daha düzenli bir program sunmanın yanı sıra, öğretmenlere de daha etkili bir şekilde ders vermelerine olanak tanır.

### En Uygun Atamaları Yapma:

Dersleri, öğretmenlerin uzmanlık alanlarına ve bölümlere en uygun şekilde atayarak, öğrencilerin daha iyi bir eğitim almasını sağlar. Bu sayede, her öğrenci kendi ilgi ve yeteneklerine uygun derslere katılabilir.

### Esneklik ve Uyum:

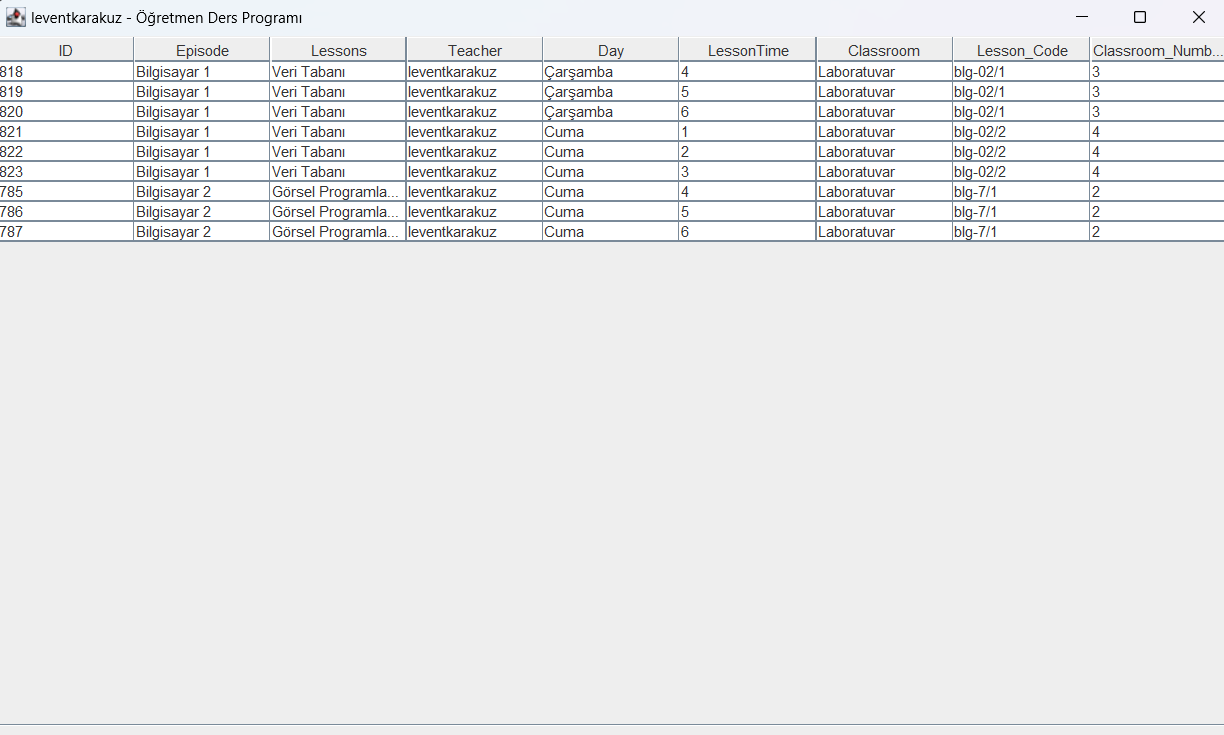
Uygulama, değişen okul koşullarına ve öğretmen programlarındaki güncellemelere hızlı bir şekilde adapte olabilir. Bu da yöneticilere esneklik sağlayarak, herhangi bir aksaklığa karşı hazırlıklı olmalarını sağlar.

Bu Akıllı Ders Programı Otomasyonu ile, eğitim kurumları daha hızlı, daha düzenli ve daha verimli bir şekilde ders programları oluşturabilir. Bu da öğrencilerin ve öğretmenlerin en iyi şekilde faydalanmasını sağlayarak eğitim kalitesini artırır. Zamanınızı daha etkili bir şekilde yönetmek ve daha iyisini başarmak için şimdi bu benzersiz uygulamayı keşfedin!"

## Teklif edilen Sistem

Sistemde 2 Adet kullanıcı vardir. Tam Yetkili ve Kısıtlı yetkili olmak üzere şimdi detaylıca anlatalım..

### 1. Kısıtlı Yetkili Kullanıcı

Kısıtlı yetkili kullanıcılar öğretmenlerdir ve öğretmenler programda sadece kendi bilgilerini görebilmektedir. Ayrıca cıktı olarak hem tablo halinde hemde ders programı halinde de cıktı alabilirler…  
  
  
Kısıtlı yetkili kullanıcılar örneğin bu şekilde verilerini görebilirler.

### 2. Tam Yetkili Kullanıcı

Tam yetkili kullanıcılar menülerde herşey yapabilecek kullanıcılardır. Ufak bir ekran görüntüsünden neler yapabileceklerini gösterelim…

Şimdi sadece bu menüye göre yorumlarsak kapataslak Tam yetkili kullanıcılar,  
  
1. Unvan İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

2. Öğretmen İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

3. Müsaitlik İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

4. Derslik İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

5. Öğretmenlere ders aktarma İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

6. Bölümlere ders aktarma İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

7. Okulun Acık olacağı günleri belirleme (Ekleme, Düzeltme, Silme, Cıktı Alma)

8. Okulun açılacağı saati, ders süresini,tenefüs süresini, öğlen molasını, öğlen mola süresini, okulun kapatılacağı saati belirleme işlemleri.

9. Bütün programları tabloda listeleyip gerekli İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

10. Sadece Öğretmene göre gerekli İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

11. Sadece Bölüme göre gerekli İşlemleri (Ekleme, Düzeltme, Silme, Cıktı Alma)

12. Ders Programını sıfırdan oluşturma.

Tam Yetkili bir kullanıcı buradaki tüm bilgileri eksiksiz bir şekilde girdiği zaman ders programını hazırlayabilir. Eksik bilgilerde hata uyarı mesajlarımız mevcuttur.

## Otomasyonumuz

### Kullanılabilirlik

Projede, ders programı hazırlama sürecini hızlandırmak için kullanıcıların ihtiyaç duyduğu ana işlevlere hızlı erişim sağlayan menüler bulunmalıdır. Bu menüler, kullanıcıların istedikleri işlemlere kolayca ulaşmalarını sağlamalı ve zaman kaybını minimuma indirmelidir. Oluşturulan program istenildiği gibi cıktı alabilir ve basit bir arayüzü sayesinde kullanıcılara güçlü zorluklar yaratmamaktadır. Ayrıca en üst düzeyde optimize sağlanmış ve geliştirilebilmesi çok basit bir şekilde kodlanmıştır

### Güvenilirlik

Ders programı otomasyonunda verilerin korunması ve güvenliği oldukça önemlidir. Yaşanılabilecek 3. Parti uygulamala açıklarının ve illegal girişleri engellemek için ufak güvenilir kod bütünlüğü kullanarak bu açıkları kapattık. Mesela login sistemindeki sql saldırılarını engelledik. Yada DDOS saldırılarını azaltmak için programda ufak engeller ekledik. Onun dışında database’nizi kodumuzun içerisinde güvenilir bir şekilde gömdüğümüz için bizlere güvenebilirsiniz. Ama en önemlisi yaşayacağınız en ufak sorunda yanınızda bulunacağımızı bilmelisiniz.

