A tag <meta> pode ser configurada para várias configurações do HTML. No momento queremos o 'viewport', ou seja, configurar a tela que o usuário vê. Isso para que ela se comporte seguindo o tamanho do dispositivo simulado pela tela de inspecionar elemento. O initial-scale é para abrir a tela sempre no zoom padrão

<meta name="viewport" content="width=device-width, initial-scale=1">

Para deixar a imagem responsiva, usar a tag abaixo na imagem

<img class="img-responsive col-sm-6" src="assets/img/empresa.jpg" alt="">

Para implementar o calapse, basta usar o data-toggle colapse indicando que a div que sera clicada sera um calapse, o data-target serve para indicar o que sera aberto ou fechado, nesse caso sera o id primeiro paragrafo. O callapse in serve para indicar qual ficara aberto por padrão.

<div class="panel-heading" data-toggle="collapse" data-target="#primeiro-paragrafo" data-parent="#paineis-sobre">

                        <h3 class="panel-title">Desde 1935</h3>

                    </div>

                    <div class="collapse in" id="primeiro-paragrafo">

                        <div class="panel-body">

                            Desde 1935 trazendo casas, mansões e prédios lindos para o mundo

                        </div>

                    </div>

Para criar um layout como o do exemplo do curso, precisamos do painel, segue código abaixo:

<div class="panel-group>

<div class="panel panel-default">

                    <div class="panel-heading">

                        Cabeçalho do painel

                    </div>

                    <div>

                        <div class="panel-body">

Corpo do painel

                        </div>

                    </div>

</div>

</div>

O painel-group é para agrupar mais de um painel.

Para poder fechar todos os painéis ao abrir um novo é necessário usar o js, mais especificamente o data-parent="#id-do-grupo de paineis ">. Esse cara vai ser o pai dos painéis e sera por ele que o js ira identificar e fechar os outros painéis ao selecionarmos algum deles

<div class="panel-group col-sm-6" id="paineis-sobre">

                <div class="panel panel-default">

                    <div class="panel-heading" data-toggle="collapse" data-target="#primeiro-paragrafo" data-parent="#paineis-sobre">

                        <h3 class="panel-title">Desde 1935</h3>

                    </div>

                    <div class="collapse in" id="primeiro-paragrafo">

                        <div class="panel-body">

                            Desde 1935 trazendo casas, mansões e prédios lindos para o mundo

                        </div>

                    </div>

                </div>

A classe thumbnnail serve para pode colocar imagens como titulo na view

<figure class="thumbnail">

                    <img src="assets/img/projetos/casa-castelo.png" alt="">

                    <figcaption class="caption">

                        <h3>Casa Castelo</h3>

                        <p>

                            A Casa Castelo é o último lançamento da TopCasaFina Arquitetura, feita para um monge.

                        </p>

                    </figcaption>

                </figure>

É como se fosse um banner cinza com algum texto para separar seções por exemplo

<div class="jumbotron">

        <div class="container">

            <h3>Mais de 300 prêmios em design e em conforto.</h3>

            <p>Mais de 5 milhões de clientes satisfeitos em todo o mundo.</p>

        </div>

    </div>

**Lembrando que:**

* Na tab components podemos achar componentes reutilizáveis criados pelo bootstrap.
* Na tab CSS temos configurações globais do CSS e elementos HTML fundamentais estilizados pelo CSS. Também temos o sistema de grid que será abordado mais à frente.
* Na tab JS temos components que utilizam javascript e também a documentação de javascript do bootstrap.

Para colocar um carrossel na pagina, basta usar o código abaixo:

<div id="carousel-example-generic" class="carousel slide" data-ride="carousel">

        <!-- Indicators -->

        <ol class="carousel-indicators">

         <li data-target="#carousel-example-generic" data-slide-to="0" class="active"></li>

         <li data-target="#carousel-example-generic" data-slide-to="1"></li>

         <li data-target="#carousel-example-generic" data-slide-to="2"></li>

        </ol>

        <!-- Wrapper for slides -->

        <div class="carousel-inner" role="listbox">

            <div class="item active">

                <img src="..." alt="...">

                <div class="carousel-caption">

                ...

