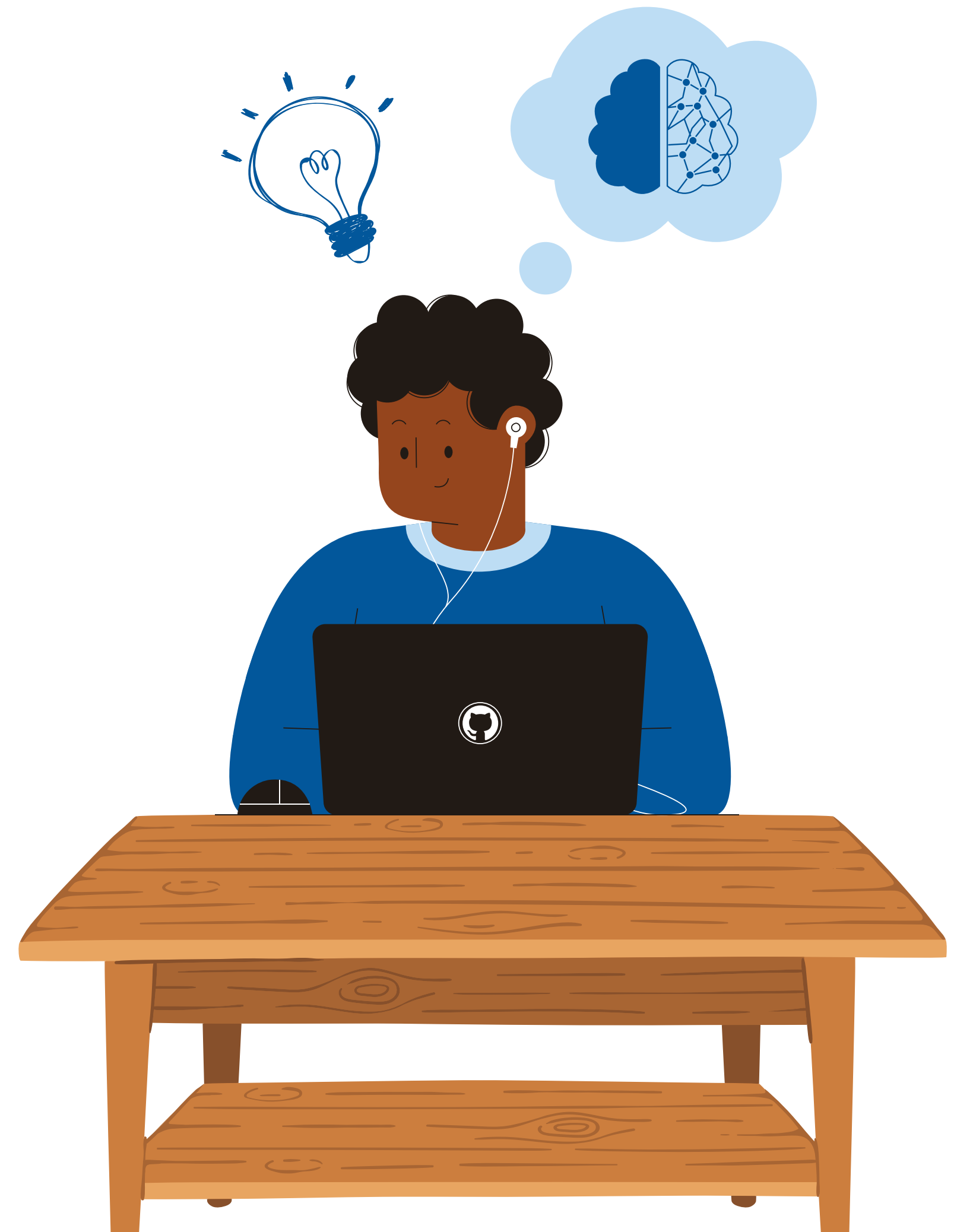
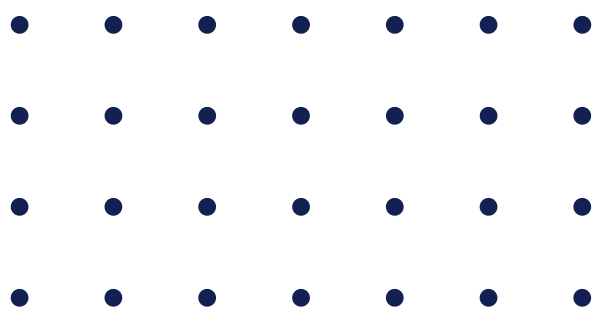