### Performans

Performans olaraktan verileri sürekli databaseden çekip yaşayacağımız ağ trafiğinden dolayı verilerimizi nodelere kayıt edip en son işlemleri veritabanından yapıyoruz bu sayede daha az dişardan bağlantı cekerek performansımızı güncelndirdik Java 17 ile en düşük sistem gereksinimlerinde bile çalışacak uygun kod bloklarını ve döngüleri en az kullanacak şekilde programı oluşturduk.

### Desteklenebilirlik

Projede kullanılan veritabanının transaction ve foreign key desteğinin olması gerekli. Programın kurulduğu platformun da Java destekli olmalı ve Java programlarını çalıştırabilmeli. Özgür yazılım lisansı sayesinde yasal gereksinimleri bulunmamakta

### İmplementasyon

Sistem için verilen veritabanlarını dikkatli bir şekilde bağlaştırıp UML Diyagramımızı oluşturduk aşağıda diyagramımız mevcut ve buna göre gerekli java kodlarında bağlantılarımızı destekledik.

### Arayüz

Otomasyonda kullanıcı arabirimi pencerelerdir. Tek bir pencerede yer alacak programda, kullanıcıyı login sayfası karşılar. Login kullanıcı, rolüne göre kendine ait bilgilere ulaşır. Görsel özellikler için gereksinim dökümanı ayrıca hazırlanacaktır. Kullanıcı düğmelere basarak veya formları doldurarak işlem yapacaktır. Ara yüzün çıktısı ekran, girdileri ise Mouse ve klavyedir.

### Gizlilik Gereksinimi

Kullanıcıların sınırlandırılması, sadece kendilerine ait bölümleri görebilmesi ve gerektiğinde bu ayarların değiştirilmesi yüksek derecede önemli. Veritabanında, dataların güvenliği için ek bir özellik bulunmuyor. Sunucunun başına geçen veya dosyaları çalan birisi kayıtlara ulaşabilir. Sunucu güvenliği işletim sisteminin kendi güvenliği ile eş. Program açılırken kullanıcı adı ve şifre soracak. Sistem bu bilgileri kullanarak izinlerini kontrol edecektir. Kullanıcılar veritabanına başka bir programla bağlanıp bilgileri görebilir. Programda kullanılan gizlilik sadece verinin bütünlüğü ve iş bölümü için. İstenmeyen kullanıcıların sisteme bağlanmasını engellemek işletim sisteminin sorumluluğundadır.

### Aktörler

|  |  |
| --- | --- |
| **Aktör** | **Tanım** |
| Tam Yetkili | Sistem içerisinde tamamen bütün yetkinin ait olduğu ve sistem içerisindeki bütün düzenlemeleri yapan kullanıcıdır. Kısaca sistemdeki YÖNETİCİ’dir |
| Kısıtlı Yetkili | Sadece ders programını görüntüler ve sistemde cıktı alabilir. Kısaca yetkisi olmayan Öğretmenlerdir. |
| Sistem | Ders programı oluşturma, çakışmaları kontrol etme, otomatik derslik ve laboratuvar atama gibi görevleri gerçekleştiren bir sistem aktörüdür.  Çakışmaları kontrol eder ve yöneticiye uyarılar verir, ancak müdahale etmez. Yönetici, uyarıları değerlendirip istediği şekilde değişiklik yapabilir. |

### Olaylar

Sistemin haftalık ders programı hazırlama otomasyonu olaylarını şu şekilde sıralayabiliriz:

**Hoca Bilgisi Girişi:**

Yönetici, hocaların bilgilerini (unvanları ile birlikte) sisteme girebilir.

**Ders Bilgisi Girişi:**

Yönetici, derslerin bilgilerini (ders kodu, adı, ders saati) sisteme girebilir.

**Hoca-Ders Atama:**

Kullanıcı, hangi hoca hangi derse gireceğini sisteme girebilir.

Bir hoca birden fazla derse atanabilir.

Bir derse birden fazla hoca atanabilir.

**Şube ve Program Ayarları:**

Yönetici, bir şube için günlük ders saatlerini, başlangıç saatini, ders süresini, teneffüs süresini, öğle tatili süresini belirleyebilir.

**Çakışma Kontrolü:**

Sistem, aynı gün ve saatte ismi farklı birden fazla derse atanmış bir hocanın durumunu kontrol eder.

Eğer çakışma varsa, yöneticiye uyarı verir.

**Müsaitlik Girişi:**

Hocalar, müsait oldukları günleri ve saatleri sisteme girebilir.

**Derslik ve Laboratuvar Atama:**

Sistem, bir dersin laboratuvarda yapılması gerekiyorsa otomatik olarak laboratuvar ataması yapar.

Eğer müsait bir laboratuvar bulunamıyorsa, sistem yöneticiye müsait derslikleri önerir.

**Otomatik Derslik Atama:**

Sistem, bir dersin öğrenci sayısına göre en uygun dersliğe otomatik olarak atama yapar, en az boş yer bırakan dersliği seçer.

**Çakışma İzni:**

Yönetici, aynı gün ve saatte ismi farklı birden fazla derse atanmış bir hocanın durumu hakkında uyarı aldığında, sistemin önerdiği çözümü değerlendirip onay verebilir.

**Program Çıktıları:**

Kullanıcı, bölüm, sistem ve hoca bazında ayrı ayrı haftalık program çıktıları alabilir.

**Manuel Değişiklikler:**

Yönetici, ders programında istediği şekilde manuel değişiklikler yapabilir.

Sistem, kriterlere uygun olmayan durumlar hakkında yöneticiyi uyarır, ancak müdahale etmez.

### Seneryolar

***Hoca Eklemek:***

Yönetici, yeni bir hoca ekleyerek unvan, ad, soyad gibi bilgileri sisteme kaydeder.

***Ders Eklemek:***

Yönetici, yeni bir ders ekleyerek ders kodu, ders adı, ders saati gibi bilgileri sisteme kaydeder.

***Hoca-Ders Ataması Yapmak:***

Kullanıcı, belirli bir hoca için hangi dersin atanacağını seçer ve sisteme kaydeder.

***Şube Programı Ayarlamak:***

Yönetici, bir şube için günlük ders saatleri, başlangıç saati, teneffüs süresi gibi program ayarlarını düzenler.

***Çakışma Kontrolü:***

Sistem, bir hoca için çakışan ders atamalarını kontrol eder ve yöneticiye uyarı verir.

***Müsaitlik Bilgisi Girmek:***

Hoca, kendi müsait olduğu günleri ve saatleri sisteme girer.

***Derslik ve Laboratuvar Ataması:***

Sistem, bir dersin laboratuvarda yapılması gerekiyorsa otomatik olarak laboratuvar ataması yapar.

***Otomatik Derslik Atama:***

Sistem, derslerin öğrenci sayılarına göre en uygun dersliklere otomatik olarak atama yapar.

***Çakışma İzni Verme:***

Yönetici, çakışan ders atamaları konusunda sistem tarafından verilen uyarılara izin verir.

