# APRENDENDO GIT E GITHUB

# REPOSITÓRIO LOCAL (GIT)

Um **repositório local** tem esse nome porque vai estar sempre no seu computador. A função dele é facilitar a gestão de diversas versões do seu projeto de forma simples e automática.

O esquema de funcionamento do Git é totalmente focado no nosso computador. Analisando a imagem a seguir, vemos que o software está monitorando uma pasta chamada meu-site que tem a versão atual do projeto e uma pasta especial chamada .git com várias versões pelas quais o site passou durante sua evolução.



A grande vantagem de usar um **sistema de controle de versões** como o **Git** é poder voltar a qualquer momento para qualquer versão anterior do projeto de forma imediata, tudo 100% transparente para o programador.

# REPOSITÓRIO REMOTO (GITHUB)

* **GitHub** é um serviço que nos permite criar um **repositório remoto** na nuvem para guardar nossos projetos e versionamentos.
* Além de guardarmos nossos códigos nos servidores, podemos nos comunicar com outros desenvolvedores e até mesmo colaborar com outros projetos que estão disponíveis publicamente para todos.
* E um recurso chamado **GitHub Pages** vai permitir a hospedagem gratuita de sites simples, que usem HTML + CSS + JS e disponibilize o acesso através de uma URL.
* **GitHub Desktop** é bem simples , mas concentra muitas funcionalidades que unificam recursos do **Git** e do **GitHub** em um só lugar.

# GIT HUB PAGES – HOSPEDAGEM GRATIS PARA DIVULGAR PROJETOS BASICOS QUE TENHA HTML – CSS – JS

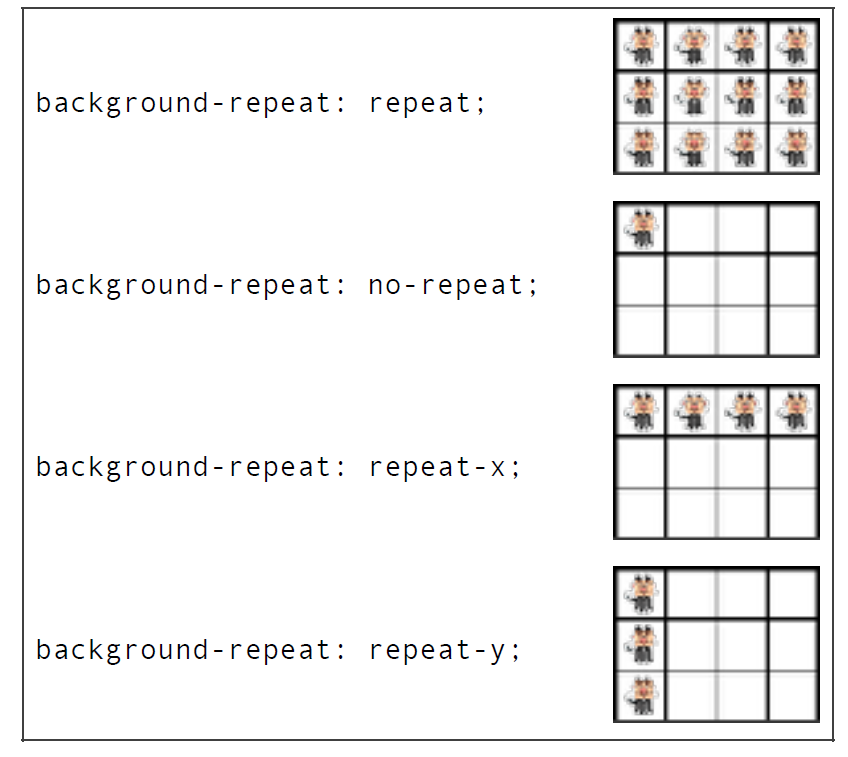


Na área de opções, existem duas maneiras de habilitar o GitHub Pages. Na primeira, clique no botão **None** e escolha a opção **master branch**. Com essa maneira, você vai precisar ter um arquivo index.html com o código da página principal que será exibida ao acessar o site. Essa técnica é mais usada quando queremos hospedar um site único e receber uma URL para apresentar o projeto a outra pessoa.

