

Tarefas de Casa – Flexbox, Grid e Preparação para JavaScript

Objetivo geral: reforçar o uso de *Flexbox* e *CSS Grid* e, ao mesmo tempo, preparar o terreno para começarmos JavaScript em breve.

Parte 1 – Exercícios de CSS (Flexbox e Grid)

Neste conjunto de exercícios, o foco é repetir padrões simples de layout até ficar natural. Priorize fazer um pouco todo dia (15–20 minutos) em vez de tentar estudar muitas horas de uma vez.

Exercício 1 – Barra de navegação (Flexbox)

Objetivo:

- Criar um cabeçalho com uma **logo à esquerda** e um **menu de links à direita**, usando `display: flex`.

Instruções:

- Crie um arquivo `index.html` simples com um elemento `<header class="header">`.
- Dentro do header, coloque uma `<div class="logo">` e um `<nav class="nav">` com 3 links.
- No CSS, aplique em `.header`:
 - `display: flex;`
 - `justify-content: space-between;`
 - `align-items: center;`
- Estilize os links do menu (`.nav a`) com
 - cor,
 - remoção do sublinhado (`text-decoration: none;`),
 - e um espaçamento lateral (ex.: `margin-left: 16px;`).

Dica: tente montar esse header **sem olhar o código da aula**. Se travar, olhe só um detalhe, ajuste, e tente de novo.

Exercício 2 – Galeria de imagens (Grid)

Objetivo:

- Criar uma grade de imagens usando **CSS Grid**, com 3 colunas no desktop.

Instruções:

- Crie uma seção `<section class="galeria">` com 6 imagens (pode usar a mesma imagem repetida com URLs de placeholder).
- No CSS, aplique em `.galeria`:
 - `display: grid;`

- `grid-template-columns: repeat(3, 1fr);`
- `gap: 16px;` (ou outro valor de espaçamento).

Opcional: se quiser, acrescente um `padding` na galeria ou coloque um fundo diferente nas imagens para visualizar melhor a grade.

Exercício 3 – Layout com Sidebar (Flex ou Grid)

Objetivo:

- Criar um layout de "dashboard" simples com **barra lateral (sidebar)** e **área principal de conteúdo**.

Instruções (escolha Flex ou Grid):

- Crie um contêiner principal, por exemplo: `<div class="layout">`.
- Dentro dele, crie dois elementos:
 - `<aside class="sidebar">` – para links de navegação.
 - `<main class="conteudo">` – para o conteúdo principal.
- **Opção 1 – Usar Flexbox:**
 - Em `.layout`, use `display: flex;`.
 - Dê uma largura fixa para a sidebar (ex.: `width: 200px;`).
 - Deixe o `.conteudo` ocupar o restante do espaço (por exemplo, com `flex: 1;`).
- **Opção 2 – Usar Grid:**
 - Em `.layout`, use `display: grid;`.
 - Defina as colunas com algo como `grid-template-columns: 200px 1fr;`.

Desafio extra: tente fazer a sidebar ficar em cima (como um header) quando a tela for muito estreita, usando `@media` e ajustando o layout.

Parte 2 – Preparando para JavaScript (sem usar JS ainda)

Enquanto você treina HTML e CSS, já podemos ir deixando o código pronto para, em breve, adicionar JavaScript. A ideia é **pensar na estrutura** que o JS vai manipular mais tarde.

Exercício 4 – Nomeando classes pensando no futuro

Objetivo:

- Criar classes claras e semânticas, fáceis de entender e de selecionar com JS depois.

Instruções:

- Revise um dos layouts que você criou (navbar, galeria ou dashboard).
- Verifique se as classes têm nomes significativos, por exemplo:
 - `.btn-primary`, `.card-list`, `.sidebar-menu`, `.main-content`.
- Evite nomes genéricos demais, como `.caixa1`, `.teste`, etc.
- Faça uma pequena lista (em um bloco de notas ou comentário no código) com 3 a 5 classes que você acha que seriam importantes de manipular com JS no futuro.

Exercício 5 – Criando elementos "interativos" (por enquanto só no HTML/CSS)

Objetivo:

- Adicionar ao layout elementos que **no futuro** serão controlados com JavaScript.

Instruções:

- Escolha uma das suas páginas (por exemplo, a que tem cards ou dashboard).
- Adicione pelo menos um dos elementos abaixo:
 - Um botão `<button class="btn-mostrar-mais">Mostrar mais</button>` abaixo de uma lista de cards.
 - Uma seção de texto extra, por exemplo `<section class="texto-extra">...`, que você pode deixar visível por enquanto.
 - Um "card destacado" com uma classe própria (ex.: `.card-destaque`), que futuramente poderá ser exibido/ocultado.
- Estilize esses elementos no CSS (cores, margens, etc.), mesmo que ainda não façam nada "dinâmico".

Importante: ainda não vamos usar JavaScript aqui. A meta é apenas deixar o HTML e o CSS organizados para que, quando o JS entrar, fique fácil selecionar e manipular esses elementos.

Exercício 6 – Imaginando interações em JavaScript

Objetivo:

- Começar a pensar *mentalmente* em como o JavaScript vai interagir com a página.

Instruções:

- Olhe para a página que você criou com botões, cards, sidebar, etc.
- Responda (em texto, em um arquivo ou papel) às perguntas abaixo:
 - Que elementos você gostaria de **mostrar/esconder** com um clique?
 - Que elementos poderiam mudar de estilo (cor, tamanho) ao interagir?
 - Que parte da página poderia ser carregada ou alterada sem recarregar tudo?

Guarde essas respostas: elas vão ser muito úteis quando começarmos a escrever o primeiro código JavaScript.

Sugestão de rotina

Tente seguir uma rotina parecida com esta (ajuste conforme o seu tempo):

- **Dia 1:** Exercício 1 (navbar com flex).
- **Dia 2:** Exercício 2 (galeria com grid).
- **Dia 3:** Exercício 3 (layout com sidebar).
- **Dia 4:** Exercícios 4 e 5 (nomes de classes e elementos interativos).
- **Dia 5:** Exercício 6 (responder às perguntas pensando em JS).

Lembrete: o importante não é estudar por muitas horas, e sim manter a constância. Se você conseguir 15–20 minutos bem focados por dia, já está ótimo.