Scss kodlarını parçalar halinde ,mantıksal isimlendirmeler vererek oluşturuyoruz.mesela utilities yardımcı sınıf olarak kullanılıyor ve içerisine,programın değişik yerlerinde birden fazla kullanmamız gereken buton,marjin,padding,container gibi parçacıkların kodları yazılıyor.

------------------------------------------------------------------------------------------------

Base scss dosyasına etiketlerin özelliklerini veriyoruz.Eğer sayfadaki değişik yerlerde dahi kullanılsa bir etiketin sahip olmasını istediğimiz özelliği değişmiyorsa base yani temel dosyasında tanımlayabiliriz.

------------------------------------------------------------------------------------------------

Sass kodlama özelliklerinden extend yapısını % işareti kullanarak isimlendiririz.Extend yapısı içinde tanımladığımız kodları dahil etmek istediğimiz etiketin içerisine @extend %isim şeklinde sahil ederiz.

Örnek;

%btn-common{

Padding:10px;

Margin:5 px;

Width:100px;

Height:100px;

}

.btn-primary{

@extend %btn-common;

Color:White;}

------------------------------------------------------------------------------------------------

button[class^='btn-'] sınıf ismi btn- ile başlayan tüm button etiketlerini seç

------------------------------------------------------------------------------------------------

.bg-{

&primary{

Color:blue;}

}

bg-primary isimli bir class oluşturup yazı rengini mavi yap

------------------------------------------------------------------------------------------------

**Navigation bar oluşturma**

Öncelikle bir container oluşturup içerisine nav etiketini açıyoruz.

Nav etiketine bir class atıyoruz.Bu genellikle main-nav oluyor.

Nav içerisine ilk olarak logoyu yerleştiriyoruz.img etiketini kullanarak,main-nav\_\_logo classı atayabiliriz.

Yine nav etiketi içerisinde iken ,oluşturacağımız menünün elemanlarını içerisinde sıralı bir şekilde tutacak olan sırasız bir listeyi ul etiketi ve main-nav\_\_items sınıfı tanımlayarak oluşturuyoruz.

Ul etiketi içerisine elemanları li etiketi ile eklemeye başlıyoruz.elemanları link olarak kullanacağımız için li içerisine bunları a etiketi yardımıyla linkliyoruz.

Oluşturduğumuz linklerden biri default seçili gelmesi için class active ekleriz.

Header içerisine hem menü hem content eklemek istiyorsak,aynı container içerisinde menüyü nav etiketi arasına hazırlayıp,content bilgisini navden ayrı bir div içerisine fakat aynı container içerisine koymalıyız.

------------------------------------------------------------------------------------------------

Bir span etiketi içerisine javascript ve animation kullanarak bir yazı yazdırmak

Script kodlarını body kapanış etiketinin hemen üstüne açıp,yapmasını istediğimiz işlemi gerçekleştirecek kütüphaneyi buraya ekliyoruz.

Animate.style üzerinden bir animasyonu kullanmak istiyorsak önce kütüphanesini link olarak projeye dahil ediyoruz.Kullanmak istediğiniz elamanın classına seçtiğiniz animasyon efektinin adını ve animate\_\_animated sınıflarını

ekliyoruz.

------------------------------------------------------------------------------------------------

url içinde bir üst klasöre çıkmak için ../ kullanılır.

------------------------------------------------------------------------------------------------

Logo ve menüyü yanyana almak için main-nav classını flex tanımlayıp,justify content özelliğini space between yaparak aralarında boşuk bırakıp birini sağa birini sola yaslarız.

------------------------------------------------------------------------------------------------

H1 ve span alt alta gelsin istiyorsak span display : block yapılmalıdır

a nın .active classı üzerine gelmeden önceki durumunu belirler

background-size özellikleri:

cover:resim en-boy oranını koruyarak,kutunun genişliğini dolduracak şekilde resmi kırpar.

contain:En boy oranını korurlen,kutuyu doldurmak için resmin boyutunu büyültür ya da küçüktür

fill:en-boy oranına bakılmaksızın görüntüyü büyütür ya da küçültür.

Bacground-position : center / center top /center right /

------------------------------------------------------------------------------------------------

**İntroduction bölümü**

Header sectionını ,content ve menü ekleyerek tamamladık.sayfanın diğer sectionlarını hazırlıyoruz.isimlendirme kolaylığı için a ve b şeklinde isimlendireceğiz.

Section home\_\_a isimli bölümü oluşturup , yazıları ortalamak için text-center classı tanımlıypruz ve padding değerini py-2 olarak önceden tanımladığımız class ismini veriyoruz.

Section\_\_a içerisine bir container koyarak işlemlerimize başlıyoruz.container sınıfını daha önce utilities klasörü içerisin padding değerlerini vererek tanımlamıştık.üstten alttan 0 ve sağdan soldan 1.4 rem boşluk ayrıca margin değeride auto verilmiştir.

İntroduction başlık bölümü için section-title classına sahip bir h2 etiketi tanımladık.section-title classına utilities içerisinde özellik atadık çünkü bu class daha sonra kullanacağız yeniden.

Başlık altındaki kırmızı çizgiyi oluşturmak için bir div tanımladık border-bottom classına sahip ve bu class ıda utilities içerisinde tanımladık çünkü daha sonra kullanacağız.

Note: Sass css önişlemcisidir.npm (node packet manager) yardımıyla css kodlarına dönüştürülür.npm kullanmak için node.js bilgisayarda kurulu olmalıdır.

Word-spacing:2px; kelimeler arasındaki boşluğu artırır.

Skills bölümü oluşturmak için yanyana 4 div içerisine ikonlar ve yazıyı ekleyip,bunları grid sisteme göre tanımladık

Grid sistemi home.scss içerisinde tanımlıyoruz.çünkü sayfanın headerdan sonraki sectionını home olarak tanımlıyoruz.

Home\_\_a id sine ulaşıp altındaki skills sınıfını grid tanımlıyoruz.

Grid-template-colums : repeat(4,1fr) olarak eşit genişlikte 4 kolona bölüyoruz.

Gap:2 rem ; diyerek kolonlar arası boşluk belirledik.

Note: CDN bir sunucu kümesidir.Farklı farklı ülkelere konumlanmıştır.Amaç web içeriğini kullanıcının konumuna göre ona en hızlı şekilde ulaştırmaktır.

Note:fontawesome icon boyutu büyütmek için class a fa-3x verilir

Note:Sass kodu yazarken vs-code da watch sass yaparak sass kodlarını css kodlarına çevirmeyi unutma

Note: Sass yazarken etiketlere özellik verirken class gibi başına . işareti koyma

------------------------------------------------------------------------------------------------

Footer

Divi yuvarlak yapmak için border-radius:50% verilir.

Hovera geçiş değeri vermek için transition:all 1s verilir.

Sass yazarak hover bilgisi içerisinde başka bir sınıfın özelliklerine müdahale ettik.

   &:hover{

                    background: $primary-color;

                    .process\_\_step{

                        background: $dark-color;

                    }

                }

4 sayfalı bir tasarı yaptık.her sayfanın scss dosyasını,sayfa adıyla aynı olacak şekilde verip,sayfaya ait sınıfları içerisine tanımladık.utilities ve main scss ler bütün saylarda ortak kullanılan sınıfları içerdi.

Sayfaya javascript Carousel ekleme

Owlcarousel sitesine gidip basic bir Carousel seçtik ve sınıf isimlerini carouselimizi kapsayan div in içerisine yazdık

------------------------------------------------------------------------------------------------

RESPONSİVE NOTES: