

Nome:

Trivia How I Met Your Mother

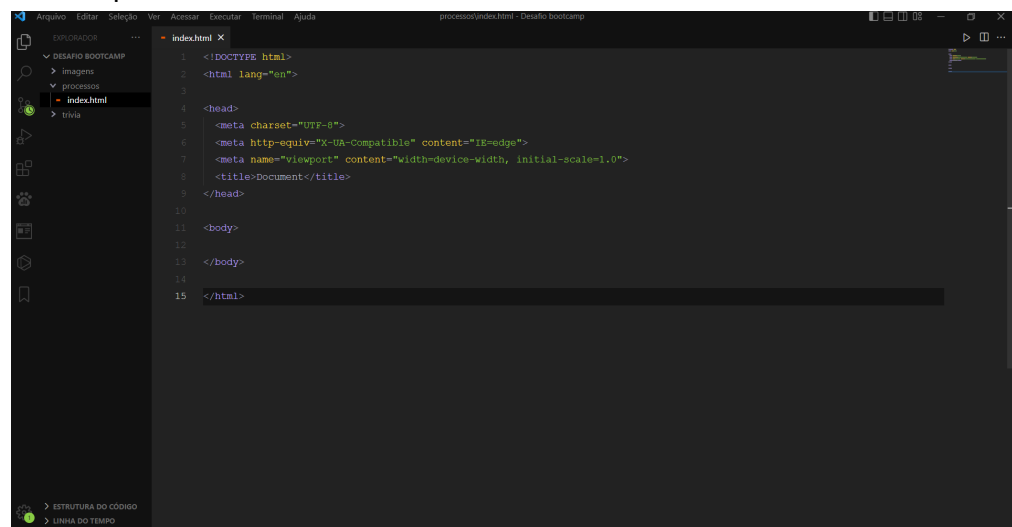
Descrição:

Este é um projeto que tem um quiz perguntas e respostas e possui duas telas.

- Recursos tela inicial: É onde o usuário coloca seu nome e tem uma mensagem de boas vindas e logo após isso pode clicar em um botão que permite que ele vá para a segunda tela.
- Recursos tela dois: Na segunda tela há o título da série que o trivia foi baseado e uma breve explicação de como funciona a pontuação de respostas corretas e incorretas. Abaixo do parágrafo de explicação, tem as respectivas perguntas com suas respectivas respostas e embaixo tem um botão de responder e ver resultado para o usuário ver qual foi sua pontuação e saber se acertou ou errou. Abaixo de todas as perguntas tem um botão de jogar novamente que leva o usuário a tela inicial novamente.

Processos e design de experiência do usuário:

1. A primeira etapa foi criar um arquivo index.html onde possuía um esboço inicial e padronizado do html:5

A screenshot of a code editor interface, likely Visual Studio Code, showing the initial HTML structure of a file named 'index.html'. The editor has a dark theme. On the left, there is a file explorer sidebar showing a project structure with folders like 'DESAFIO BOOTCAMP', 'imagens', 'processos', and 'trivia'. The 'index.html' file is selected under the 'trivia' folder. The main editor area displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <title>Document</title>
9 </head>
10
11 <body>
12
13 </body>
14
15 </html>
```

- Logo após isso foi adicionado ao arquivo index.html as tags de título e parágrafo (parágrafo contendo as perguntas do trivia)