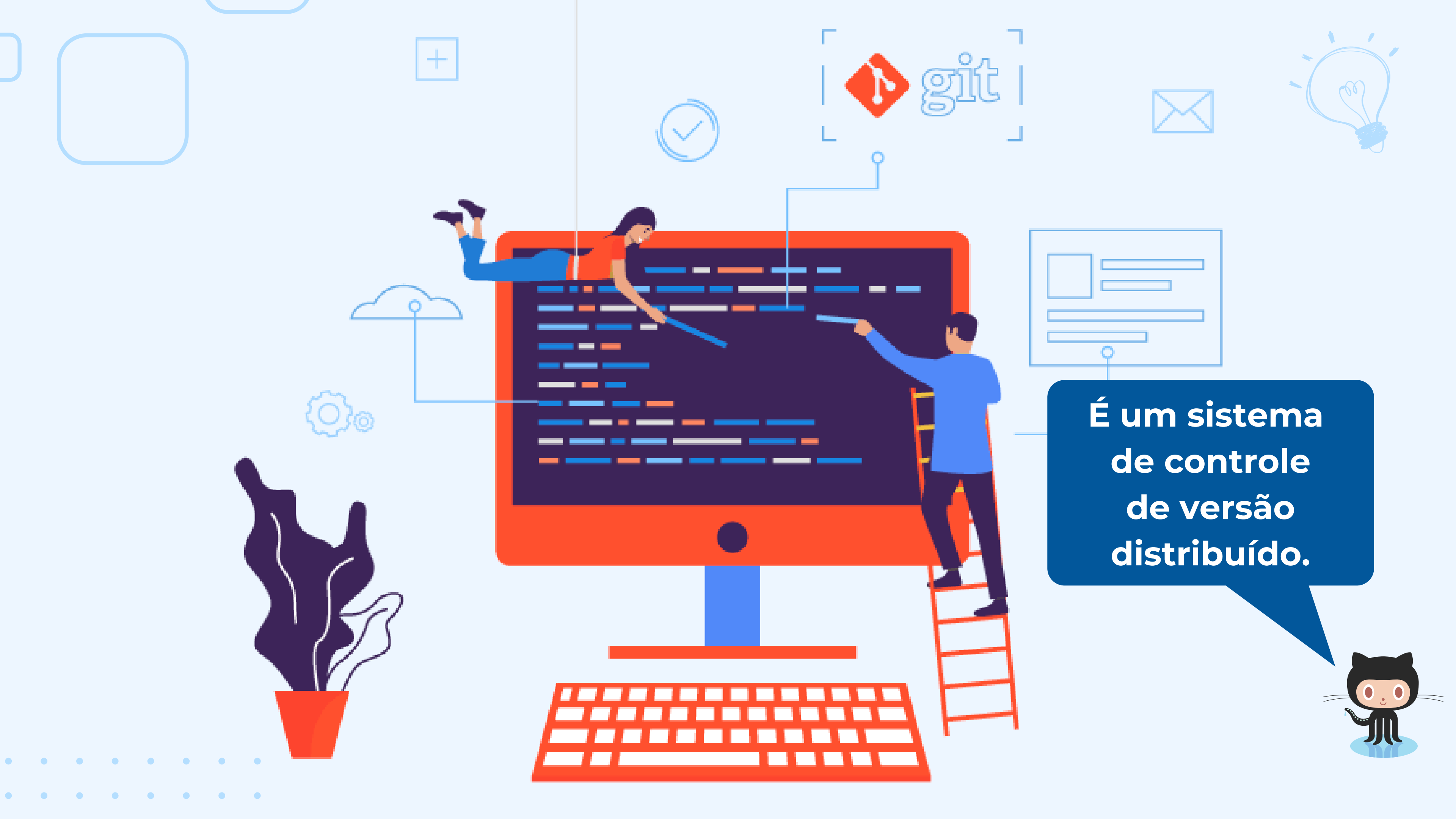


03





É um sistema
de controle
de versão
distribuído.





Iniciando Git

verificando se tem git instalado no Linux Ubuntu

```
git --version
```

instalação do git no Linux Ubuntu

```
sudo apt update
```

```
sudo apt install git
```

configuração inicial

```
git config --global user.name "Meu Nome"
```

```
git config --global user.email "meu.email@email.com"
```





acesse o diretório do seu usuário no linux (o til é um atalho para a pasta do usuário logado)
cd ~

vamos criar uma pasta para colocarmos todos os nossos projetos dentro dela
mkdir Projetos

ao usar o comando ls, verificamos que nossa pasta foi criada
ls -la

agora vamos entrar na nossa pasta recém criada
cd Projetos

vamos criar uma nova pasta que será a pasta que colocaremos nosso projeto desta aula
mkdir MeuPrimeiroGit

após criar a nova pasta, vamos entrar nela
cd MeuPrimeiroGit

para começarmos a utilizar o git, devemos inicializá-lo na pasta do nosso projeto
git init



Iniciando Git

Após iniciar o git no projeto vamos vincular ele ao nosso projeto do github
`git remote add origin https://github.com/gabrielomena/meuprojeto.git`

Vamos criar nossa Branch
`git branch -M main`

vamos utilizar um editor de texto por linha de comando para criar um arquivo chamado teste.md
`nano teste.md`

após editar o arquivo vamos verificar o status no git
`git status`

antes de subir para o github nosso arquivo vamos adicionar ele a lista
`git add teste.md`

agora podemos realizar nosso commit
`git commit -m "Meu primeiro Commit"`

agora só realizar o push
`git push`





Iniciando Git

Provavelmente você deve ter recebido a seguinte mensagem fatal: The current branch main has no upstream branch. To push the current branch and **set** the remote as upstream, use

```
git push --set-upstream origin main
```

rode o comando indicado pelo terminal

```
git push --set-upstream origin main
```