***Program Çıktısı Almak (Bölüm):***

Kullanıcı, belirli bir bölüm için haftalık ders programını sisteme sorgular ve çıktı alır.

***Program Çıktısı Almak (Sistem):***

Kullanıcı, genel sistem haftalık ders programını sorgular ve çıktı alır.

***Program Çıktısı Almak (Hoca):***

Kullanıcı, belirli bir hoca için haftalık ders programını sisteme sorgular ve çıktı alır.

***Manuel Değişiklik Yapmak:***

Yönetici, ders programında manuel değişiklik yapar ve sistem tarafından verilen uyarıları değerlendirir.

***Ders Sayısı Azaltmak:***

Yönetici, bir dersin haftalık saat sayısını azaltır ve sistemden olası çakışma uyarılarını değerlendirir.

***Derslik Kapasitesini Güncellemek:***

Yönetici, bir dersliğin kapasitesini günceller ve sistemin otomatik atama algoritmasını etkiler.

***Hoca Bilgisi Güncellemek:***

Yönetici, bir hocanın bilgilerini günceller ve sistemdeki ilgili atamalara yansıtır.

***Ders Bilgisi Güncellemek:***

Yönetici, bir dersin bilgilerini günceller ve sistemdeki ilgili atamalara yansıtır.

***Derslik Bilgisi Eklemek:***

Yönetici, yeni bir derslik ekler ve sistemde otomatik atama algoritmasını etkiler.

***Tüm Ders Atamalarını Silmek:***

Yönetici, tüm ders atamalarını siler ve sistemde temiz bir başlangıç yapar.

***Çakışma Kontrolsüz İzin:***

Yönetici, çakışan ders atamalarına sistem uyarısı olmadan izin verir.

## Veri Tabanı Diyagramı:

Veritabanında kurduğumuz bağlantılar aşağıda bir görselde belirtilmiştir. Belirtilen görselde kurulan bağlantılar (FOREIGN KEY) güncelleme işlemlerinde veri güncellemesi için ayrıntılı kullanılmıştır.

ekran görüntüsü, diyagram, metin, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Program Aşamaları Kodlarla

## Çalıştırma:

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

Program ilk olarak bu class’tan başlamaktadır. NotExits.Main methoduna ilk olarak gidecektir. Gittikten sonra Mevcut olmayan Tableleri oluşturmamızı sağlıyor.  
metin, ekran görüntüsü, yazı tipi, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Buradan tek tek table’leri oluşturuyoruz. Mesela örnek bir methodu açalım. metin, ekran görüntüsü, ekran, görüntüleme, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Buradan bu şekilde databaseleri oluşturuyoruz. Gördüğünüz method lesson.main() methodur. Bu method sayesinde databasemizde eğer Lesson tablosu yoksa oluşturacaktır. Bütün tabloları bu şekilde oluşturduktan sonra bir menü açmamız lazım. Menü açılmasını diğer başlıkta daha detaylı anlatacağım o yüzden bu kısmı geçiyorum.

## Giriş Yapma:

Uygulamayı çalıştıktan sonra login ekranına atar login ekranında ise kullanıcının olup olmamasını tabloda o kullanıcı adına ve şifresine sahip 1 kullanıcı varmı yokmu diye bir sorgu yapıp kullanıcının doğruluğunu aldım.

metin, yazılım, yazı tipi, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu kod parçasında da gördüğünüz üzere…  
  
metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Burada buttona bir eylem aktardım ve buton ismini Giriş yap olarak değiştirdim. Acılan pencereden username ve password bilgilerini alıp boş olup olmadiğinin kontrolunu sağladım. Daha sonrasında bir verify böyle bir kullanıcı olup olmadiğina baktım. Bundan sonra artık bu kullanıcının op (Yetkisine) kontrol etmem gerekli. Kullanıcımı yoksa yönetici mi diye bunuda getOpValueByUsername methoduyla yani metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Bu kod parçasından alıyorum. Eğer 1 ise bu kullanıcı Tam Yetkili eğer 0 ise bu kullanıcı Kısıtlı yani bir öğretmen olduğunu gösteriyorum.  
  
metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

Açıklama otomatik olarak oluşturuldu

Buradan 0 (Kısıtlı Yetki) ,1 (Tam Yetki) yetki türüne göre pencerelere atıyorum. Ve gerekli hata mesajlarım ekranda göründüğü gibi eklenmiştir.

## Menü Oluşturma :

Asıl konumuzdan birisi menü oluşturma, menü oluşumlarını en basit olacak şekilde ayarlamaya çalıştım. Şimdi ufak bir kod parçasından ilerleyerek nasıl oluştuğuna bakalım…

Bilgilendirme (?)