                </div>

            </div>

            <div class="item">

                <img src="..." alt="...">

                <div class="carousel-caption">

                ...

                </div>

            </div>

            ...

            </div>

            <!-- Controls -->

            <a class="left carousel-control" href="#carousel-example-generic" role="button" data-slide="prev">

            <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>

            <span class="sr-only">Previous</span>

            </a>

            <a class="right carousel-control" href="#carousel-example-generic" role="button" data-slide="next">

            <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>

            <span class="sr-only">Next</span>

            </a>

        </div>

Exemplo para colocar vídeo responsivo:

<div class="embed-responsive embed-responsive-16by9">

    <iframe width="560" height="315" src="https://www.youtube.com/embed/i\_3Pv1AygeI" frameborder="0" allowfullscreen></iframe>

</div>

Linkar texto com campo input

<label for="contato-nome">Nome:</label>

                        <input id="contato-nome" class="form-control" type="text" placeholder="Seu nome">

Colocar algo a mais no começo do input quando necessário

<div class="input-group">

                            <div class="input-group-addon">@</div>

                       <input id="contato-email" class="form-control" type="email" placeholder="Seu e-mail">

                   </div>

Para fazer um select deixando desabilitado alguma opção mas inicializando com ela

<select class="contato-select form-control" name="tipo-solicitacao">

                        <option value="" disabled selected>Tipo de solicitação</option>

                        <option value="casa">Casa</option>

                       <option value="apartamento">Apartamento</option>

                       <option value="mansao">Mansão</option>

                    </select>

Existe um bug que é necessário um espaço em branco(comentário entre dois inline-block)

Para pegar o primeiro elemento encontrado

.grupo-radio .radio:first-child {

Pegar o ultimo elemento encontrado

footer address:last-child {

    margin-bottom: 0;

}

Menu escondendo

<nav class="navbar navbar-default">

            <div class="container">

                <div class="navbar-header">

                    <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#collapse-navbar" aria-expanded="false">

                        <span class="sr-only">Toggle navigation</span>

                        <span class="icon-bar"></span>

                        <span class="icon-bar"></span>

                        <span class="icon-bar"></span>

                    </button>

                    <a class="navbar-brand" href="#">TopCasaFina Arquitetura</a>

                </div>

                <div class="collapse navbar-collapse" id="collapse-navbar">

                    <ul class="nav navbar-nav">

                        <li><a href="#sobre-nos">Sobre Nós</a></li>

                        <li><a href="#nossos-projetos">Nossos Projetos</a></li>

                        <li><a href="#depoimentos">Depoimentos</a></li>

                        <li><a href="#contato">Contato</a></li>

                    </ul>

                </div>

            </div>

        </nav>

A .container define um tamanho fixo definido pelo Bootstrap para cada tamanho de tela. A .container-fluid pega o tamanho de 100% diferentemente da .container.

header {

    background-image: url(../img/castelo.jpg);

    background-size: cover;

    background-position: top center;

    height: 100%;

}

rgbA – O A é de alpha ou opacidade

background-color: rgba(0, 0, 0, .5);

Observe como está o banner:



Temos que centralizá-lo e recortar a borda para que ela fique com um desenho mais interessante. Para isso, vamos adicionar o border-radius que serve para justamente arredondar as bordas, adicionaremos as medidas de "8 px":

.topCasaFina-banner {

color: #FFF;

background-color: rgba(0, 0, 0, 0.5);

padding: 20px;

width: 80%;

border-radius: 8px;

}

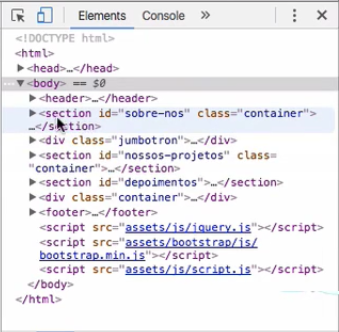
Teremos o seguinte:



Já conseguimos deixar as bordas arredondadas e deixar o "banner" bem próximo ao que desejamos. Vamos posicionar nosso item na tela, o ideal é que ele esteja centralizado. Nesse instante ele está distribuído da seguinte maneira:



Vamos observar o "Console", a aba "Expect Elements", para entendermos com relação a quem desejamos que o "banner" se situe.



