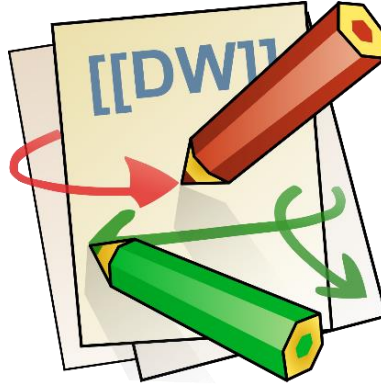


# BLM209 PROGRAMLAMA

## LABORATUVARI I

### PROJE 1

PROJE TESLİM TARİHİ: 21.10.2022



**PROJENİN ADI:** Dokuwiki Kocaeli Üniversitesi

**PROJENİN AMACI:** Dosya okuma, dosyaya yazma ve dosya güncelleme becerilerinin geliştirilmesi hedeflenmektedir.

**PROJENİN KONUSU:** Bir veri tabanı kullanmadan üniversitede bulunan bölümleri ve bölümlerde okutulan derslerin içeriklerini gösteren bir uygulama oluşturulması hedeflenmektedir.

#### Dosya İçeriği

İç içe klasörlerden oluşan bir *universite* adlı ana klasör bulunmaktadır. Ana klasör altında *bolumlar* ve *dersler* olmak üzere en az iki adet alt klasör bulunmaktadır. (Değerlendirme esnasında daha fazla klasör içeren örnek verilebilir. Arama işlemi tüm alt klasörlerde yapılmalıdır.) Alt klasörlerde bir veya daha fazla txt dosyası ve farklı uzantılarda (.docx, .config, .png, .gif vb.) birçok dosya bulunmaktadır. Her txt dosyasının içerisinde metin ve bu metinler içerisinde *etiket*ler bulunmaktadır. Bir etiket birden fazla txt dosyası içerisinde geçebilir.

- **Etiket formatı:**

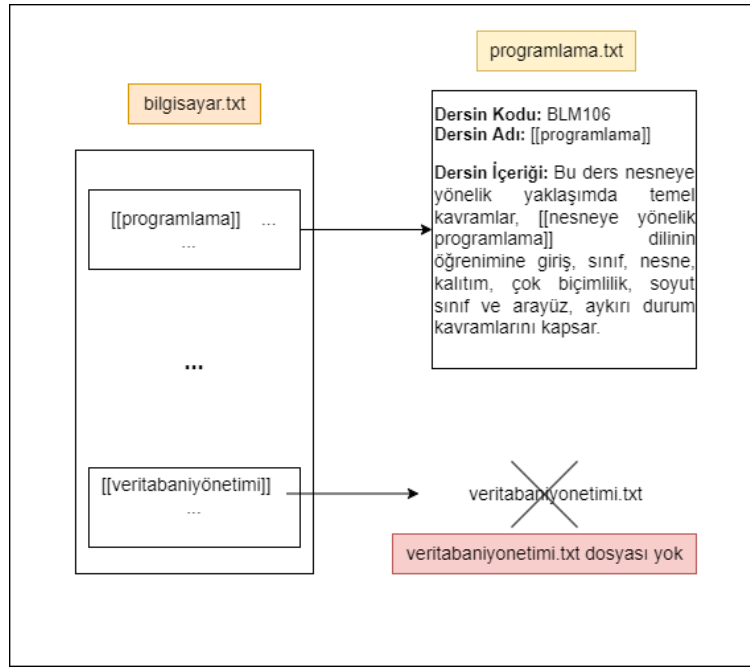
- Eğer bir kelime etiket ise formatı şu şekillerde olmalıdır:
  - [[kelime]], [[kelime1 kelime2]]
- Hatalı etiket formatları:
  - [[kelime], [kelime]], [kelime], {{kelime}} vb.

- **Yetim etiket:**

- Alt klasörlerde etiketlere ait txt dosyaları bulunmaktadır. Ancak her etikete ait txt dosyası olmak zorunda değildir. Aynı şekilde her txt dosyasının etiketi olmak zorunda değildir.
- Eğer bir etikete ait txt dosyası yoksa o etiket, *yetim etikettir*.

## **Ders Şablon İçeriği**

Her bir dersin bir kodu, adı ve içeriği olmalıdır. Tüm dersler bu şablona uygun olmalıdır. Örneğin bir ders şablonu Şekil 1'deki programlama.txt şeklinde olmalıdır.



Şekil 1: Örnek şablon

## **Arama (20 puan)**

Kullanıcı kelime ve etiket araması yapabilmelidir. Aranılan kelime etiket olabileceği gibi etiket olmayan bir kelime de olabilir. Arama fonksiyonu bu iki kontrolü de sağlamalıdır. Arama sonucunda hangi dosyada, hangi satır/satırlarda arama ifadesinin geçtiği ekranda yazdırılmalıdır. Ayrıca arama arayüzünde yetim etiketler ve istenen etiketler listelenmelidir.