Iniciando Git

```
##### trabalhando com repositório remoto
git clone https://github.com/gabrielomena/curso-ifpk.git

##### entre na pasta do projeto
cd curso-ifpk

##### você pode abrir o projeto usando o vscode
code .

##### após realizar suas alterações é uma boa prática atualizar o repositório
##### do seu computador com as atualizações que foram feitas no remoto.
git pull #Atualiza todas as branches com as alterações do remoto

git pull origin master # Atualia apenas a branch master apenas do remoto origin

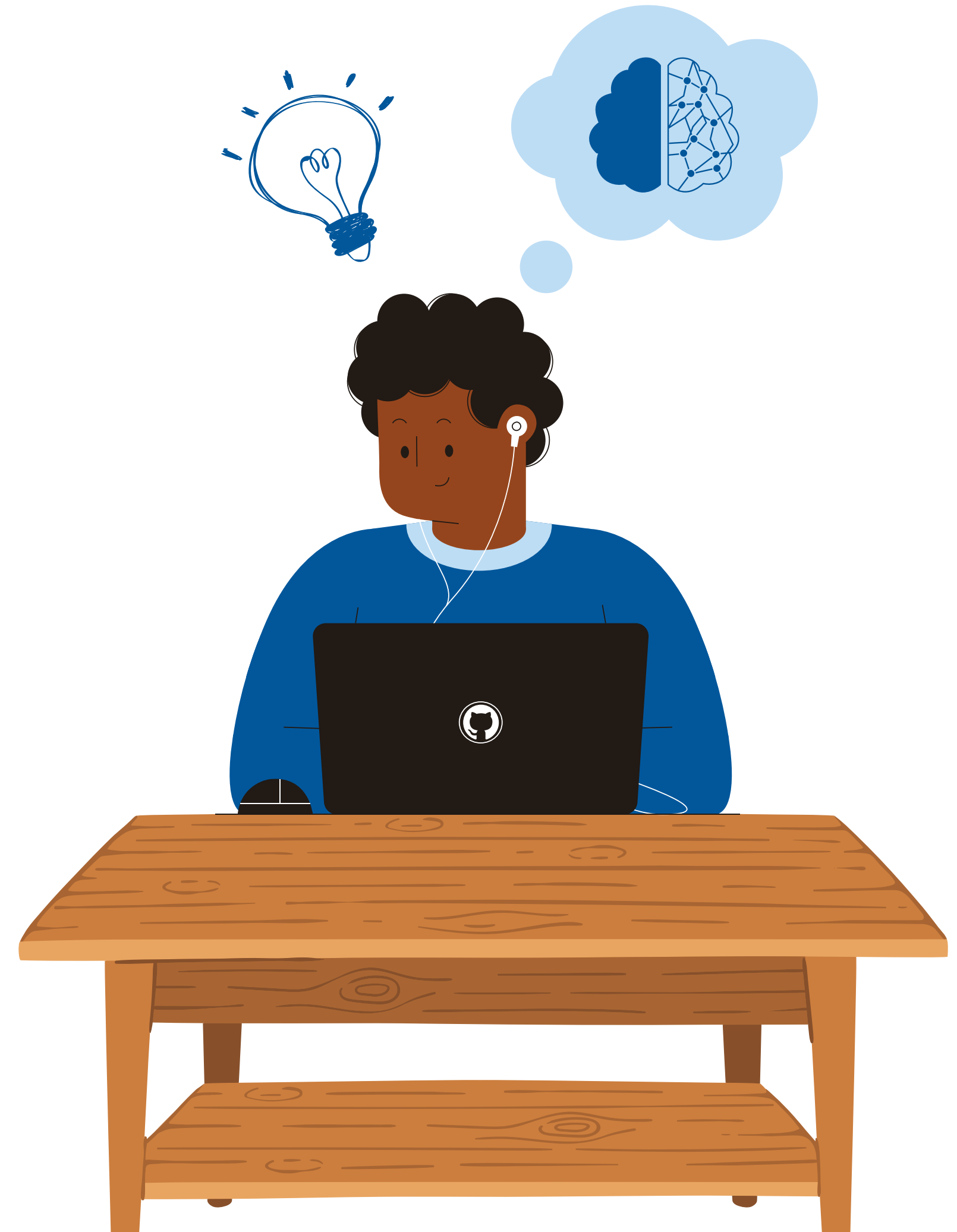
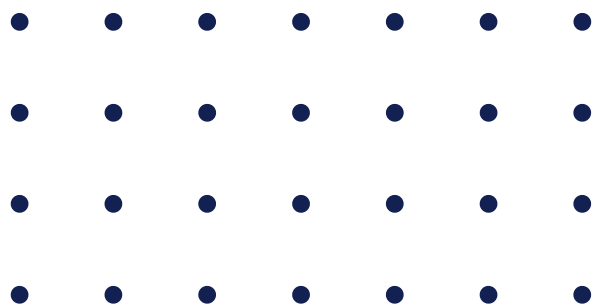
##### após realizar suas alterações realize os passos de commit e push
git add .
git commit -m "Alteração de código"
git push
#ou
git pull origin master
```



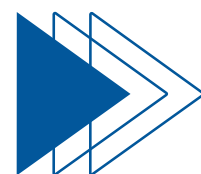
03



Html

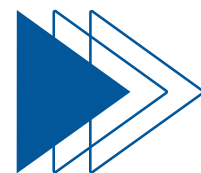
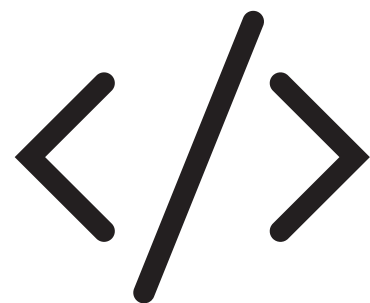


HTML



HTML significa Hypertext Markup Language.

Ele permite que os usuários criem e estruturem seções, parágrafos, cabeçalhos e links para páginas da internet ou aplicações.

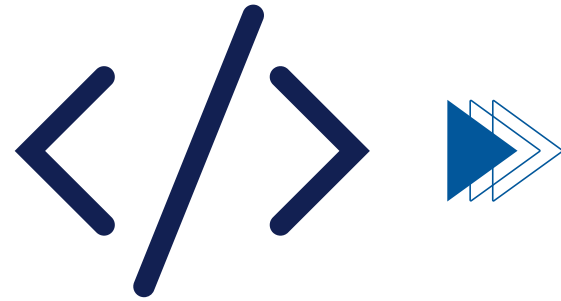
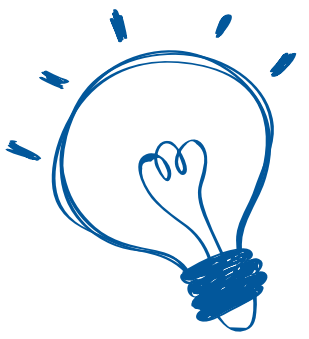


*Ao trabalhar com HTML simplesmente codificamos estruturas (**tags e atributos**) para marcar a página de um site.*

Podemos criar um parágrafo colocando o texto entre as tags `<p>` e `</p>`.



