

Mão na massa

1) Ajustando o espaçamento para telas menores

Você está trabalhando em um projeto web e percebe que o design para telas menores precisa ser aprimorado. O conteúdo parece "espremido" ao centro, devido ao espaçamento excessivo nas laterais. Seu objetivo é ajustar o padding da classe `.apresentacao` no CSS para melhorar a visualização em dispositivos menores.

2) Refinando a largura do conteúdo em dispositivos menores

Após ajustar o espaçamento, você percebe que o conteúdo dentro da classe `.apresentacao__conteudo` ainda ocupa muito espaço em telas menores. Seu desafio agora é ajustar a largura dessa classe para que ela acompanhe o novo padding aplicado na classe `.apresentacao`.

3) Avaliando o efeito do `width: auto` em dispositivos menores

A última tarefa envolve avaliar o impacto da mudança do `width` para `auto` na classe `.apresentacao__conteudo`. Você precisa verificar se a largura do conteúdo se adapta corretamente ao padding de 5% aplicado, garantindo uma exibição adequada em telas menores.

4) Criando uma conta no GitHub

Neste exercício, você irá criar uma conta no GitHub, uma plataforma essencial para qualquer pessoa que atua com desenvolvimento web. O GitHub permite hospedar e compartilhar projetos, facilitando a colaboração. Sua tarefa é navegar pelo processo de inscrição no GitHub, configurar sua conta e explorar as principais funcionalidades disponíveis.

5) Subindo um projeto no GitHub

Agora que você tem uma conta no GitHub, o próximo passo é aprender a "subir" um projeto. Você utilizará o GitHub para hospedar seu projeto web, que inclui arquivos HTML, CSS, JavaScript, entre outros. O foco será no processo de criação de um novo repositório e no envio (upload) dos arquivos do seu projeto para o GitHub.

Opinião do instrutor

1) Ajustando o espaçamento para telas menores

- Abra o arquivo `style.css`.
- Encontre a regra CSS para `.apresentacao` dentro do bloco `@media (max-width: 1200px)`.
- Altere o `padding` para `5%` para todos os lados. O código ficará assim:

```
@media (max-width: 1200px) {  
  .apresentacao {  
    flex-direction: column-reverse;  
    padding: 5%;  
  }  
  ...  
}
```

2) Refinando a largura do conteúdo em dispositivos menores

- Abra o arquivo `style.css`.
- No bloco `@media (max-width: 1200px)`, encontre a classe `.apresentacao__conteudo`.
- Defina a propriedade `width` para `auto`. O código ficará assim:

```
@media (max-width: 1200px) {
```

```
...  
.apresentacao__conteudo {  
  width: auto;  
}  
}
```

3) Avaliando o efeito do `width: auto` em dispositivos menores

- Após fazer as alterações no arquivo `style.css`, abra o site no navegador.
- Use a ferramenta de inspeção para verificar a classe `.apresentacao__conteudo`.
- Observe a adaptação da largura do conteúdo ao `padding` de 5%. Faça ajustes no `width`, se necessário.

4) Criando uma conta no GitHub

1. Acesse github.com em seu navegador.
2. Escolha "Sign Up" para criar uma nova conta.
3. Preencha as informações solicitadas, incluindo e-mail, senha e nome de usuário.
4. Complete a verificação humana e insira o código de verificação enviado ao seu e-mail.
5. Selecione suas preferências, como "Just Me" para o tamanho da equipe e "Student" para o tipo de usuário.
6. Escolha as funcionalidades que pretende utilizar, como "Collaborative coding".
7. Complete o processo e explore a interface do GitHub, observando áreas como "Pinned" para projetos em destaque.

5) Subindo um projeto no GitHub

1. Faça login em sua conta do GitHub.
2. Clique em "New repository" para criar um novo repositório.
3. Dê um nome ao seu repositório e, se desejar, adicione uma descrição.
4. Defina a visibilidade do repositório (público ou privado).
5. Crie o repositório.
6. Siga as instruções do GitHub para "push" (enviar) seus arquivos locais do projeto para o repositório criado. Isso geralmente envolve comandos Git no terminal do seu computador.