```
index.html X
6 <meta http-equiv="X-UA-Compatible" content="IE=edge">
7 <meta name="viewport" content="width=device-width, initial-scale=1.0">
8 <title>Document</title>
9 </head>
10
11 <body>
12 <h1>Trivia de How I Met Your Mother</h1>
13 <div>
14 <p>Questão 1: Onde os personagens Lily e Marshall se conheceram?</p>
15 </div>
16 <div>
17 <p>Questão 2: Qual é o bar predileto dos personagens?</p>
18 </div>
19 </body>
20
21 </html>
```

imagem: Entrada

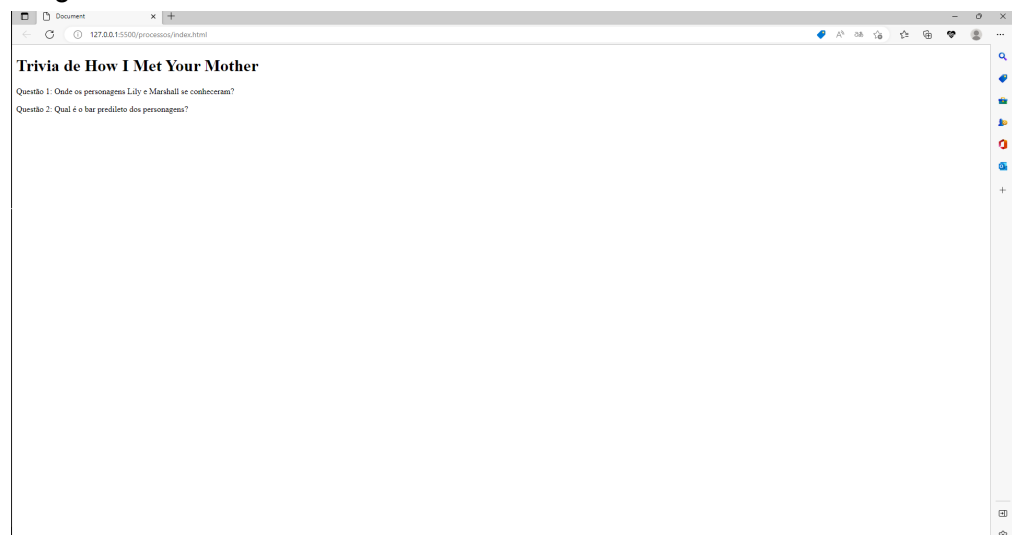


imagem 2: Saída

2. Na terceira parte foram feitas as alternativas corretas e incorretas das perguntas citadas acima, nesse momento foi usado o input type radio para selecionar as alternativas, o label for para escrever as alternativas e foi usado a tag html
 para fazer quebra de linha e ficar visualmente mais

agradável ao usuário.

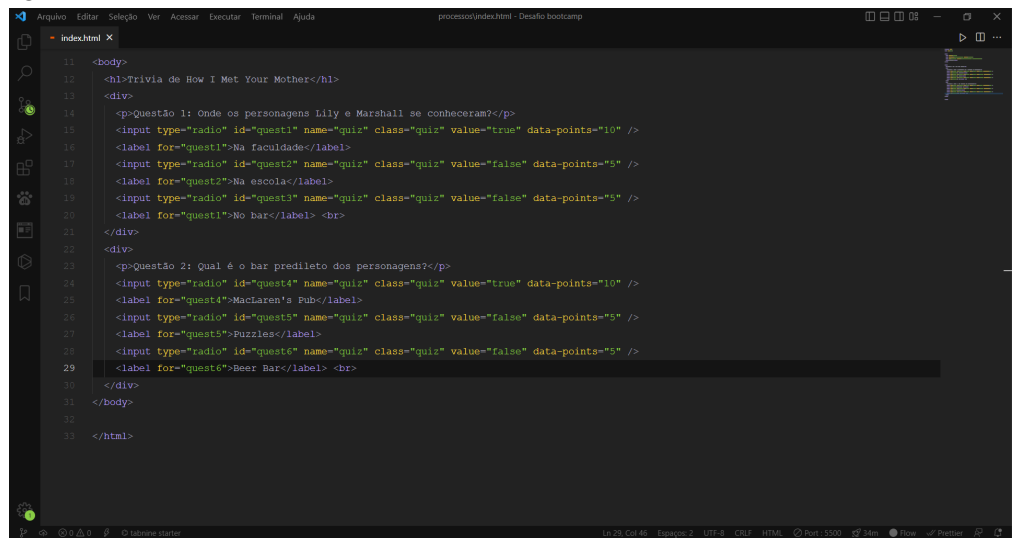


imagem 3: Entrada

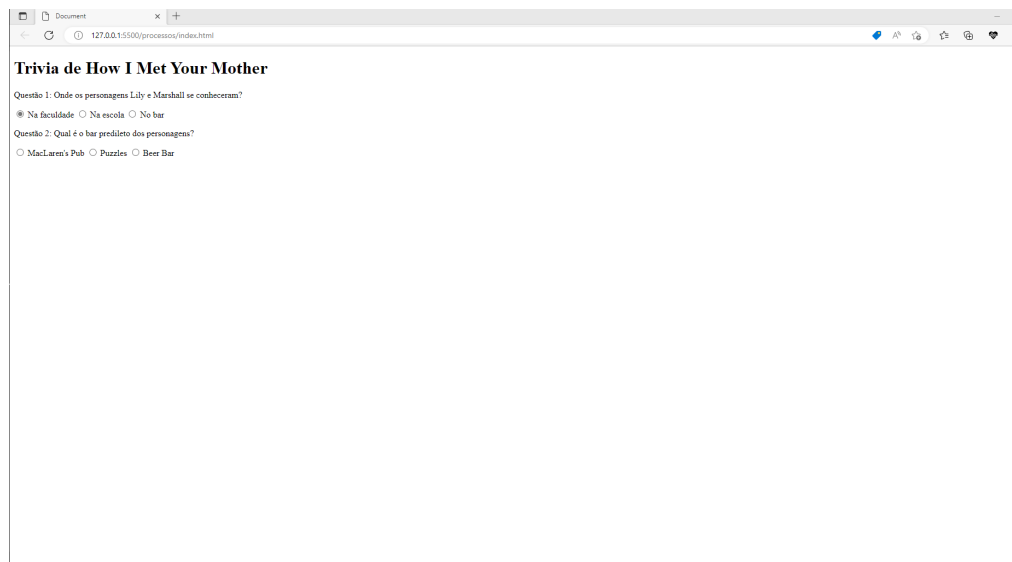
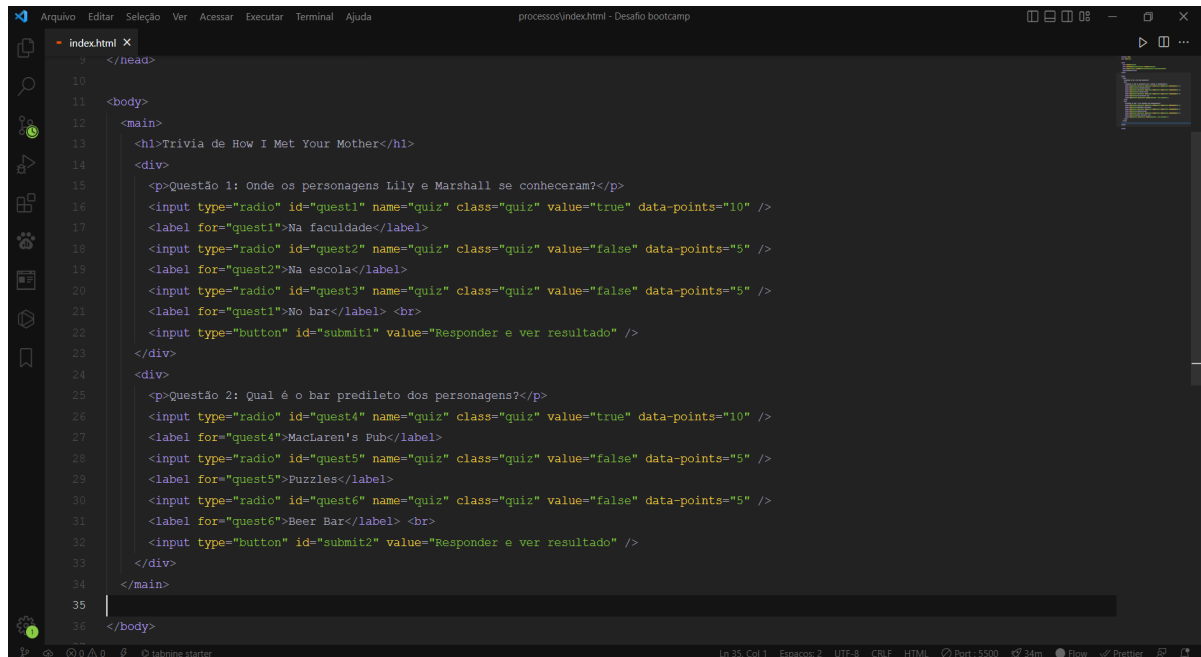


imagem 4: Saída

3. Na quarta parte foi criado um input do type button para ser o botão onde o usuário irá clicar para ver se a alternativa correta estava correta ou não (nessa etapa foi

criado somente o botão sem a funcionalidade javascript).