**Yetim etiket:** Dosyası olmayan etiket

**İstenen etiket:** Dosya var fakat etiket yok. Dosya ismi veritabanı\_yonetimi.txt olduğunda istenen etiket : veritabanı yönetimi olmalıdır.

## **Güncelleme (20 puan)**

Kullanıcı bir etiketin adını değiştirebilir. Örneğin *veritabanıyönetimi.txt* dosyasında *[[veritabanıyönetimi]]* etiketi değiştirilip *[[veritabanı yönetimi]]* olduğunu varsayalım. Bu durumda yeni txt dosyasının adı da *veritabanı\_yonetimi.txt* şeklinde olmalıdır.

Yetim bir etiketin txt dosyası oluşturulurken, oluşturulan txt dosyası yukarıda belirtilen ders şablon içeriğine uygun olmalıdır. Burada Dersin Kodu 200'den başlayarak artarak atanmalıdır. Dersin Adına yetim etiket atanmalıdır. Dersin içeriği boş olmalıdır. Dolayısıyla artık yetim olmayan bu etiketin yetim etiketler listesinden silindiğinden emin olunmalıdır.

### **Dosyaya Yazma (20 puan)**

Ana klasörün içerisinde bir *output.txt* dosyası olmalıdır ve programın tüm çıktıları bu dosyaya yazdırılmalıdır. Çıktılarda tüm etiketler listelenmeli, toplamda kaç adet olduğu bilgisi yer almalıdır. Bunun yanı sıra yetim etiketlerin de listesi oluşturulmalıdır. Örnek output.txt Şekil 2'de verilmiştir.

```
output.txt - Not Defteri

Dosya   Düzenle   Görünüm

Etiket Listesi - Tekrar Sayısı
veritabaniyonetimi    3
yöneylem              1
programlama          5
matematik             9
tarih                 8

Yetim Etiketler
yöneylem
tarih

Güncel Liste
Etiket Listesi - Tekrar Sayısı
veritabani yönetimi   3
yöneylem              1
programlama          5
matematik             9
tarih                 8

Yetim Etiketler
yöneylem
```

Şekil 2: Örnek output.txt

### **Menü (Arayüz) (20 Puan)**

Tüm kontrollerin sağlanması için bir menü oluşturulmalıdır. Menüden kullanıcı arama yapabilmeli, etiketi ve o etikete ait txt dosyasının adını güncelleyebilmeli ve dosyaya yazma işlemi yapabilmelidir.

### **Rapor (10 Puan)**

#### **Yetim ve istenen etiketin nasıl bulunduğunu gösteren akış diyagramı (10 Puan)**

**NOT: Dosya okuma ve yazma işlemlerinde Türkçe karakterler de göz önünde bulundurulmalıdır.**

**NOT: Raporda projeye ait akış diyagramının dışında yetim ve istenen etiketin nasıl bulunduğunu gösteren akış diyagramı yer almalıdır.**

## ÖDEV TESLİMİ

- Proje **C veya C++ dilinde yazılmalıdır**. Bu diller dışındaki projeler değerlendirmeye alınmayacaktır. Ayrıca dosya işlemleri için **hazır kütüphane kullanımı yasaktır**.
- Ödevin raporu **LaTeX** kullanılarak yazılmalıdır.
- Rapor IEEE formatında (önceki yıllarda verilen formatta) 4 sayfa olmalı, akış diyagramı veya yalancı kod içermeli, özet, giriş, yöntem, deneysel sonuçlar, sonuç ve kaynakça bölümünden oluşmalıdır.
- Dersin takibi projenin teslimi dâhil edestek.kocaeli.edu.tr sistemi üzerinden yapılacaktır. edestek.kocaeli.edu.tr sitesinde belirtilen tarihten sonra getirilen projeler kabul edilmeyecektir.
- <https://tr.overleaf.com/latex/templates/ieee-conference-template/grfzhncsfqn> adresinden IEEE formatlı örnek metin yapısını bulabilirsiniz.
- Proje ile ilgili sorular edestek.kocaeli.edu.tr sitesindeki **forum üzerinden** Arş. Gör. Kübra Erat veya Arş. Gör. Dilara Görmez'e sorulabilir.
- Sunum tarihleri daha sonra duyurulacaktır.
- Sunum sırasında algoritma, geliştirdiğiniz kodun çeşitli kısımlarının ne amaçla yazıldığı ve geliştirme ortamı hakkında sorular sorulabilir.
- Kullandığınız herhangi bir satır kodu açıklamamız istenebilir.

**Sunum: Proje sunumu E-Destek üzerine yükleyeceğiniz projenizdeki kodlar indirilerek alınacaktır. Bu nedenle E- Destek üzerine yükleyeceğiniz projenin doğruluğundan emin olunuz.**