

Vaadin & CSS



Vaadin ve CSS



- Hemen hemen bütün görsel ihtiyaçlar
 <div> generic elemanı üzerinden
 gerçekleştirilebilir
- Vaadin de <div> elemanından yaygın biçimde yararlanmaktadır
- Dolayısı ile CSS kurallarında HTML eleman tipine göre selector kullanmak Vaadin için çok uygun değildir
- Vaadin ile çalışırken sadece class attribute'lar üzerinden CSS kuralları tanımlamak en doğrusudur

 www.javaegitimleri.com
 2

Vaadin ve CSS



- Her bileşen için o bileşenin isminin önüne
 "v-" öneki gelecek biçimde bir class selector ifadesi oluşturulur
- Örneğin Button için .v-button, Label için .vlabel vb
- Component.setStyleName(String)
 metodu ile custom style name
 tanımlanabilir
- Bir veya daha fazla style ismi String parametrede yer alabilir

Vaadin ve CSS



- Custom style ismi bileşenin HTML elemanında iki formda yer alacaktır
 - Verildiği biçimde (.mystyle)
 - Bileşenin built-in style ismi ön ek olarak (.vlabel-mystyle)
- addStyleName() ve removeStyleName() metotları ile de benzer biçimde bir veya daha fazla style ismi eklenebilir veya çıkarılabilir

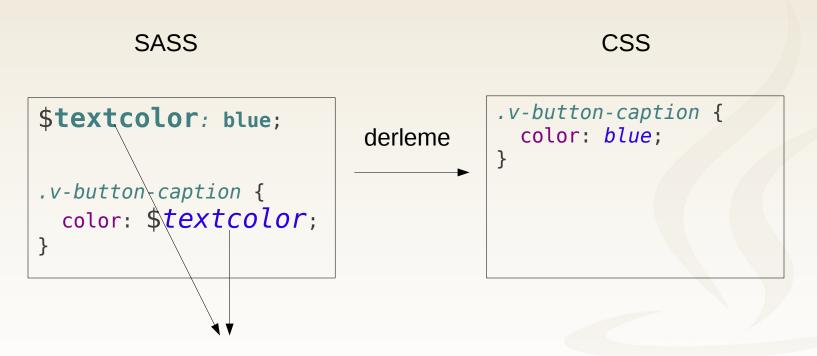
Syntactically Awesome Stylesheets (SASS)



- CSS3'ün bir extension'ıdır
- CSS'de mevcut olmayan nested kurallar, selector inheritance, değişkenler, mixin'ler gibi özellikler sunar
- Vaadin 7'den itibaren temalarda SASS kullanılmaktadır
- Vaadin ile çalışırken *.scss dosyaları oluşturulur, bunlar derlenerek styles.css dosyasına dönüştürürülerek kullanılır

SASS ve Değişkenler





Kurallarda kullanmak üzere değişkenler tanımlanabilir Böylece farklı yerlerde tekrar eden değerler ortaklanabilir

Ayrıca mixin'lerde de input parametre olarak değişkenler tanımlanabilmektedir

SASS ve Nested Kurallar



SASS

CSS

```
.v-app {
  background: yellow;

.mybutton {
  font-style: italic;

  .v-button-caption {
    color: blue;
  }
  }
}
```

derleme

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

.v-app .mybutton {
  font-style: italic;
}

.v-app .mybutton .v-button-caption {
  color: blue;
}
```

SASS ve Mixins



SASS

```
@mixin mymixin {
  background: yellow;
}

@mixin yourmixin($param) {
  margin: $param;
}

.v-button-caption {
  @include mymixin;
  @include yourmixin(10px);
}
```

CSS

derleme

```
.v-button-caption {
  background: yellow;
  margin: 10px;
}
```

Mixin'ler diğer kurallar içe<mark>risi</mark>ne gö<mark>müle</mark>bilen kurallardır.

@mixin ile tanımlanırlar.

@include ile diğer bir kural içerisine eklenebilirler

Input parametre alabilirler

Mixin içerisinde de nested kurallar tanımlanabilir

@StyleSheet



- Vaadin uygulamasına harici lokasyondan css kurallarının eklenmesini sağlar
- Uygulamanın kendi css kuralları herzaman için tema üzerinden yönetilmelidir
- Bu anotasyon daha çok custom client side
 UI bileşenlerinde kullanılır

```
@Theme("petclinic")
@StyleSheet("http://www.example.org/styles.css")
public class PetClinicUI extends UI {
...
}
```

İletişim



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