```
7 </head>
10
11 <body>
12 <main>
13 <h1>Trivia de How I Met Your Mother</h1>
14 <div>
15 <p>Questão 1: Onde os personagens Lily e Marshall se conheceram?</p>
16 <input type="radio" id="quest1" name="quiz" class="quiz" value="true" data-points="10" />
17 <label for="quest1">Na faculdade</label>
18 <input type="radio" id="quest2" name="quiz" class="quiz" value="false" data-points="5" />
19 <label for="quest2">Na escola</label>
20 <input type="radio" id="quest3" name="quiz" class="quiz" value="false" data-points="5" />
21 <label for="quest1">No bar</label> <br>
22 <input type="button" id="submit1" value="Responder e ver resultado" />
23 </div>
24 <div>
25 <p>Questão 2: Qual é o bar predileto dos personagens?</p>
26 <input type="radio" id="quest4" name="quiz" class="quiz" value="true" data-points="10" />
27 <label for="quest4">MacLaren's Pub</label>
28 <input type="radio" id="quest5" name="quiz" class="quiz" value="false" data-points="5" />
29 <label for="quest5">Puzzles</label>
30 <input type="radio" id="quest6" name="quiz" class="quiz" value="false" data-points="5" />
31 <label for="quest6">Beer Bar</label> <br>
32 <input type="button" id="submit2" value="Responder e ver resultado" />
33 </div>
34 </main>
35
36 </body>
```

imagem 5: Entrada

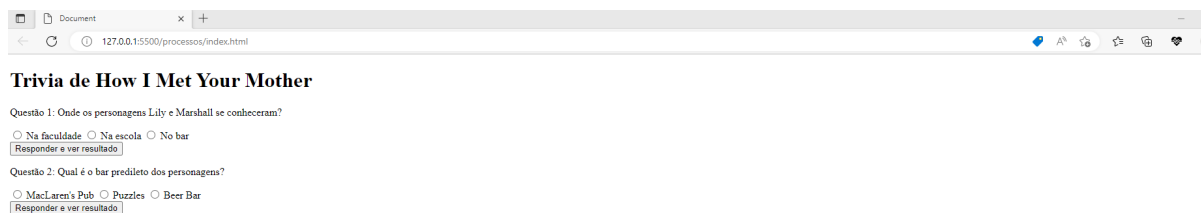
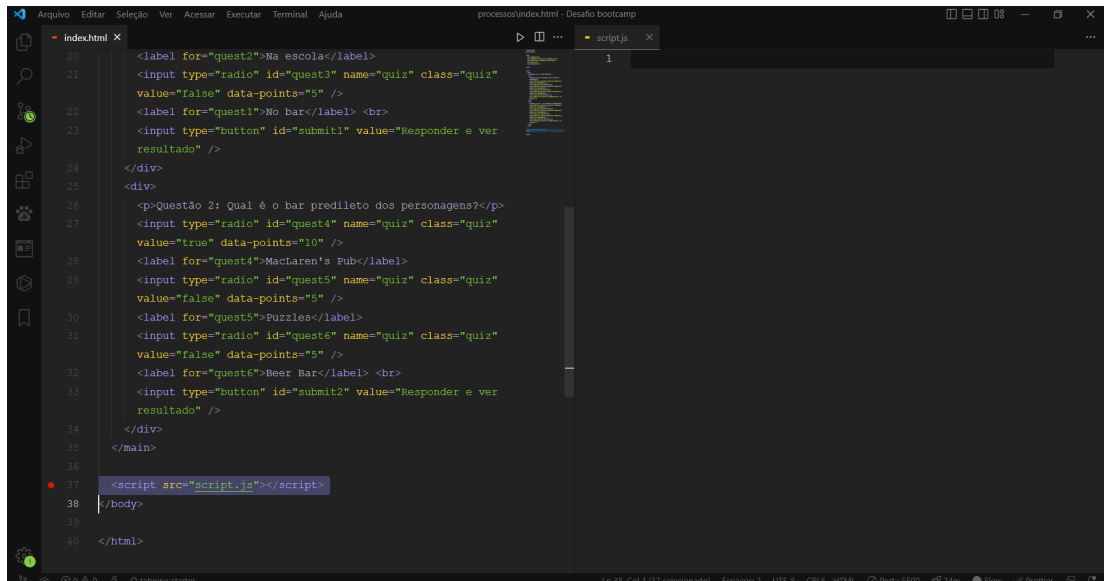


imagem 6: Saída

4. Nessa etapa foi criada a funcionalidade javascript do botão da etapa 4, onde essa funcionalidade consiste em mostrar um alerta para o usuário informando se ele tinha respondido corretamente ou não a pergunta. Essa etapa como envolve mais conhecimento e lógica foi feita por etapas, testando passo a passo.

- Primeiramente no arquivo index.html foi adicionado um script que leva a um novo arquivo onde é trabalhado o javascript do meu index. Linha 37 e do lado tem a pasta

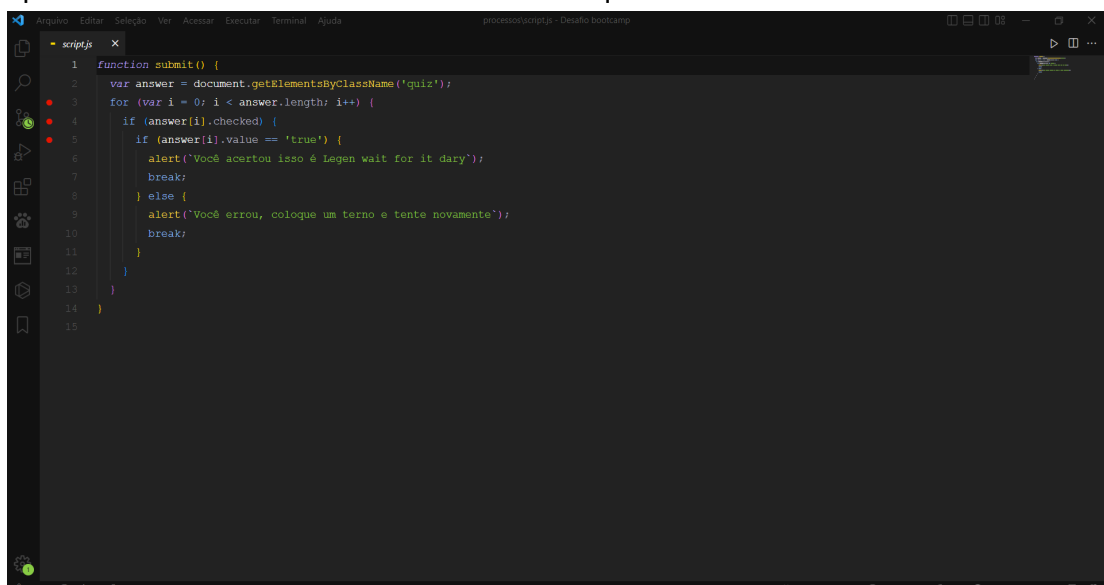
que foi criada a partir desse comando.



