



# CSS BOOTCAMP EĞİTİMİ

## 4 GÜN



**Digital Vizyon**  
Akademi

[www.digitalvizyon.net](http://www.digitalvizyon.net)



## İçindekiler

Eğitim Hakkında.....	4
Neler Öğreneceksiniz? .....	4
Ön Koşullar .....	5
Kimler Katılmalı.....	5
Outline .....	5
CSS Introduction .....	5
CSS Syntax.....	6
Selectors and Specificity .....	6
Feature Detection .....	6
The CSS Reset .....	7
CSS Custom Properties .....	7
CSS Typography .....	7
CSS 3 .....	7
CSS Layout .....	8
CSS Flexbox.....	8
CSS Grid Layout.....	8
CSS Multi-Column Layout .....	8
Introduction to CSS Optimization .....	9
Advanced CSS Optimization.....	9
Understanding CSS Rendering and Performance .....	9
CSS Introduction .....	10
CSS Syntax.....	10
Selectors and Specificity .....	10
Feature Detection.....	10
The CSS Reset .....	11
CSS Custom Properties .....	11
CSS Typography .....	11
CSS 3 .....	11
CSS Layout .....	12
CSS Flexbox.....	12
CSS Grid Layout.....	12
CSS Multi-Column Layout .....	13



Introduction to CSS Optimization .....	13
Advanced CSS Optimization.....	13
Understanding CSS Rendering and Performance .....	14

## Eđitim Hakkında

CSS Bootcamp Eđitimi, web geliştirme süreçlerinde stil ve tasarım odaklı yetenekler kazandırmak için tasarlanmış detaylı bir kurs programıdır. Bu eğitim, katılımcılara CSS'in temel ve ileri düzey kavramlarını öğreterek, estetik ve responsif web sayfaları tasarlamalarına yardımcı olur.

Eđitimin ilk bölümünde, katılımcılar CSS'in temel yapı taşlarıyla tanışır. CSS seçiciler, renkler, yazı tipleri ve kutu modeli gibi konseptlerle katılımcılar, web sayfalarına stil eklemeye başlar. Bu bölümde, sayfaların temel düzenini ve görünümünü nasıl değiştireceklerini öğrenirler.

İkinci bölümde, katılımcılar responsive web tasarımına odaklanır. Bununla beraber, medya sorguları, flexbox ve grid sistemleri gibi CSS özellikleri ile katılımcılar, farklı ekran boyutları ve cihazlara uyum sağlayabilen dinamik web sayfaları oluşturmayı öğrenir.

Üçüncü bölümde, CSS animasyonları ve geçişler ele alınır. Bu bölümde, kullanıcı etkileşimlerine yanıt olarak sayfa öğelerinin nasıl animasyonla hareket ettirileceğini ve estetik geçişlerle nasıl zenginleştirileceğini öğrenirler.

Dördüncü bölümde, katılımcılar CSS ön işlemcileri ile tanışır. Bununla birlikte SASS ve LESS gibi ön işlemciler sayesinde, CSS kodunun daha modüler olmasını sağlamak için ileri düzey teknikler öğrenirler.

CSS Bootcamp Eđitimi boyunca, katılımcılar gerçek dünya örnekleri üzerinden projeler geliştirir. Aynı zamanda, katılımcıların öğrendikleri teorik bilgileri pratiğe dökerek uygulamalı deneyim kazanmalarına olanak tanır.

CSS Bootcamp Eđitimi, web tasarımına ilgi duyanlar ve CSS becerilerini geliştirmek isteyenler için idealdir. Aynı zamanda web sayfalarının görünümünü ve kullanıcı deneyimini iyileştirme konusunda sağlam bir temel oluşturarak, katılımcılar bu alanda kariyer yapma potansiyelini artırabilirler.

## Neler Öğreneceksiniz?

CSS (Cascading Style Sheets) Bootcamp eğitimi, katılımcılara web sayfalarının görsel görünümünü nasıl düzenleyeceklerini öğretir. Eğitim süresi boyunca, şunları öğrenebilirsiniz:

- CSS seçicileri ve stil tanımları
- CSS boyutları, renkler ve tipografi
- CSS kutuları ve sıralama
- CSS responsive tasarım ve ekran boyutları
- CSS animasyonları ve geçiş efektleri
- CSS sınıf ve ID seçicileri
- CSS içeriği ve görüntüleri düzenleme

Bu eğitim, CSS'yi daha iyi anlamanıza ve web sayfalarının görsel görünümünü profesyonel bir şekilde düzenlemenize yardımcı olacaktır.

## Ön Koşullar

- HTML düzenleme becerileri: CSS'nin nasıl kullanılacağını anlamak için HTML'nin nasıl çalıştığını ve düzenlendiğini bilmeniz gerekir.
- Bilgisayar ve İnternet erişimi: Eğitim süresi boyunca kod yazmak ve çalıştırmak için bir bilgisayara ve İnternet'e ihtiyacınız olacak.
- Kod editörü: Eğitim süresi boyunca kod yazmanız ve çalıştırmanız gerekmektedir, bu nedenle bir kod editörüne ihtiyacınız olacak (örneğin: Sublime Text, Visual Studio Code).
- Temel programlama bilgisi: CSS, programlama dillerinden biridir ve bu nedenle programlama kavramlarını anlamanız gerekir.
- İngilizce bilgisi: Eğitim materyalleri ve belgeleme genellikle İngilizce olarak sunulur.
- İşletim sistemi bilgisi: Eğitim sırasında kullanacağınız işletim sistemi hakkında temel bilgilere sahip olmanız gerekir (örneğin Windows, macOS, Linux).