Label Penceredeki Yazi

Field Pencerede yazı girilen kutu

ComboBox Pencerede seçim yaptıran kutu

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

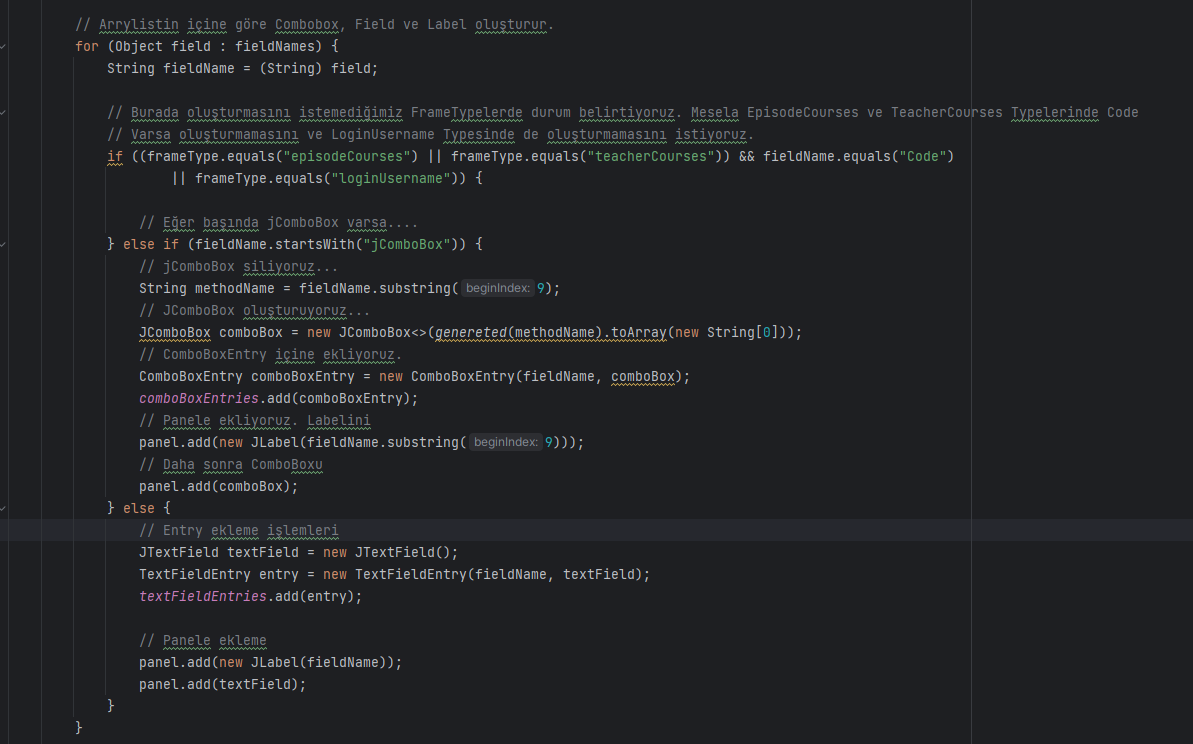
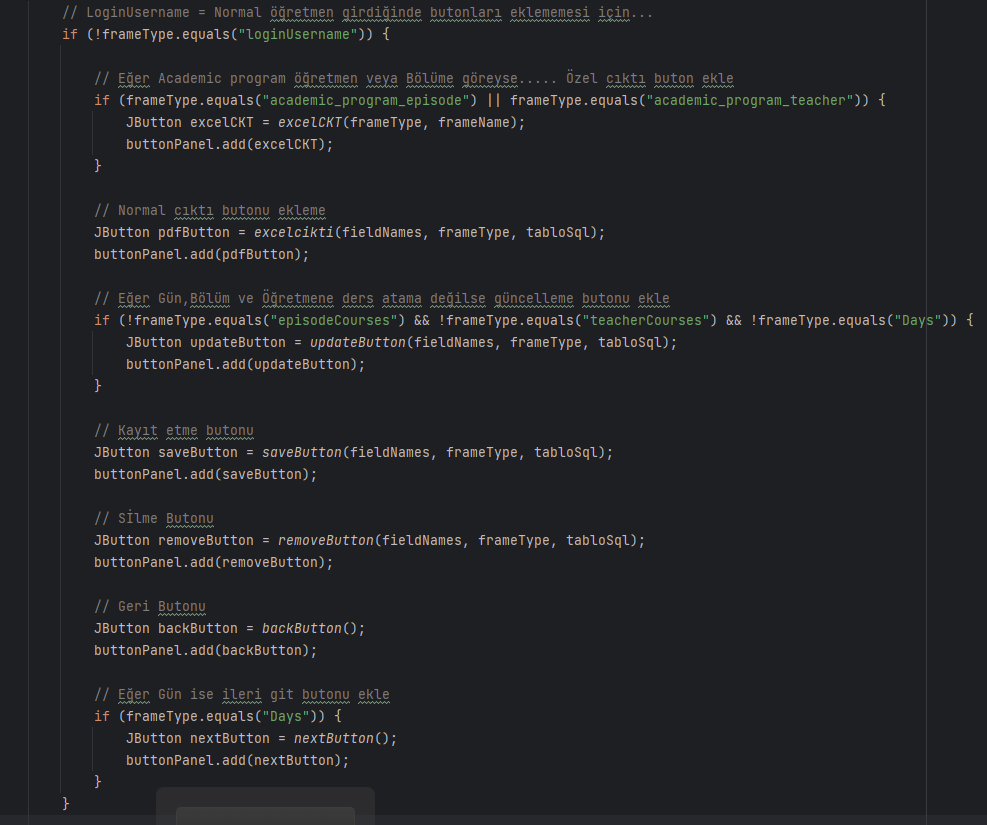
Açıklama otomatik olarak oluşturuldu  
Burada öğretmenler penceresini oluşturmuşuz. İlk önce mevcut frame kapatmak için frame.dispose() methodunu çağırdım. Daha sonrasında fieldName adında bir ArryList oluşturdum. Oluşturduğum Arryliste Eklemeleri belirtiyorum. Mesela öğretmenler tablosunda Title(Ünvan), Op(Yetkisi),TC,PhoneNumber(Telefon Numarası), Name(İsim), Surname(Soyisim),(Username) kullanıcı adı, password(Şifre) bilgilerini istiyorum. İngilizce olmasına bakmayın databasenizde table isimlerini yazmanız lazım. Ben tercihen ingilizce kulandığım için ingilizce yazdım. Şimdi burada şunu dile getiriyoruz herhangi bir kısıtlama olmadiği zaman bu Fieldleri oluştur ve Labellerinide haliyle otomatik oluşturacak. Fakat biz bunu Field değilde combobox olmasını istersek eğer kelimenin başına jComboBox yazmak zorundayız.  
  
Bu ArryList sayesinde metin, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Böyle bir görüntüye ulaşabiliriz. Pencerenin en üst kısmı şuanda bu şekilde oluşturulmuştur. Daha sonrasında FrameName yani pencerenin sol üst kısmında gözükecek isim, Pencerenin büyüklüğüne gelirsekte genişlik olarak frameSizeX, yükseklik olarak frameSizeY değerlerini giriyoruz. Daha sonrasında Arrylistimizi ekliyoruz. Rows yani kaç stün olacak 4 stün olarak yapılmasını istedik ve bu şekilde max 4 stün olacak şekilde bir görüntü verdi. Daha sonrasında bu olaya bir FrameType isim belirliyoruz. Daha sonra aşağıda oluşacak Table için sql sorgumuzu yazıyoruz.  
metin, ekran görüntüsü, sayı, numara, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
ArryList ve Sql sorgumuz bağlantılı sırasıyla belirtilmelidir. Mesela table’mizde ilk gelen Arrylist ilk eklenen, Tabloda da ilk gözüken olur. Sırasını değiştirmek isterseniz \* ifadesi yerine tek tek sırasını yazmak zorundasınız.  
  
Şuan güncel olarak   
metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu Bu fotoğrafta gözükecek kısma kadar geldik. Şimdi daha kodlama kısımlarına geçip anlatacağım.  
metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu  
CreateFrame.main() methodu parametreleri ile bu şekilde çalışır. Frame verilen bilgilere göre oluşturur. Daa sonrasında sırasıyla ilk önce almış olduğumuz ArryListlerimizi   
  
Buradan özelliğine göre çözümleyip menüye ekliyoruz. Detayları başlıklarda anlatacağım… Şimdi butonları nasıl menüye ekleyeceğimizi göstereceğim… CreateFrame classında 

Bu şekilde butonları kısıtlayarak veya normal bir şekilde ekleyebilirsiniz..

## Field Oluşturma:

Field oluşturmamızda çok fazla detayımız yok aslında metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

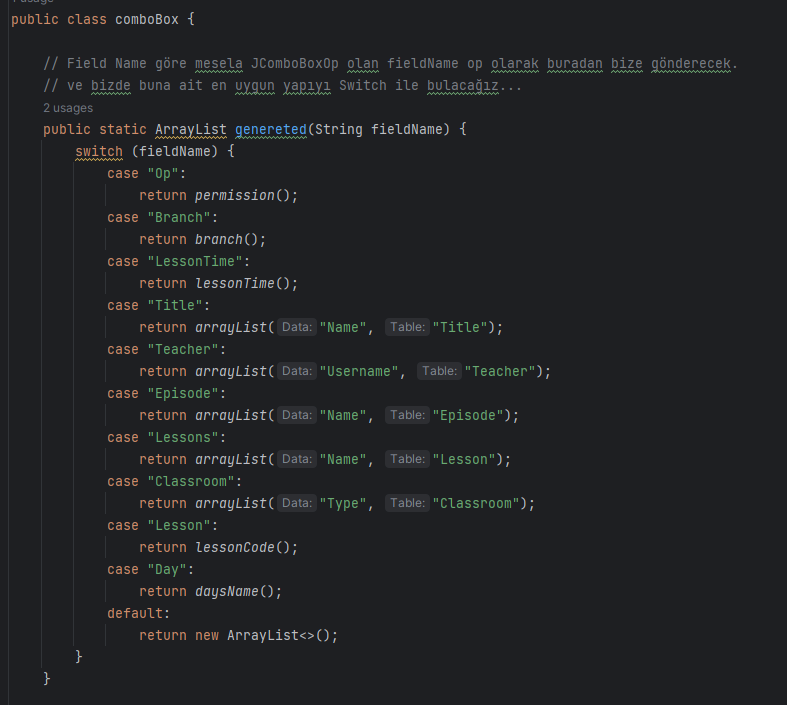
## ComboBox Oluşturma:

ComboBox oluşturma methodumuz aslında biraz uzun cünkü veri kaydetmesi mevcut…  
metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu şekilde öğeleri oluşturduktan sonra , 

Bu kısımda Combobox oluştururkenki genereted methoduna gidiyoruz..

  
Buradan ilk önce gelen veriyi alacağız ve return olarak bir liste göndereceğiz. Buradan Swich içerisinde comboBox içerisindeki veriyi oluşturabiliriz.

## Tablo Özelleştirme:

Menümüzde olacak tablo hakkında kullandığım kod blokları ve açıklamaları,

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Ve tekrardan createFrame methodunu cağirirkenki framelerden FieldNames arrylistini kullanarak burada tablo oluşturur ve sonuç olarak   
Bu kısmı kodladık.. şimdi içine bilgi ekleme kısmına geliyoruz…

  
Bu kod bloğu sayesinde artık tablomuzun güncel görüntüsü metin, ekran görüntüsü, sayı, numara, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu bu şekilde oluşturulmuştur. Şimdi sırayla butonlara gelelim…

## Butonlar:

Şimdi size sırayla butonların nasıl çalıştığını anlatacağım. Sanırım en uzun kısımlara gelmiş bulunmaktayız.

### Geri Git:

Aslında en kısa butonlarımızdan birtanesini şuanda size anlatacağım..

metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

Açıklama otomatik olarak oluşturuldu

### Veriyi Sil:

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

### Veriyi Kayıt Et:

Veriyi kayıt etmek için belirli şartlar ve olaylar vardir. Sırayla inceleyelim..

metin, elektronik donanım, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şimdi saveData methoduna gelelim…

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

getFieldOrder methodumuz.. metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

Açıklama otomatik olarak oluşturuldu

valid (Kontrol) methodumuzda bütün kontrolleri sağlayacağız… Fakat eğer öğe ekliceksek False öğe düzenliceksek true olmalı buna lütfen dikkat edin…

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Mesela burada kelime uzunluğu ve boş olup olmadiğini kontrol ediyoruz….

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Mesela bunun gibi birçok kontrolleri buradan (Valid methodu) gerçekleştireceğiz…

Şimdi sonraki olayımız DataComboBox methodu burası cok önemli cünkü Foreing key olan verileri comboboxta adını gösteriyorduk fakat bunları id olarak kayıt edeceğimiz için verileri toplamamız lazım… metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Buradan özellikle toplayabiliriz bu verileri….

handleComboBoxField methoduna gelecek olursak…  
metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şimdi herşeyi düzenlediğimize göre bir Sql sorgusuna ihtiyacımız var bunuda artık

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Buradan gerekli sql sorgusunu oluşturabiliriz ve artık veriyi kayıt edebiliriz…

### Veriyi Güncelle:

Kayıt ederken aslında buradaki coğu şeyi yaptığımız için güncellemenin üstünde cok fazla durmayacağım…

metin, ekran görüntüsü, yazılım içeren bir resim

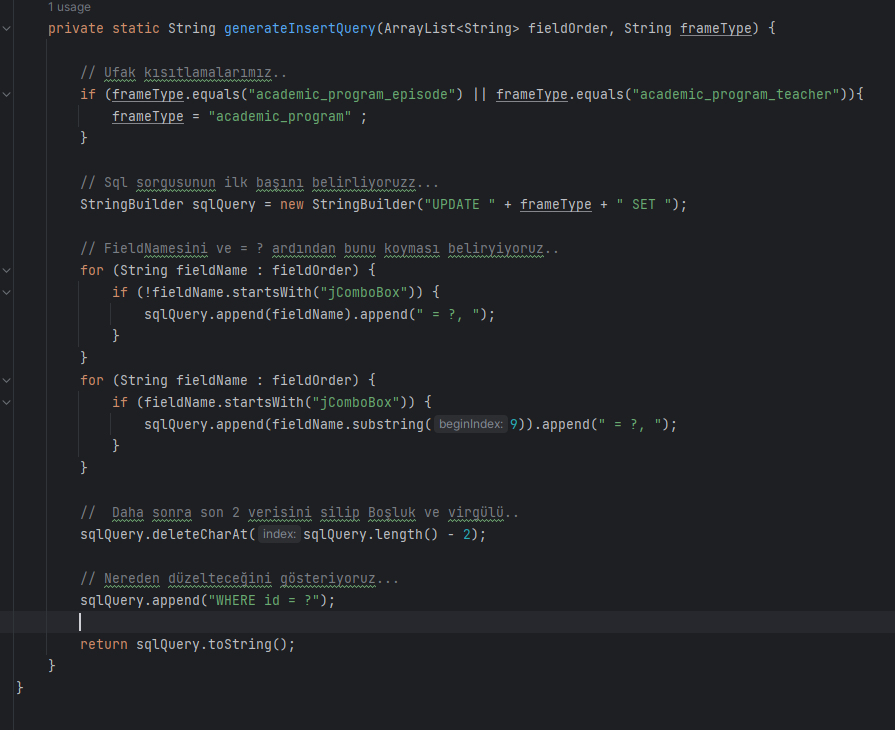
Açıklama otomatik olarak oluşturuldu

Uyarıya basıldığı zaman metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu böyle bir yazı mesajı gelmesi lazım ki kullanıcı bilgileri kontrol edip onaylasın…

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Onun içinde böyle bir kod bloğu kullandım… sql sorgumuzuda  bu sefer bu şekilde oluşturuyoruz…

### Cıktı Al:

Cıktı alma işleminde normal çıktıda tablo gibi bir cıktı alabilmemizi sağlıyor. Aslında tablodaki veriyi cıktı olarak alıyoruz burada…. Gerekli kodları inceleyelim..

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu kod bloğunda tablonun aynısını excelde bize yansıtacaktır… Örnek bir cıktı gösterim…  
metin, ekran görüntüsü, sayı, numara, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Örnek çıktımız bu şekildedir.

### Özel Cıktı Al:

Ders programı olarak cıktı almak istediğimiz zaman nasıl cıktı alabileceğimizi şimdi size anlatacağim… İlk öncelikle kendimize olası durumlara karşılık bir placeholders hazırlayacağız ve daha sonrasında verileri uygun Placeholderslere aktaracağız. Örnek taslağımı aşağıda göstereyim.

Taslağın tamamını gösteremezsemdde büyük bir kısmı aşağıda görüntülenmektedir.