```
20 <label for="quest2">Na escola</label>
21 <input type="radio" id="quest3" name="quiz" class="quiz"
22 value="false" data-points="5" />
23 <label for="quest1">No bar</label> <br>
24 <input type="button" id="submit1" value="Responder e ver
25 resultado" />
26 </div>
27 <div>
28 <p>Questão 2: Qual é o bar predileto dos personagens?</p>
29 <input type="radio" id="quest4" name="quiz" class="quiz"
30 value="true" data-points="10" />
31 <label for="quest4">Maclaren's Pub</label>
32 <input type="radio" id="quest5" name="quiz" class="quiz"
33 value="false" data-points="5" />
34 <label for="quest5">Puzzles</label>
35 <input type="radio" id="quest6" name="quiz" class="quiz"
36 value="false" data-points="5" />
37 <label for="quest6">Beer Bar</label> <br>
38 <input type="button" id="submit2" value="Responder e ver
39 resultado" />
40 </div>
41 </main>
42 <script src="script.js"></script>
43 </body>
44 </html>
```

- Na segunda parte foi adicionado uma função de onclick no meu botão, essa função foi nomeada e invocada no meu index e no meu script foi feito a lógica da função. Dentro dessa função também foi usado uma seleção de nós dom para pegar meus valores do index e passar para o meu script, nessa etapa foi escolhido a função `getElementsByClassName()` para retornar uma lista de nós, a class de todos os inputs rádios foram iguais para todos serem chamados numa mesma função.

Na função foi utilizado um loop for para checar item por item e dentro desse loop uma estrutura condicional de if e else, onde foi definido que no if apareceria um alert de acerto e no else de erro. No if foi usado `.checked(linha 4)` para ver qual questão o usuário tinha selecionado, se o usuário tiver selecionado uma alternativa iria para outra estrutura condicional que nessa iria checar se o valor é true ou false(linha 5). Após isso foi criado os alert contendo o texto que o usuário veria.