Tags HTML



*A Html funciona baseada em marcações específicas chamadas tags.
Uma tag é um conjunto de palavras entre sinais de colchete angular.*

Conteúdo

<p>

*abertura
de tag*

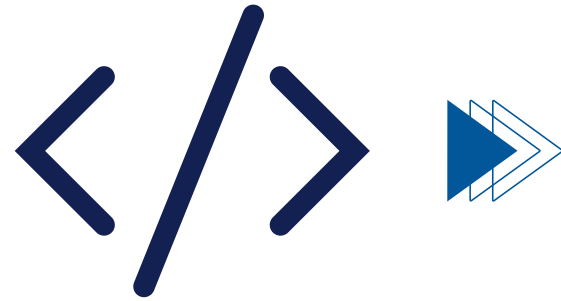
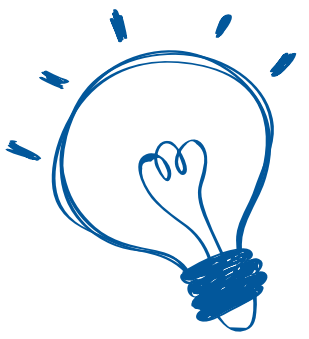
Exemplo de parágrafo

</p>

*fechamento de
tag*



Tags HTML



Além disso, as tags também podem ter atributos e valores, que vão configurar seu comportamento.