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Burada gördüğünüz gibi placeholderslerimi oluşturdum.. {lesson1} bu placeholdersim aynı şekilde pazartesi 1. Ders için {lesson\_1\_1\_lesson\_Pazartesi} gibi placeholderslerimiz mevcut. 1 Bölüme sadece 2 şube açmaya izin verdiği için programımız bu şekilde hazırladım ama siz isterseniz bunu artırabilirsiniz….  
  
Şimdi placeholdersleri nasıl kullanacağımızı anlatayım..  
metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

İlk önce bir node oluşturmamız lazım.. ve bu node gerekli bilgileri işliyoruz.. Node içeriğimiz aşağıdaki gibidir…  
metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Bu şekilde node bilgiler çekiyoruz ve bunları kullanıyoruz…  
metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Gerekli excel olaylarını işliyoruz… şimdi sırada değiştirme işlemleri… Öncelikle replace işlemi yapmak için gerekli olan methodumuz.   
metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Bu şekilde methodu çağırıp replace yapacağız…

İlk önce ders saatlerini ve kaçıncı ders olduklarını replace yapalım…

ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bunları replace çekip aşağıdaki gibi göstereceğiz..  
  
Bunun için gerekli kod satırı,

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şimdi gelelim ders içerisini güncelleme kodlarımıza bunlarıda aşağıdaki gibi güncelleyeceğiz..

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Artık son olarak tablo ismini belirteceğiz bu tablonun ne olduğunu bilmesi için.. metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Onuda böyle ufak bir kodla replace cekiyoruz… Son olarak doldurulmamış stünları kaldırmak için gerekli kodumuz

metin, yazı tipi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu  
Cağirdiğimiz methoddaki kod bloğunun içerisi…

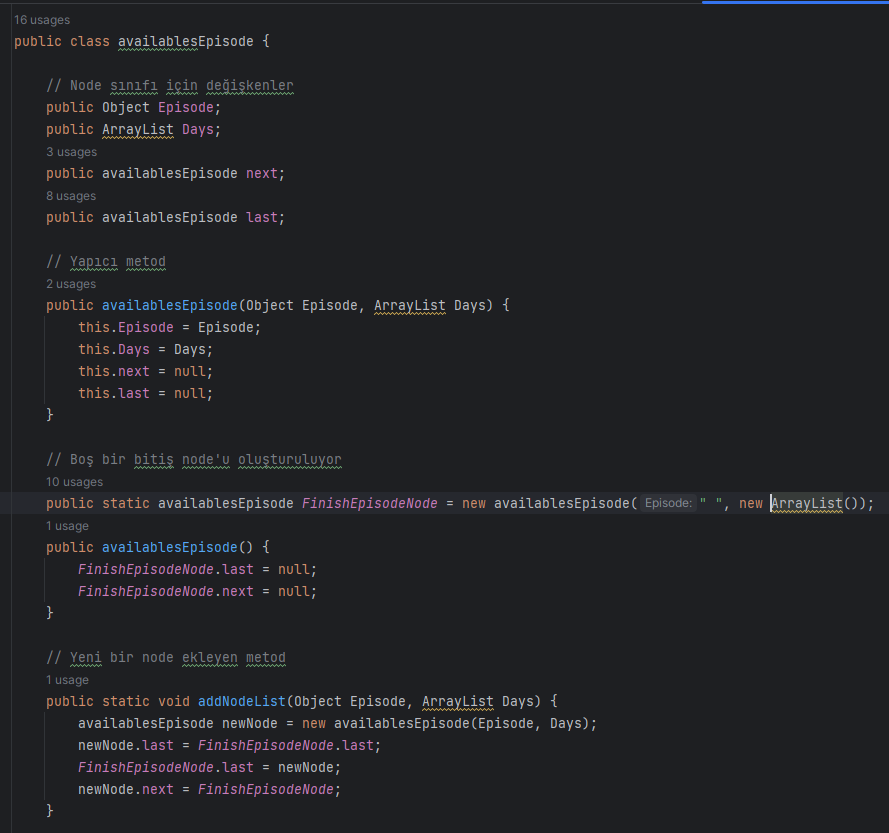
metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Algoritma Aşamaları Kodlarla

Şimdi şuana kadar yazılan kodlardaki herşeyi kapataslak anlatmış bulunmaktayız. Şuana kadar kurut işlemlerinin hepsinin nasıl yapılacağını ve yapıldığını gösterdim. Şimdi ders programı otomasyonunu çalıştırmayı anlatacağım.  
  
Algoritmada yaşadığım en büyük sıkıntı Local makinemde çok fazla sorgu yaptığım için xamp’ın firewallını çok fazla tetikledim. Bu yüzden algoritmada çok fazla sorgu yapmamak için gerekli bütün bilgileri bir Node’ye kayıt edeceğim.

## 1. Bölümlerin Müsaitliği

Algoritmada bölümlerin o gün o ders başka ders almaması için müsait olup olmadiğini kontrol edeceğiz algoritma ilk olarak tüm gün müsait olarak düşünecek ve nodelere o şekilde kayıt edecek daha sonrasında ders atadıkça bu kaydı silecek.  


metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

## 2. Dersliklerin Müsaitliği

Burada aynı dersliklere birden fazla bölüm ve ders eklememek için dersliklere de müsaitlik durumu belirtip ekleyeceğiz..

metin, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

## 3. Öğretmen Müsaitliği

Öğretmenlere aynı gün aynı ders birden fazla ders atamamak için onlara müsaitlik durumu ekleyeceğiz ama availabilityteacher tablosuna dikkat ederek ekleyeceğiz.

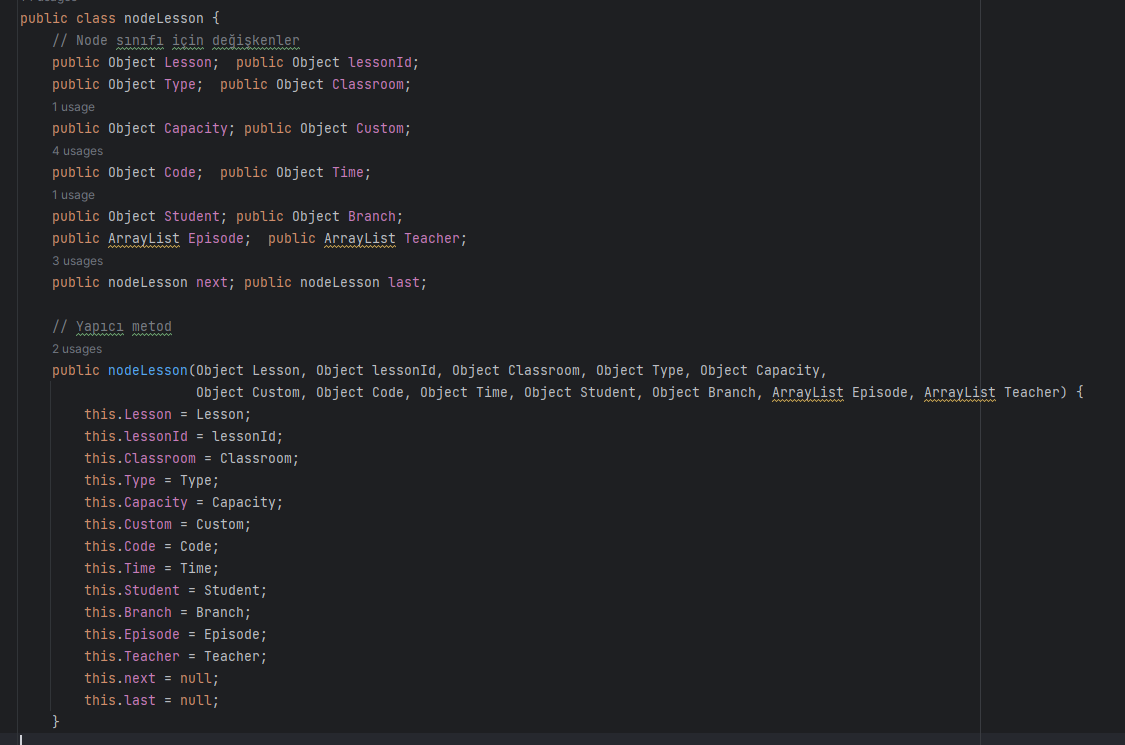
metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