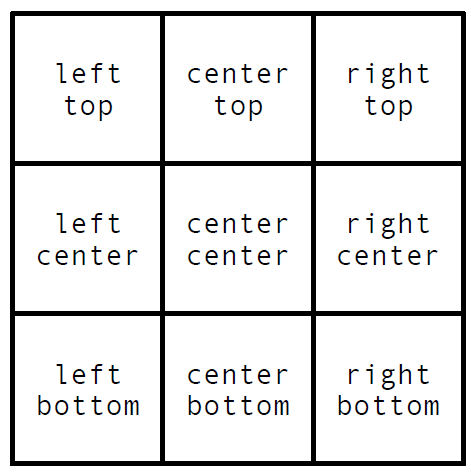


A segunda maneira é clicando sobre o botão **Choose a theme**, que vai permitir escolher um tema para exibir o conteúdo do arquivo README.md como uma página. Essa segunda técnica é mais usada quando vamos hospedar vários projetos em um mesmo repositório ou quando vamos guardar informações sobre um projeto que não é um site, como download de instaladores e coisas do tipo.

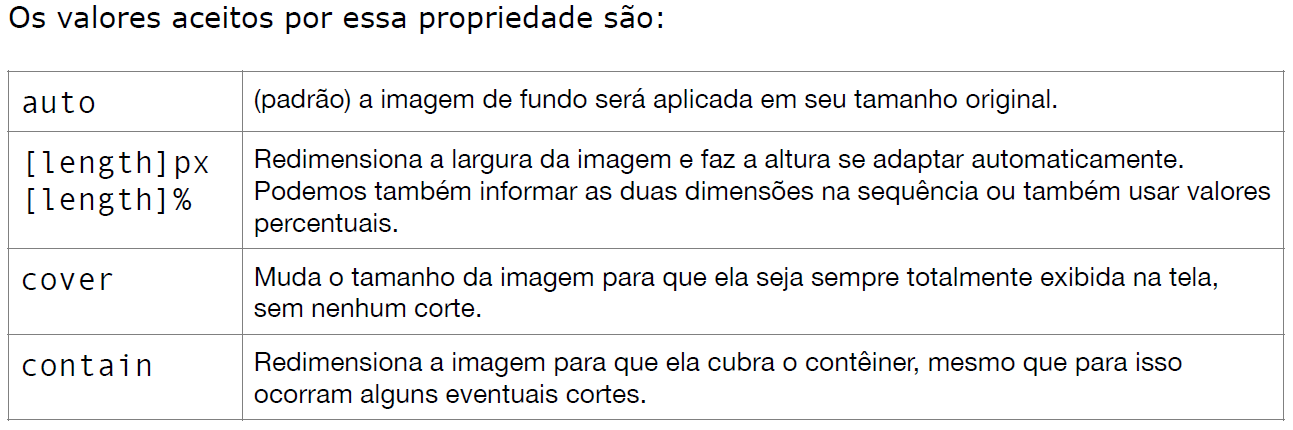
# PERSONALIZANDO O BACKGROUND DA CAIXA



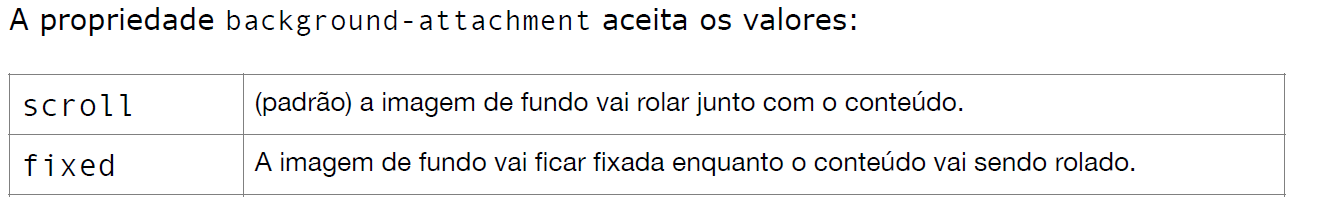
Além de escolher o nível de repetição do *background*, também podemos mudar a **posição de referência** de início das repetições. Por padrão, é considerado o canto esquerdo superior (left top), mas podemos ter várias opções. Use a imagem abaixo como referência sempre que precisar definir a posição do fundo com a propriedade **background-position** no seu código.