parâmetro
conteúdo
**Meu Link **
abertura de tag *valor* *fechamento de tag*



HTML 5

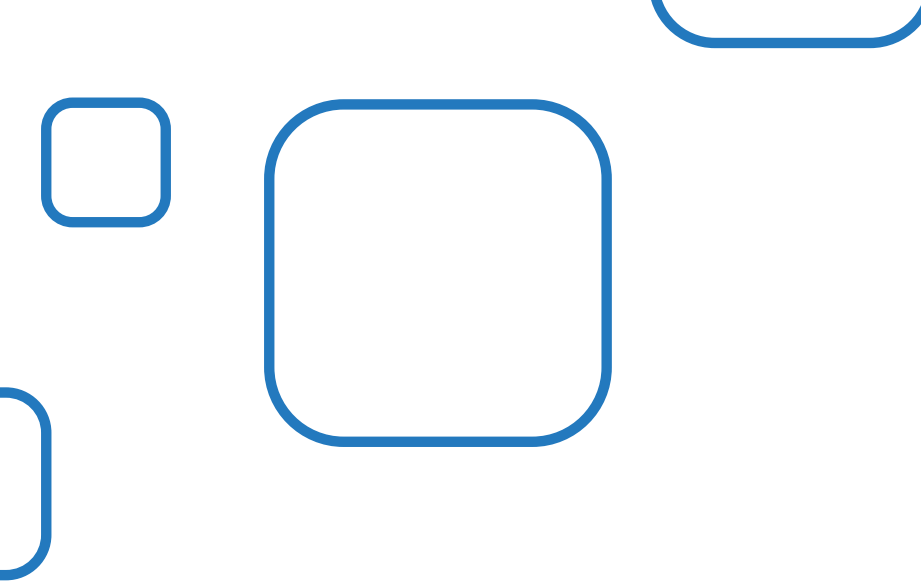


HTML 5

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

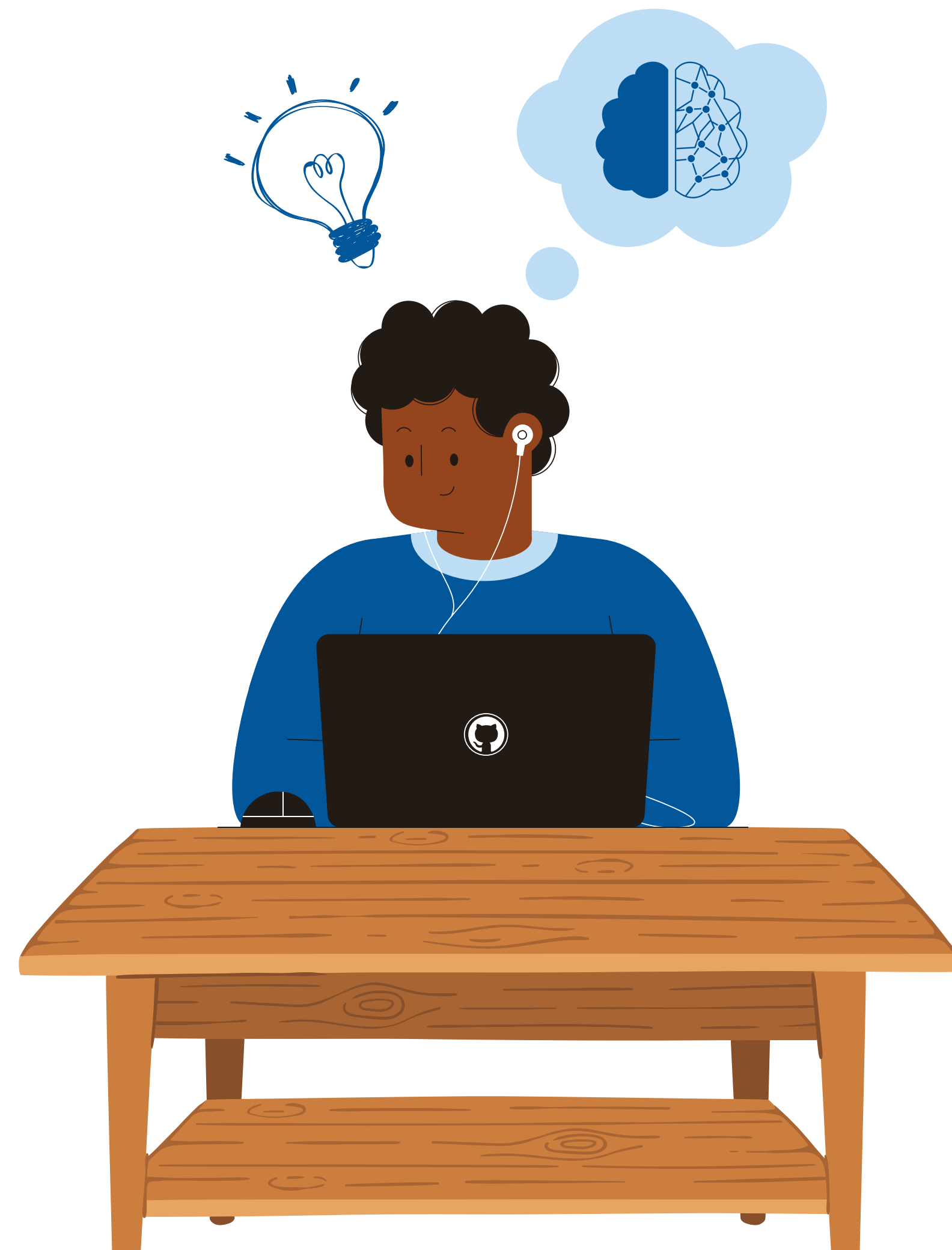
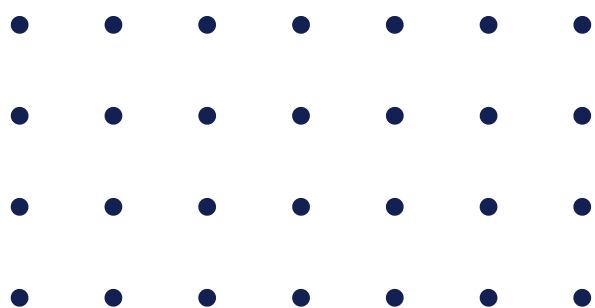




03



Css



CSS



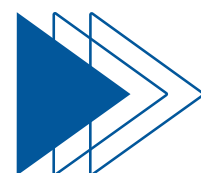
O Cascading Style Sheets (CSS) é uma linguagem utilizada para definir a apresentação (aparência) de documentos que adotam para o seu desenvolvimento linguagens de marcação. O CSS define como serão exibidos os elementos contidos no código de um documento e sua maior vantagem é efetuar a separação entre o formato e o conteúdo de um documento.