## 4. Ders Bilgileri

Şimdi ders bilgilerini kayıt etmek için bir node oluşturacağız..  


metin, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

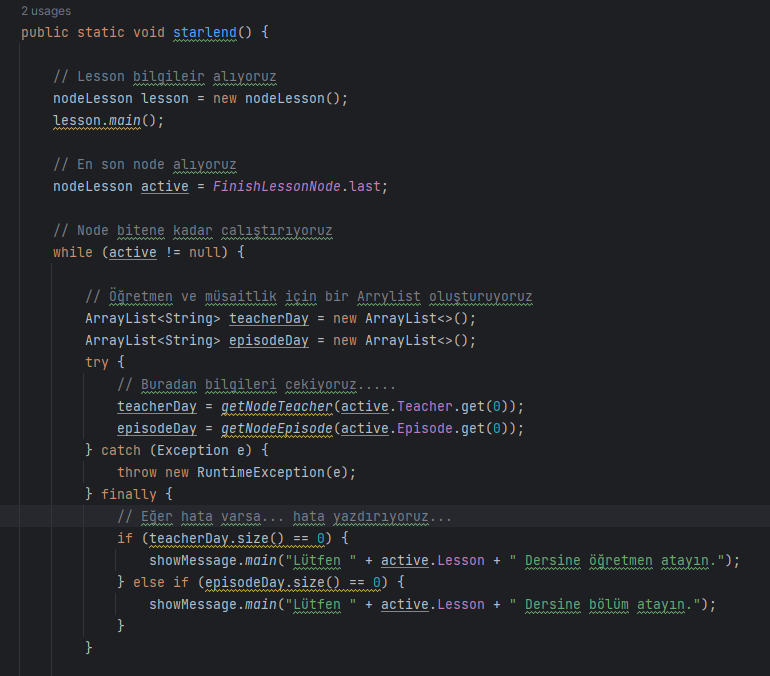
## Programı Başlatmak

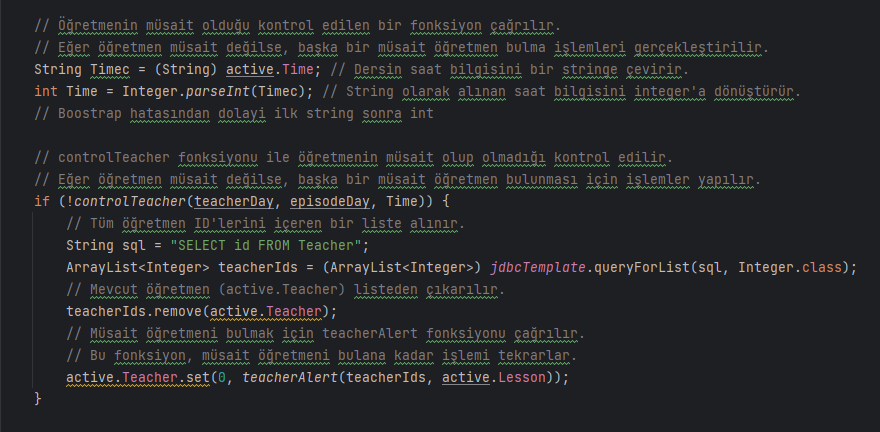
Şimdi programı başlatmak için ilk önce nodeleri çalıştıracağız çalıştırmak için