Bu ön koşullar, eğitimin daha verimli ve kolay bir şekilde tamamlanmasını sağlar, ancak eksikliği halinde de eğitimi tamamlamanız mümkündür.

## Kimler Katılmalı

CSS Bootcamp eğitimi, aşağıdaki kişiler için uygun olabilir:

- Web tasarımına ilgi duyanlar: CSS Bootcamp eğitimi, web sayfalarının görsel görünümünü düzenlemek isteyenler için idealdir.
- HTML kullanmakta olanlar: Eğitim HTML ile birlikte kullanılması gereken bir teknolojidir, bu nedenle HTML kullanmakta olanlar daha verimli bir şekilde eğitimi tamamlayabilir.
- Web geliştiricileri: CSS Bootcamp eğitimi, web sayfalarının daha profesyonel görünüm kazanmasını amaçlayan web geliştiricileri için ideal bir seçenektir.
- Başlangıç seviyesindeki programcılar: CSS Bootcamp eğitimi, programlama dillerine ve web geliştirmeye adım atmak isteyen başlangıç seviyesindeki programcılar için idealdir.

Eğitimi almak isteyen herkes, CSS Bootcamp eğitimine katılabilir ve CSS'yi daha iyi anlamaya başlayabilir.

## Outline

### CSS Introduction

- What is CSS?
- CSS Version Numbers and CSS Levels
- Who is responsible for CSS?



- Why should I be writing CSS?
- When was CSS created?
- Where is the official CSS specification?

## CSS Syntax

- Introduction to Writing CSS
- CSS @rules
- CSS functions
- CSS Scope
- Critical CSS
- CSS Keywords

## Selectors and Specificity

- Introduction to CSS Selectors and Specificity
- Helpful definitions
- Selectors
- Simple selectors
- Combinators
- Compound selectors
- Complex selectors
- The CSS “key”
- CSS Selector types
- Tag or element selectors
- Id Selectors
- Class selectors
- Specificity
- Specificity Hierarchy
- CSS Cascade Rules
- CSS Specificity Rules
- Logically-applied selectors

## Feature Detection

- Introduction to Feature Detection
- What is Modernizr?
- CSS Feature Queries
- CSS @supports
- supports
- Introduction to the CSS box model
- What is the CSS box model?
- CSS border Outline
- Block inline boxes
- CSS Float property
- Setting the box size



- Challenge Exercise: Using the CSS box model
- Z-Index and stacking orders
- CSS logical properties

## The CSS Reset

- Introduction to the CSS reset
- Why use a CSS reset?
- Why not use a CSS reset?

## CSS Custom Properties

- Introduction to CSS Custom Properties
- What are custom properties?
- How to declare a custom property
- Using a custom property
- Variable types
- Variable use cases
- Variable scope
- Using a custom property
- @property rule

## CSS Typography

- Introduction to Typography
- Units of Measurement
- Fixed units of measurement
- Relative units of measurement
- Absolute relative units
- Font Sizes and browser rendering

## CSS 3

- Introduction to CSS3
- What is CSS3?
- Understanding the CSS Version numbers and CSS Levels
- CSS3 Transforms
- What is a CSS transform?
- CSS3 2D transforms
- CSS3 3D transforms
- CSS transform: properties and methods
- CSS3 Transitions
- What is a CSS3 transition?
- How do I execute a CSS3 transition?
- CSS3 Animations
- CSS3 animation code
- CSS3 3D Animations



- 3D transformations

## CSS Layout

- Introduction to page layout with CSS
- Position property value
- position: static
- position: relative
- position: absolute
- position: fixed
- position: sticky
- CSS display property
- Introduction to the CSS display property
- Hiding elements
- Using the display property for layout
- Using Viewport units for layout
- Introduction to Viewport Units
- Viewport width
- Viewport height
- Viewport minimum and maximum

## CSS Flexbox

- Introduction to the display: flex property
- What is a flexbox?
- Creating a flexbox
- The flex layout model
- Flex display properties
- Flex Grow, Shrink, and Basis

## CSS Grid Layout

- Introduction to the CSS grid layout
- What is a grid?
- Features of a grid layout
- Grid glossary
- Placing items on the grid
- Grid properties

## CSS Multi-Column Layout

- Introduction to CSS columns
- Creating multi-column layout
- Column properties
- Column-related design issues





## Introduction to CSS Optimization

- CSS minification
- CSS color manipulation
- Use resets
- Sort CSS rules
- Write object-oriented CSS
- Make your CSS modular
- Use CSS properties and functions
- CSS calc()
- CSS inherit
- Use sprites
- Using an online minifier
- Add vendor prefixes
- Check code quality
- CSS lint
- Identify unused CSS
- Use an online CSS beautifier
- CSS code organization
- Code organization tips
- Use a CSS style guide
- Use a CSS preprocessor
- CSS loading techniques