● ● ● CSS 3

```
seletor {  
  propriedade: valor;  
}
```



CSS



Existem três formas para incluir o código CSS em seu projeto

- > Inline
- > Interno
- > Externo



```
<p style="color: blue">  
  Parágrafo com fonte azul.  
</p>
```

```
<p>  
  Esse outro parágrafo não é azul, a não ser que exista <span style="color: red">CSS em outro  
  lugar</span>.  
</p>
```



CSS



Existem três formas para incluir o código CSS em seu projeto

- > *Inline*
- > *Interno*
- > *Externo*



```
<head>
  <style type="text/css">
    seletor {
      propriedade: valor;
    }
  </style>
</head>
```



CSS



Existem três formas para incluir o código CSS em seu projeto

- > *Inline*
- > *Interno*
- > *Externo*



```
<head>  
  <link rel="stylesheet" type="text/css" href="/style.css">  
</head>
```



CSS



Seletores: Em HTML e CSS, há a possibilidade de aplicar estilos através de 'class' e 'id'.

```
CSS 3

body {
  background-color: #000;
}

p {
  color: blue;
}

.form {
  width: 200px;
  border: 1px solid #aaa;
  padding: 5px 1px;
  border-radius: 5px;
}

#form02 {
  width: 200px;
  border: 1px solid #aaa;
  padding: 5px 1px;
  border-radius: 5px;
}
```



CSS



Propriedades Mais Utilizadas



Tamanhos e Espaçamentos



```
.classe {  
  width: 50px; /* Largura */  
  height: 50px; /* Altura */  
  border: 1px solid gray; /* Borda */  
  padding: 10px; /* Espaçamento interno */  
  margin: 15px; /* Espaçamento externo */  
}
```



CSS



Propriedades Mais Utilizadas



Backgrounds



```
div {  
  background-color: gray; /* Background como cor */  
  background-image: url("/img/bg.png"); /* Background como imagem */  
}
```



CSS



Propriedades Mais Utilizadas



Tipografias



```
div {  
  font-family: "Times New Roman", Times, serif; /* Mudar a fonte */  
  font-size: 14px; /* Tamanho da fonte */  
  font-weight: bold; /* Fonte negrita */  
  text-decoration: underline; /* Fonte sublinhada */  
  font-style: italic; /* Fonte itálica */  
  text-align: center; /* Centralizar texto */  
}
```