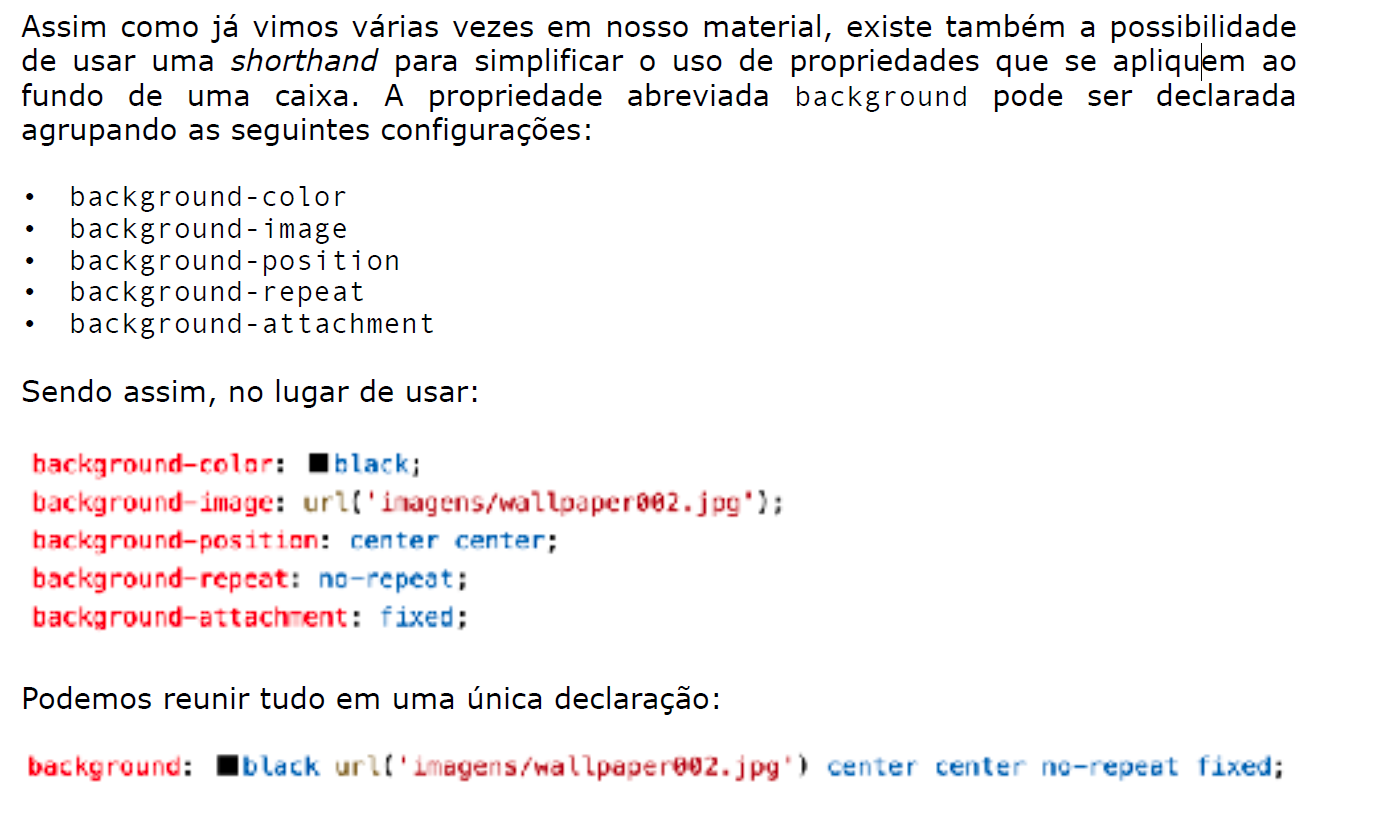


Outra coisa que podemos fazer é **redimensionar** a imagem para forçá-la a caber na caixa. Por padrão, nenhum redimensionamento será aplicado, e a imagem será exibida do seu tamanho natural. Porém, podemos usar a propriedade background-size para alterar esse comportamento.



A última propriedade que podemos configurar é o vínculo (attachment) da imagem de fundo com o resto do documento, principalmente se o conteúdo for maior do que a altura da página e seja necessário vazar uma rolagem vertical.





# TÉCNICA DE CENTRALIZAÇÃO DE CAIXA I



**Como Funciona**

* **Centralização Vertical e Horizontal**: O <div id="dentro"> será centralizado vertical e horizontalmente dentro do <div id="fora">. A combinação das propriedades left: 50%;, top: 50%; e transform: translate(-50%, -50%); garante que o contêiner interno fique exatamente no centro do contêiner externo.

**Resultado Final**

* O contêiner externo (roxo) ocupará 96% da altura da viewport e terá uma largura ajustada automaticamente pelo navegador.
* O contêiner interno (amarelo) terá 200px de altura e 300px de largura e será centralizado dentro do contêiner externo.

Se precisar de mais informações ou se tiver outras perguntas, estou aqui para ajudar!