```
1 function submit() {
2   var answer = document.getElementsByClassName('quiz');
3   for (var i = 0; i < answer.length; i++) {
4     if (answer[i].checked) {
5       if (answer[i].value == 'true') {
6         alert('Você acertou isso é Legen wait for it dary!');
7         break;
8       } else {
9         alert('Você errou, coloque um terno e tente novamente!');
10        break;
11      }
12    }
13  }
14 }
15
```

imagem 7: Entrada

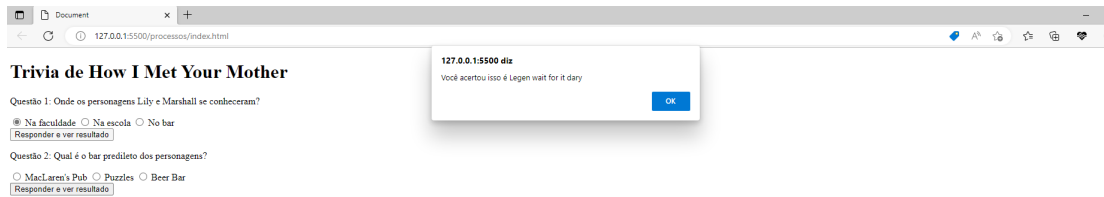
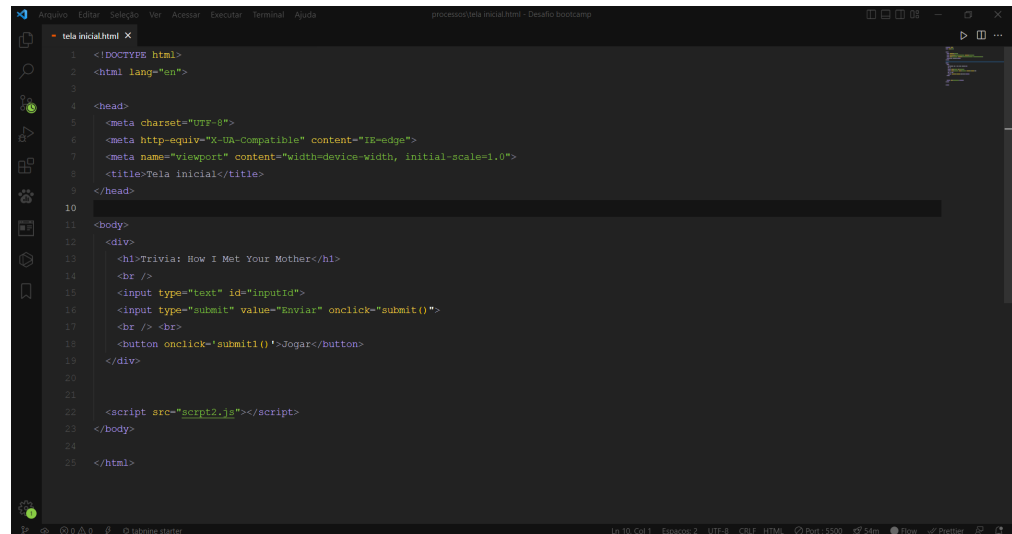


imagem 8: Saída

5. Nessa etapa foi criada a tela inicial do Trivia, primeiro foi criado um outro arquivo html, chamado de tela inicial, nesse arquivo foi colocado título, um input do type text para o usuário colocar seu nome e dois botões com funcionalidades distintas. O primeiro botão de ok para pegar o nome do usuário e emite um alert de bem vindo com seu nome e o outro botão de jogar que direciona o usuário para o trivia de How I Met Your Mother contendo perguntas e respostas. A funcionalidade de ambos os botões foi feita com javascript seguindo a mesma lógica das etapas anteriores de criar um script js próprio para esse arquivo de tela inicial.

- Primeiramente foi criado o arquivo tela inicial.html contendo título, dois inputs de types diferentes, um input de type text para o nome do usuário e um input de type button. No botão com valor enviar foi nomeada e invocada uma função(submit()) para ser criada a lógica e funcionalidade no meu arquivo scrip2.js, nesse botão é para o nome do usuário ser pego e aparecer num alert desejando boas vindas. No meu botão de jogar foi nomeada e invocada também uma função chamada submit 2(), esse botão tem como função

direcionar o usuário para a página que contém o trivia.



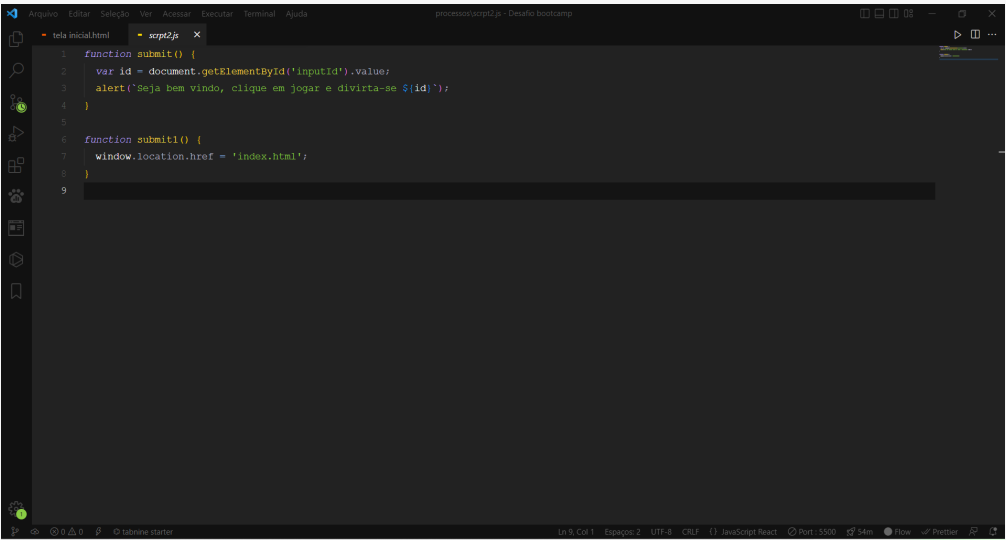
```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <title>Tela inicial</title>
9 </head>
10
11 <body>
12   <div>
13     <h1>Trivia: How I Met Your Mother</h1>
14     <br />
15     <input type="text" id="inputId">
16     <input type="submit" value="Enviar" onclick="submit()">
17     <br /> <br>
18     <button onclick="submit1()">Jogar</button>
19   </div>
20
21   <script src="script2.js"></script>
22 </body>
23 </html>
```

imagem 9: Entrada



imagem 10: Saída

- Após esse momento, começou-se a trabalhar no meu arquivo script2.js que faz com que as funcionalidades dos botões funcione de maneira esperada, nessa etapa foi usada duas funções, a primeira novamente se usou uma função `getElementById()` para acessar diretamente o nó do nome do usuário e após isso emite um alerta de Bem vindo com o respectivo nome.
- A segunda função foi usado uma propriedade `window.location.href` para direcionar o usuário para a minha pagina que contem o trivia



