

Você é o desenvolvedor free-lancer e está participando de um processo seletivo para uma multinacional, o salário inicial para programador Júnior gira em torno de \$3000,00 dólares por mês. A empresa não exige experiência anterior mas pediu seu portfólio no github.com ou gitlab.com. Você teve a ideia de compartilhar uma lista de exercícios organizando tudo por temas (você define como organizar seus exercícios).

Compartilhe o link do git e marque a entrega da atividades, seus commits serão usados como forma de avaliação para formar sua nota.

1. Utilizar semântica de linguagem de marcação conforme normas

1. Crie uma página HTML com estrutura semântica usando `<header>`, `<nav>`, `<main>`, `<section>`, `<article>`, `<aside>` e `<footer>`.
2. Converta um layout feito com `<div>` em um layout utilizando apenas elementos semânticos.
3. Utilize corretamente `<figure>` e `<figcaption>` para exibir imagens com legendas.
4. Marque um artigo de blog usando `<article>` e seus respectivos subtópicos dentro de `<section>`.
5. Estructure um menu de navegação usando `<nav>` e listas ``, aplicando estilos adequados.

2. Elaborar formulários de página web e mobile

6. Crie um formulário de cadastro com campos de nome, e-mail, senha e um botão de envio.

7. Adicione validação de e-mail e senha no formulário usando atributos HTML5 como `required` e `pattern`.
 8. Crie um formulário de login responsivo, utilizando Flexbox ou Grid para alinhamento.
 9. Implemente um formulário que utilize `<fieldset>` e `<legend>` para organizar informações pessoais e de endereço.
 10. Desenvolva um formulário de pesquisa com botão de envio estilizado.
-

3. Adequar a interface web para diferentes dispositivos de acesso

11. Use media queries para criar um layout responsivo que se adapte a diferentes tamanhos de tela.
 12. Implemente um menu de navegação que se transforme em um menu hamburguer em telas menores.
 13. Ajuste um layout usando unidades relativas como `em`, `rem` e `%` ao invés de valores fixos em pixels.
 14. Adicione `meta viewport` no `<head>` e teste a diferença sem ele.
 15. Crie um carrossel de imagens responsivo usando apenas CSS.
-

4. Desenvolver interfaces web interativas com linguagem de programação

16. Desenvolva um botão que muda de cor ao passar o mouse usando JavaScript.
 17. Crie um contador de cliques que exiba a contagem em tempo real na página.
 18. Faça um modal de aviso que aparece ao clicar em um botão e desaparece ao clicar fora dele.
 19. Desenvolva uma galeria de imagens onde as imagens se ampliam ao serem clicadas.
 20. Crie um sistema de tabs (abas) para exibir diferentes conteúdos ao clicar nelas.
-

5. Aplicar técnicas de estilização de páginas web

21. Desenvolva um layout usando apenas CSS Grid.
 22. Crie um botão animado usando `@keyframes`.
 23. Implemente um efeito de sombra (`box-shadow`) para dar profundidade aos elementos.
 24. Utilize `clip-path` para criar botões com formatos diferenciados.
 25. Estilize um formulário usando `focus` e `placeholder` customizado.
-

6. Desenvolver interfaces web utilizando frameworks

26. Crie um layout básico utilizando Bootstrap.

- 27. Desenvolva um modal utilizando a biblioteca MUI.
- 28. Crie um sistema de cards utilizando Tailwind CSS.
- 29. Implemente uma barra de navegação fixa com Material-UI.
- 30. Utilize o Grid do Bootstrap para organizar um layout responsivo.

7. Desenvolver interfaces web consumindo API

- 31. Utilize `fetch` para buscar dados de uma API pública e exibir na tela.
- 32. Crie um campo de busca que filtre informações de uma API e exiba os resultados.
- 33. Desenvolva um sistema de login onde os dados são validados por uma API fictícia.
- 34. Exiba uma lista de usuários consumida de uma API JSON.
- 35. Crie uma interface que exibe a previsão do tempo baseada na localização do usuário.

8. Diferenciar os aspectos de aplicabilidade entre UX e UI

- 36. Analise um site famoso e liste aspectos de UI e UX bem implementados.
- 37. Faça um redesign de um formulário mal estruturado, melhorando a experiência do usuário.

38. Realize um teste A/B com duas versões de um botão e registre qual recebe mais cliques.
39. Simule um fluxo de cadastro otimizado para dispositivos móveis.
40. Crie uma versão acessível de um formulário, incluindo suporte para leitores de tela.