Dentro do "header" temos o "banner" e queremos que ele fique no centro do "header". Ou seja, temos que dizer que o "banner" deve se posicionar em relação ao "header". Para fazer isso temos que alterar o position dele no "estilo.css". Vamos usar o position: absolute:

.topCasaFina-banner {

color: #FFF;

background-color: rgba(0, 0, 0, 0.5);

padding: 20px;

width: 80%;

border-radius: 8px;

position: absolute;

}

Se dermos um "reload", não teremos nenhuma mudança em nossa página, temos que passar o posicionamento em relação ao eixo vertical e eixo horizontal. Podemos observar isso através do expect elements. Se o header todo é 100%, o seu meio é 50%, então, podemos acrescentar top: 50%:



Podemos perceber que a imagem não está exatamente no meio, o seu topo está no meio, fazendo com que o restante da imagem fique mais para baixo. Temos que deslocar isso para cima, a grande questão é: o quanto temos que deslocar?

Observando podemos intuir que visualmente queremos que fique 50% mais para cima. Vamos deixar de pensar o posicionamento da caixa em relação a header, vamos pensar em seu posicionamento em relação a caixa mesmo, vamos utilizar o transform: translate e como queremos que o deslocamento seja na vertical vamos usar o eixo Y e como queremos voltar metade da caixa, vamos usar o -50%. Queremos também que ele fique com 50% no top:

top: 50%;

transform: translateY(-50%);

Teremos:

.topCasaFina-banner {

color: #FFF;

background-color: rgba(0, 0, 0, 0.5);

padding: 20px;

width: 80%;

border-radius: 8px;

position: absolute;

top: 50%;

transform: translateY(-50%);

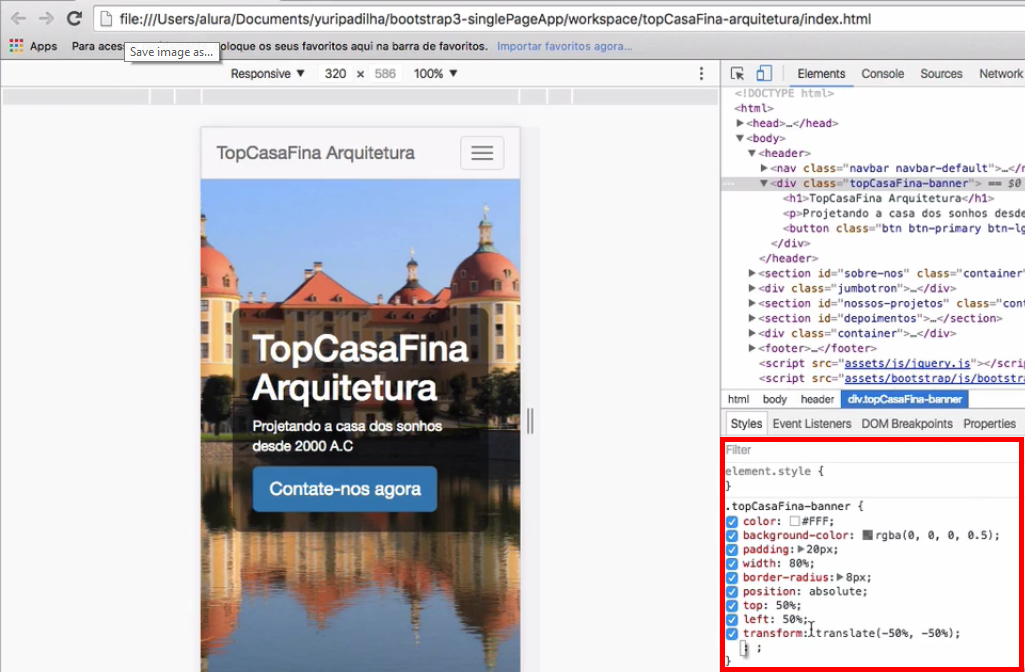
}

E dando reload ficaremos com:



Agora, o que falta é ficar apenas um pouco mais para a esquerda e para fazer isso, vamos pensar, se passarmos um left, por exemplo ele irá se deslocar completamente a partir de sua borda. Vamos passar para o transform as duas referências, não apenas uma que marque o eixo "X":

Observe o que temos:



E assim, ele fica alinhado conforme queríamos. Escrevemos tudo no "Expect elements", então, precisamos passar isso para o "estilo.css":

.topCasaFina-banner {

color: #FFF;

background-color: rgba(0, 0, 0, 0.5);

padding: 20px;

width: 80%;

border-radius: 8px;

position: absolute;

top: 50% ;

left: 50%;

transform: translate(-50%, -50%);

}

Temos a imagem posicionada da maneira que queríamos, a única questão é que o elemento quando é position absolute ele procura se posicionar em relação ao "pai" que é relativo, por isso, temos que acrescentar no header que ele é position relative:

header {

background-image: url(img/castelo.jpg)

background-size: cover;

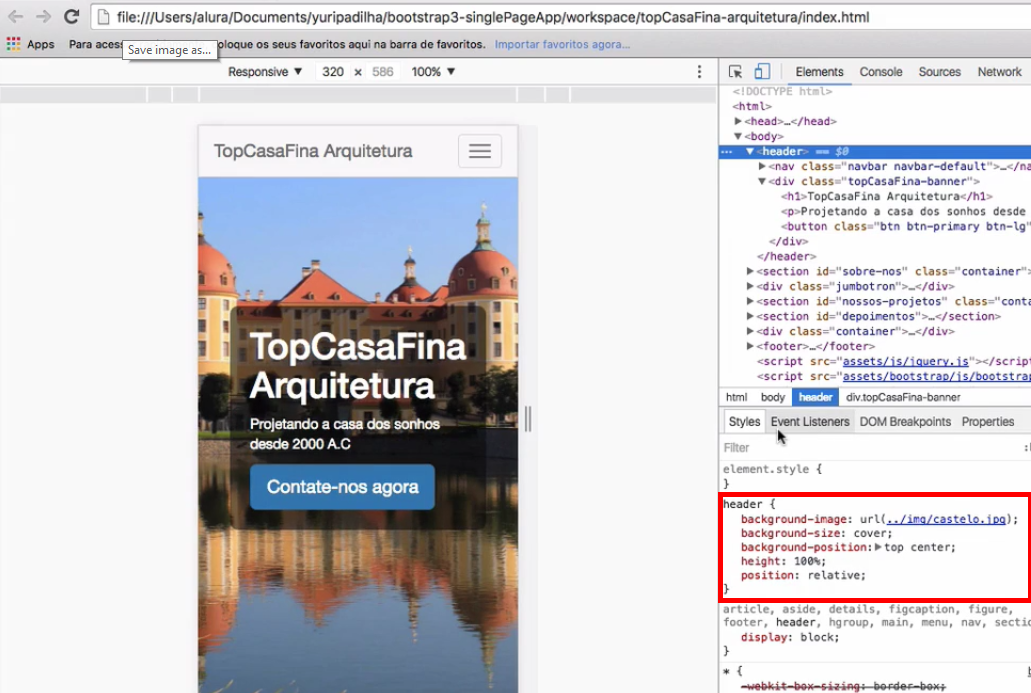
background-position: top-center;

height: 100%;

position: relative;

}

E teremos já nossa imagem da maneira como queríamos!



Evento ao esconder o menu

$('#collapse-navbar').on('hide.bs.collapse', function() {

    $('.topCasaFina-banner').css('transform', 'translate(-50%, -50%');

});

Imagens decorativas são inseridas através do CSS pela propriedade background. Imagens de conteúdo são inseridas através do HTML pela tag <img>.

Lembre-se de colocar position: relative no pai correto para que a chamada com absolute se posicione conforme esse pai.

Devemos primeiramente colocar position:relative no <header> para que a nossa chamada se posicione com relação à ele quando colocarmos position:absolute nela.

Depois de colocar position:relative no pai e position:absolute no filho devemos usar as propriedades left e top para colocar a chamada no lugar correto.