```
1 function submit() {  
2   var id = document.getElementById('inputId').value;  
3   alert('Seja bem vindo, clique em jogar e divirta-se ${id}');  
4 }  
5  
6 function submit1() {  
7   window.location.href = 'index.html';  
8 }  
9
```

imagem 11: Entrada

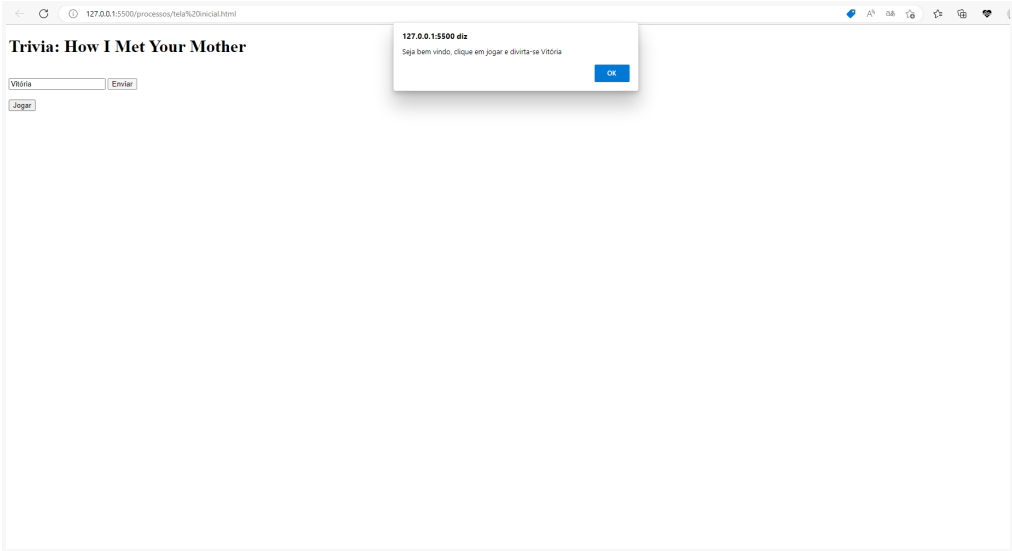


imagem 12: Saída botão de enviar

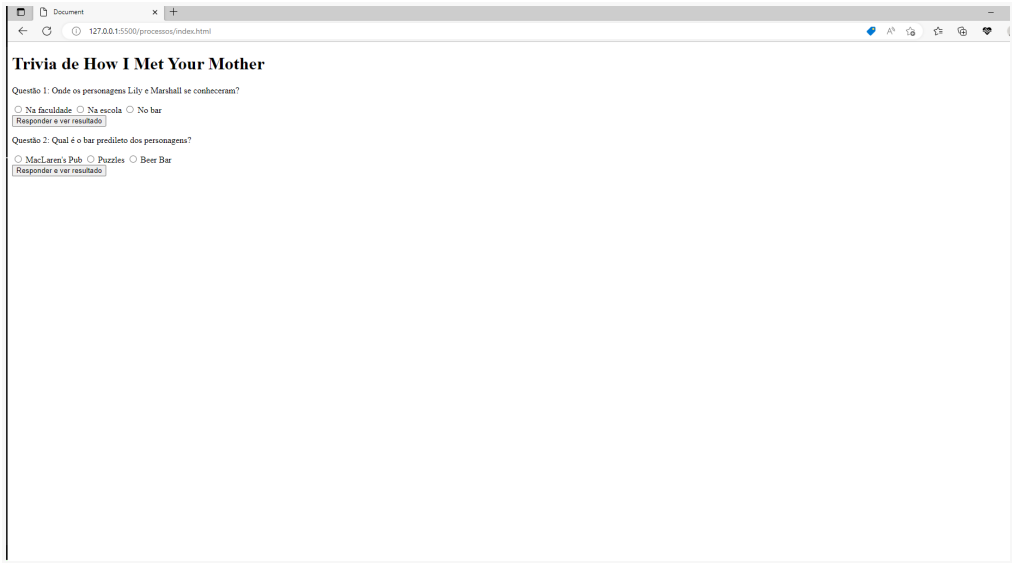
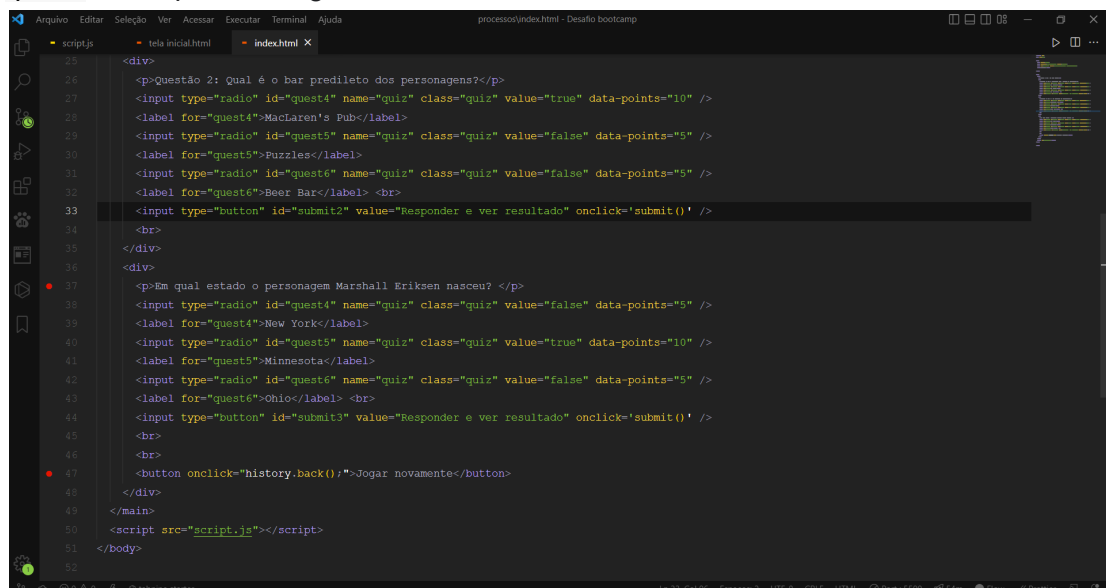


imagem 13: Saída botão de jogar

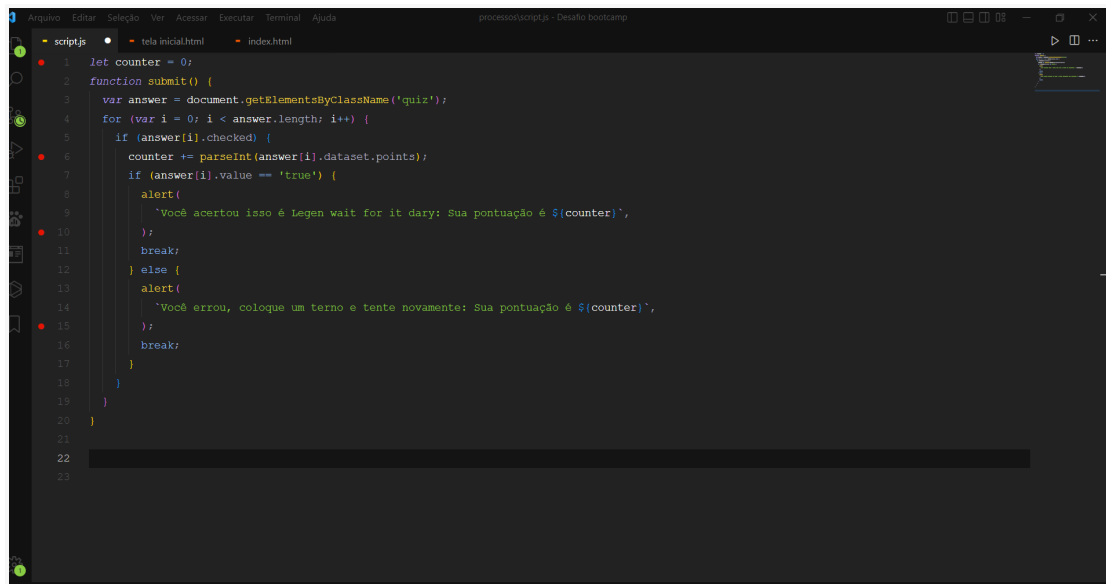
6. Nessa etapa foram feitas três melhorias que achamos que iria melhorar a experiência do usuário com o trivia. A primeira foi adicionar mais uma pergunta ao usuário. A segunda foi adicionar um botão de jogar novamente que retornaria o usuário à página inicial. A terceira melhoria foi adicionar no nosso alert de alternativa correta ou errada uma pontuação para tornar o jogo mais divertido.

- A pergunta foi adicionada no arquivo index.html seguindo a mesma lógica dos códigos anteriores
- Para adicionar a funcionalidade do botão jogar novamente foi criada uma função no arquivo index.html usando javascript no html, foi utilizado o método `history.back()` que é usado para carregar a URL anterior na lista de histórico.



