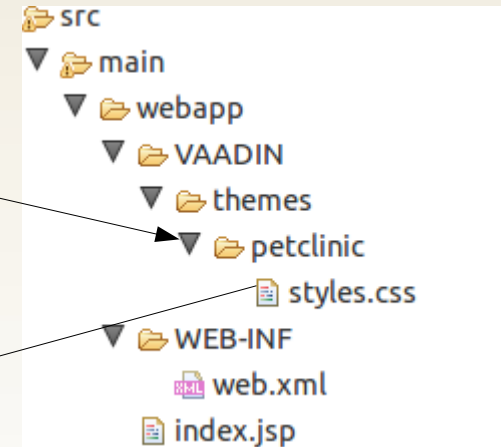


Custom Tema Oluşturma

Legacy CSS'ler ile Custom Tema Oluşturma

```
@Theme("petclinic")
public class PetClinicUI extends UI {

    @Override
    protected void init(VaadinRequest request) {
        ...
    }
}
```

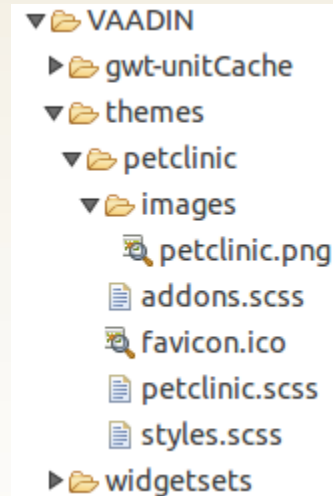


```
@import "../reindeer/legacy-styles.css";

.v-menubar {
    background: yellow;
}
```

SCSS ile Custom Tema Oluşturma

- **VAADIN/themes** dizini altında tema ismine karşılık gelen bir dizin olmalıdır
- Bu dizin altında **birden fazla SCSS dosyası** olabilir
- Başka **resource'lar** vs de olabilir



SCSS ile Custom Tema Oluşturma: 1. adım

import işlemi sadece valo tema dosyasındaki mixin'lerin tanımlanmasını sağlar. Import edilen mixin'deki hiçbir kural bu aşamada custom temaya dahil değildir.

```
@import "../valo/valo.scss";  
  
@mixin petclinic {  
  @include valo;  
  .v-menubar {  
    background: yellow;  
  }  
}
```

Birinci adımda uygulamaya özel kuralların tanımlandığı SCSS mixin(ler)i oluşturulmalıdır.

Bu mixin(ler) içerisinde ayrıca built-in temaların kural tanımları da import edilebilir.

include işlemi ile **valo** temasındaki bütün kurallar petclinic temasında da kullanılabilir hale getirilir.

Petclinic temasında yer alacak kurallar da burada tanımlanabilir.

SCSS ile Custom Tema Oluşturma: 2. adım

```
@import "petclinic.scss";  
  
.petclinic {  
  @include petclinic;  
}
```

import işlemi ile petclinic.scss içerisindeki mixin(ler) buraya tanıtılmış olur.

İkinci adımda **styles.scss** dosyası oluşturularak içerisinde ilk adımda oluşturulmuş SCSS mixin(ler)indeki tanımlı kurallar eklenir.

include ile **petclinic** mixin'i buraya eklenir. Böylece bu mixin'deki kural tanımları da buraya eklenmiş olur.

Bu tema dosyası içerisine eklenen bütün kurallar **temanın ismi ile prefix**'lenmiş olurlar. Böylece farklı temaların kurallarının birbirleri ile çakışmalarının önüne geçilir.

SASS Temalarının Derlenmesi: 3. adım

- **SASS** kuralları **CSS**'e dönüştürüldükten sonra kullanılabilir
- Derleme uygulama **çalıştırılmadan önce** yapılır
- Eclipse vaadin plugin'i veya **vaadin-maven-plugin**'i ile gerçekleştirilebilir
- Derleme sonucu VAADIN/themes/<tema> dizini altında **styles.css** dosyası oluşur

SASS Temalarının Derlenmesi: 3. adım

- Derleme istenirse **runtime'da** da yapılabilir
- Runtime'da yapılabilmesi için uygulamanın **development modunda** olması ve **vaadin-theme-compiler** ile bağımlılıklarının da classpath'de olması gerekir
- Runtime'daki derlemede **tarayıcı styles.css** dosyasını istediği zaman **VaadinServlet** derlemeyi kendisi gerçekleştirir

- Harezmi Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- info@java-egitimleri.com