metin, elektronik donanım, ekran görüntüsü, yazılım içeren bir resim

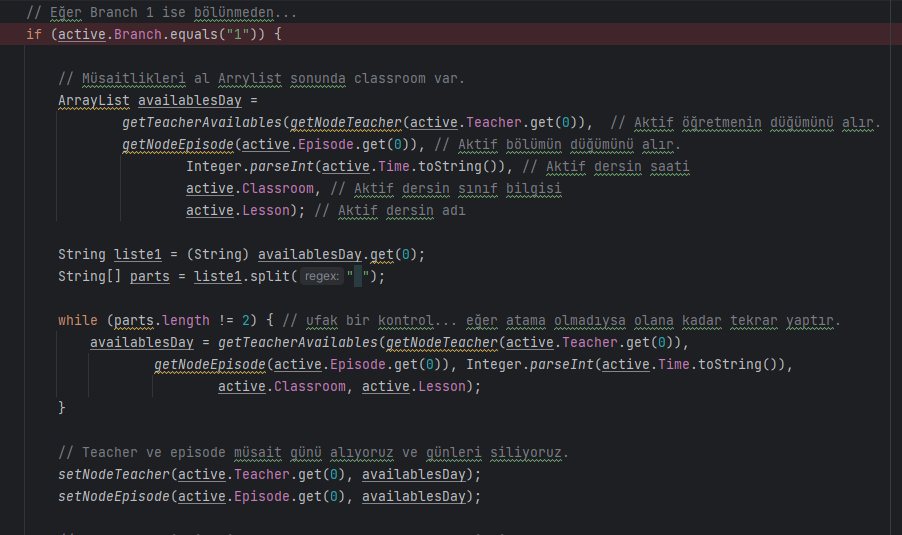
Açıklama otomatik olarak oluşturuldu

Nodelerimizi tamamen oluşturduktan sonra starlend menüsüne geçiyoruz…

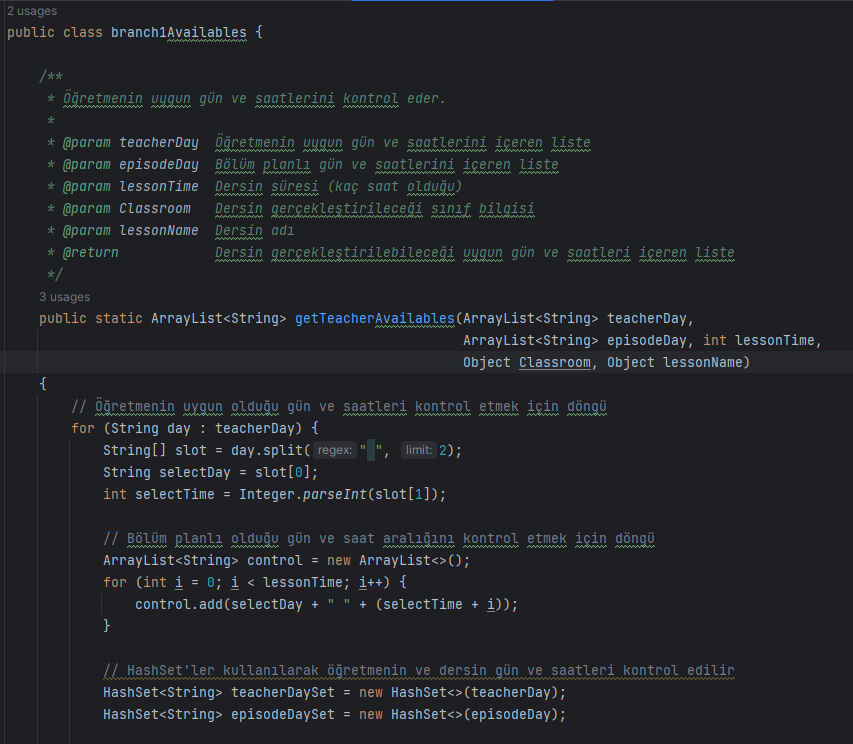


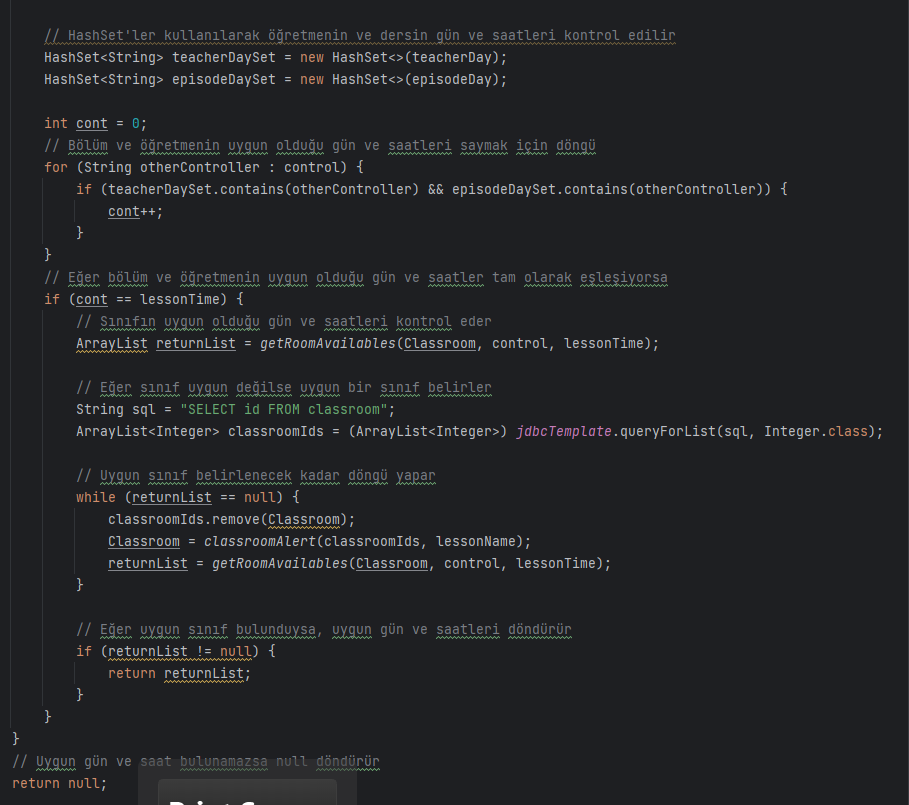


Şimdi müsait günleri alacağız…



getTeacherAvailables ile müsait günleri ve classroomları alıyoruz. Methodun içini acarsak eğer…



  
Bu şekilde methodumuz şimdi ise. … metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

setNodeEpisode ve setNodeTeacher ile o gün müsaitliğini siliyoruz…

metin, ekran görüntüsü, yazı tipi, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şimdi algoritmamıza devam edelim..

metin, ekran görüntüsü, yazı tipi içeren bir resim

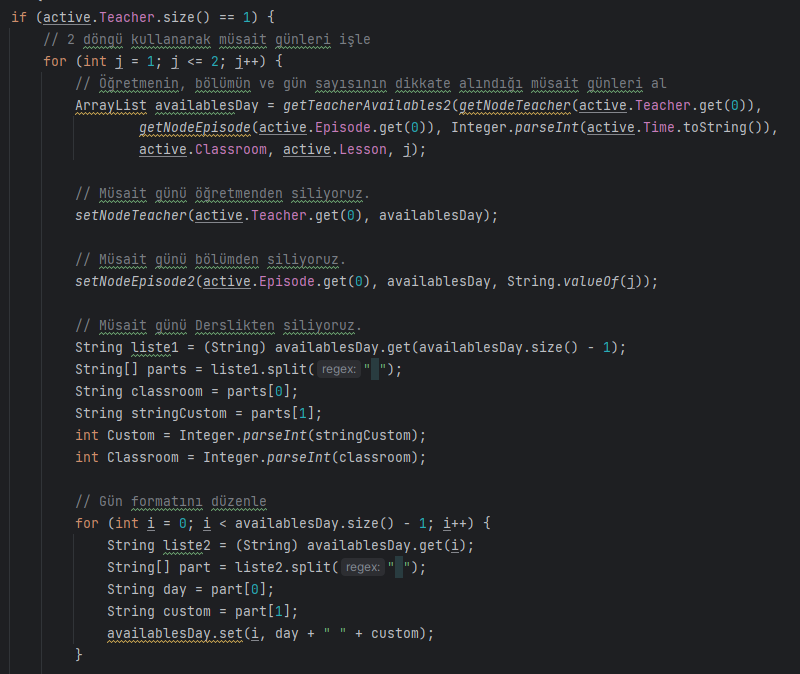
Açıklama otomatik olarak oluşturuldu  
  
Arrylistin sonunda classroom olduğu için arrylistin sonunu alıyoruz…. Ve parçalara ayırıp Derslik 2 diye alıyoruz…

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Kayıt işlemi bitmiştir…. Fakat eğer şubelere ayrılacaksa durum daha farklıdır şimdi bu durumda ne yapılacağına bakalım.

Eğer ders bölünecekse ve o derse tek 1 hoca aktarıldıysa…

2 Öğretmen yapılacağı zaman yaptiğimiz tek şey teacher.get ile arrylist sabit 0 olarak değil j değişkenine göre alcağız.. Sadece teacher get kısımlarını dikkatli bakın…

 metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

## Alogirtma Akış Diyagramı…

* 1. Bölümlerin okul acık olduğu tüm gün müsait olarak kayıt et.
  2. Derslikleri okul acık olduğu tüm gün müsait olarak kayıt et.
  3. Öğretmenlerin okul acık olduğu tüm gün müsait olarak kayıt et.
  4. Öğretmenlerde müsait olmadiği günleri sil.
  5. Derslerin gerekli bilgilerini al.
  6. Aldiğin bilgileri topla ve sırayla oku.
  7. Uygun öğretmen ve bölümü uygun saatlerini bul.
  8. Uygun Derslik bul.
  9. Uygun zamanı bölüm, öğretmen, derslik kayıt et.
  10. Bölüm ve öğretmen, derslik müsaitliğini sil o gün için