```
25 <div>
26 <p>Questão 2: Qual é o bar predileto dos personagens?</p>
27 <input type="radio" id="quest4" name="quiz" class="quiz" value="true" data-points="10" />
28 <label for="quest4">MacLaren's Pub</label>
29 <input type="radio" id="quest5" name="quiz" class="quiz" value="false" data-points="5" />
30 <label for="quest5">Puzzles</label>
31 <input type="radio" id="quest6" name="quiz" class="quiz" value="false" data-points="5" />
32 <label for="quest6">Beer Bar</label> <br>
33 <input type="button" id="submit2" value="Responder e ver resultado" onclick='submit()' />
34 <br>
35 </div>
36 <div>
37 <p>Em qual estado o personagem Marshall Eriksen nasceu? </p>
38 <input type="radio" id="quest4" name="quiz" class="quiz" value="false" data-points="5" />
39 <label for="quest4">New York</label>
40 <input type="radio" id="quest5" name="quiz" class="quiz" value="true" data-points="10" />
41 <label for="quest5">Minnesota</label>
42 <input type="radio" id="quest6" name="quiz" class="quiz" value="false" data-points="5" />
43 <label for="quest6">Ohio</label> <br>
44 <input type="button" id="submit3" value="Responder e ver resultado" onclick='submit()' />
45 <br>
46 <br>
47 <button onclick="history.back();" >Jogar novamente</button>
48 </div>
49 </main>
50 <script src="script.js"></script>
51 </body>
52
```

- Para adicionar uma pontuação ao jogo voltamos ao nosso primeiro arquivo script.js e adicionamos uma variável global chamada counter, foi usado uma função `parseInt` para converter uma string em number e também foi usado um atributo `data` para pegar elemento de points que foi definido no meu index.html, foi colocado logo após a linha 5 para pegar e dar valor ao rádio que o usuário marcou. Na linha 8 e 11 foi mudado ' por ` para dessa forma mostrar ao usuário no alert seus pontos.