## Advanced CSS Optimization

- Writing scalable, efficient, and maintainable CSS
- Naming Conventions
- Semantic CSS
- BEM
- OOCSS
- SMACSS
- Atomic
- Formatting CSS

## Understanding CSS Rendering and Performance

- Browser rendering and optimization terms
- Rendering stages
- Guided Exercise: Understanding the browser's rendering stages
- Optimizing CSS
- Optimize selectors
- Remove unused CSS
- Use testing tools
- Avoid unnecessary DOM manipulations



- Use a CDN
- Minimize repaint and reflows
- Paint cycles and compositor layers
- CSS and JavaScript
- More optimization tips

## CSS Introduction

- What is CSS?
- CSS Version Numbers and CSS Levels
- Who is responsible for CSS?
- Why should I be writing CSS?
- When was CSS created?
- Where is the official CSS specification?

## CSS Syntax

- Introduction to Writing CSS
- CSS @rules
- CSS functions
- CSS Scope
- Critical CSS
- CSS Keywords

## Selectors and Specificity

- Introduction to CSS Selectors and Specificity
- Helpful definitions
- Selectors
- Simple selectors
- Combinators
- Compound selectors
- Complex selectors
- The CSS “key”
- CSS Selector types
- Tag or element selectors
- Id Selectors
- Class selectors
- Specificity
- Specificity Hierarchy
- CSS Cascade Rules
- CSS Specificity Rules
- Logically-applied selectors

## Feature Detection

- Introduction to Feature Detection



- What is Modernizr?
- CSS Feature Queries
- CSS @supports
- supports
- Introduction to the CSS box model
- What is the CSS box model?
- CSS border Outline
- Block inline boxes
- CSS Float property
- Setting the box size
- Challenge Exercise: Using the CSS box model
- Z-Index and stacking orders
- CSS logical properties

## The CSS Reset

- Introduction to the CSS reset
- Why use a CSS reset?
- Why not use a CSS reset?

## CSS Custom Properties

- Introduction to CSS Custom Properties
- What are custom properties?
- How to declare a custom property
- Using a custom property
- Variable types
- Variable use cases
- Variable scope
- Using a custom property
- @property rule

## CSS Typography

- Introduction to Typography
- Units of Measurement
- Fixed units of measurement
- Relative units of measurement
- Absolute relative units
- Font Sizes and browser rendering

## CSS 3

- Introduction to CSS3
- What is CSS3?
- Understanding the CSS Version numbers and CSS Levels
- CSS3 Transforms



- What is a CSS transform?
- CSS3 2D transforms
- CSS3 3D transforms
- CSS transform: properties and methods
- CSS3 Transitions
- What is a CSS3 transition?
- How do I execute a CSS3 transition?
- CSS3 Animations
- CSS3 animation code
- CSS3 3D Animations
- 3D transformations

## CSS Layout

- Introduction to page layout with CSS
- Position property value
- position: static
- position: relative
- position: absolute
- position: fixed
- position: sticky
- CSS display property
- Introduction to the CSS display property
- Hiding elements
- Using the display property for layout
- Using Viewport units for layout
- Introduction to Viewport Units
- Viewport width
- Viewport height
- Viewport minimum and maximum

## CSS Flexbox

- Introduction to the display: flex property
- What is a flexbox?
- Creating a flexbox
- The flex layout model
- Flex display properties
- Flex Grow, Shrink, and Basis

## CSS Grid Layout

- Introduction to the CSS grid layout
- What is a grid?



- Features of a grid layout
- Grid glossary
- Placing items on the grid
- Grid properties

## CSS Multi-Column Layout

- Introduction to CSS columns
- Creating multi-column layout
- Column properties
- Column-related design issues

## Introduction to CSS Optimization

- CSS minification
- CSS color manipulation
- Use resets
- Sort CSS rules
- Write object-oriented CSS
- Make your CSS modular
- Use CSS properties and functions
- CSS calc()
- CSS inherit
- Use sprites
- Using an online minifier
- Add vendor prefixes
- Check code quality
- CSS lint
- Identify unused CSS
- Use an online CSS beautifier
- CSS code organization
- Code organization tips
- Use a CSS style guide
- Use a CSS preprocessor
- CSS loading techniques

## Advanced CSS Optimization

- Writing scalable, efficient, and maintainable CSS
- Naming Conventions
- Semantic CSS
- BEM
- OOCSS



- SMACSS
- Atomic
- Formatting CSS

## Understanding CSS Rendering and Performance

- Browser rendering and optimization terms
- Rendering stages
- Guided Exercise: Understanding the browser's rendering stages
- Optimizing CSS
- Optimize selectors
- Remove unused CSS
- Use testing tools
- Avoid unnecessary DOM manipulations
- Use a CDN
- Minimize repaint and reflows
- Paint cycles and compositor layers
- CSS and JavaScript
- More optimization tips