```
1 let counter = 0;
2 function submit() {
3   var answer = document.getElementsByClassName('quiz');
4   for (var i = 0; i < answer.length; i++) {
5     if (answer[i].checked) {
6       counter += parseInt(answer[i].dataset.points);
7       if (answer[i].value == 'true') {
8         alert(
9           'Você acertou isso é Legen wait for it dary: Sua pontuação é ${counter}',
10        );
11        break;
12      } else {
13        alert(
14          'Você errou, coloque um terno e tente novamente: Sua pontuação é ${counter}',
15        );
16        break;
17      }
18    }
19  }
20 }
21
22
23
```

imagem 14: Entrada

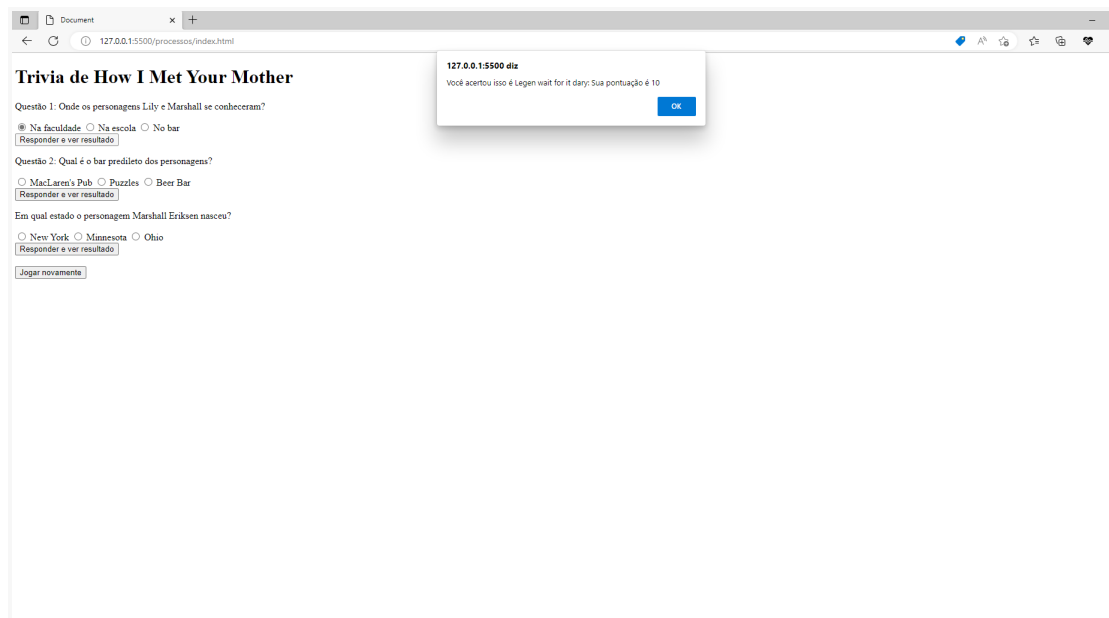


imagem 15: Saída

7. Nessa última etapa de finalização do projeto foi feita estilização para ficar visualmente mais bonito e atrativo para o usuário, onde foi colocado cor de fundo, fonte da letra, tamanho, posicionamento dos textos e formatos

- Para isso foi criado e linkado aos arquivos tela1.html e index.html um arquivo próprio para css (linha 9)

```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width,
8     initial-scale=1.0">
9   <link href="estilo.css" rel="stylesheet" type="text/css" />
10 </head>
11
12 <body>
13   <div>
14     <h1>Trivia: How I Met Your Mother</h1>
15     <br />
16     <input type="text" id="inputId">
17     <input type="submit" value="Enviar" onclick="submit()">
18     <br />
19     <button onclick="submit1()">Jogar</button>
20   </div>
21
22   <script src="script2.js"></script>
23 </body>
24 </html>
  
```

- Após isso começou a estilização nos arquivos css onde foi utilizado background para a cor, fonte para os textos, padding para definir distâncias, border-radius para arredondar a div, posição para definir o local dos textos.

```

1 body {
2   font-family: Arial;
3   background-image: linear-gradient(100deg, #EEEBAA, #333A05);
4 }
5
6 div {
7   background-color: #EEEBAA;
8   position: absolute;
9   top: 50%;
10  left: 50%;
11  transform: translate(-50%, -50%);
12  /*para posicionar certinho */
13  padding: 50px;
14  /*a distancia que fica um do outro */
15  border-radius: 15px;
16  /*para arredondar as bordas*/
17 }
18
19 h1 {
20   position: absolute;
21   top: 10%;
22   left: 22%;
23   transform: translate(-10%, -200%);
24   /*para posicionar certinho */
25   padding: 10px;
26   /*a distancia que fica um do outro */
27 }
  
```

imagem 16: Entrada dos respectivos css

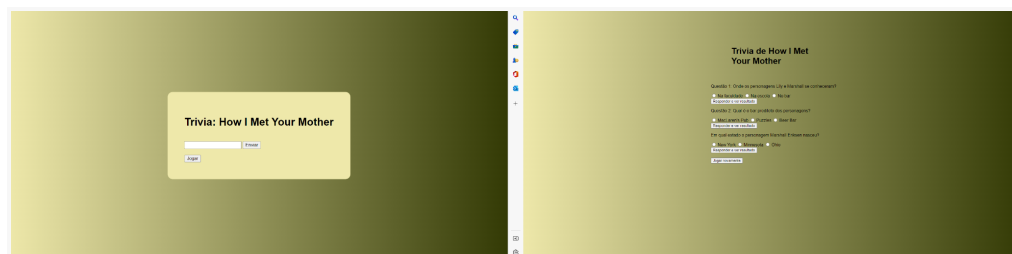


imagem 17:Saída