CSS



Propriedades Mais Utilizadas



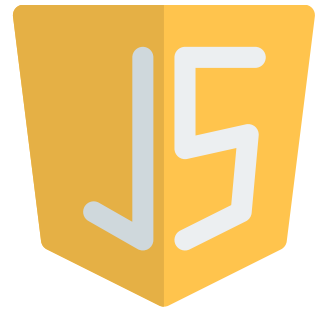
Display



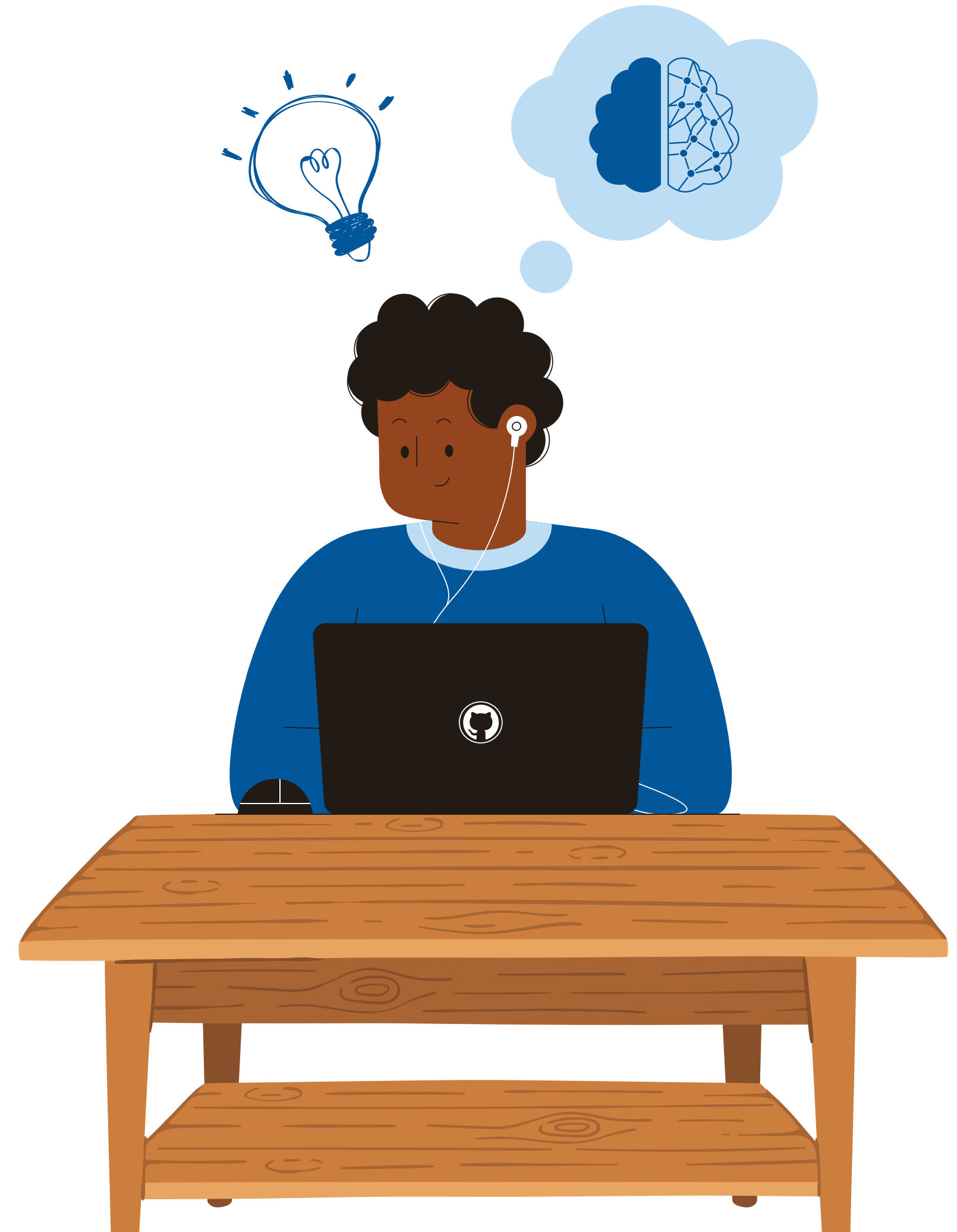
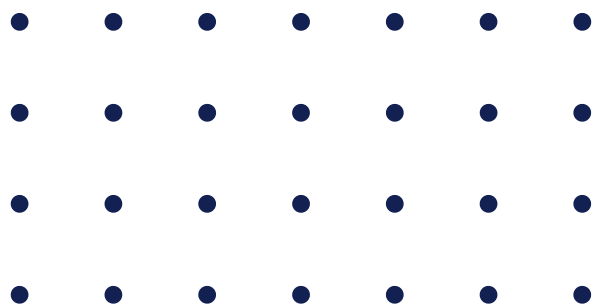
```
div {  
  display: block; /* Blocos */  
  display: inline-block; /* Blocos em linha */  
  display: none; /* Não exibir */  
}
```



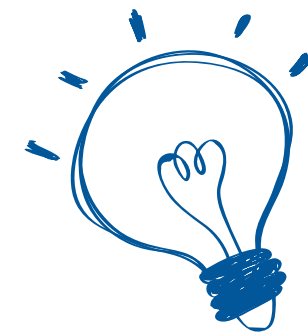
03



JavaScript



JAVASCRIPT



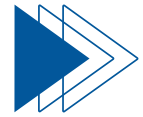
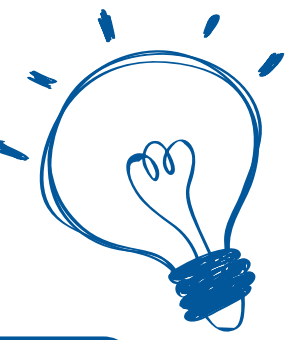
Neste contexto, é uma linguagem de programação client-side. Ela é utilizada para controlar o HTML e o CSS para manipular comportamentos na página.

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <title>Título</title>
    <meta charset="utf-8">
  </head>
  <body>

    <!-- Ter o costume de colocar código JavaScript no final do body -->
    <script type="text/javascript">
      alert('Hello World!');
    </script>
  </body>
</html>
```



JAVASCRIPT



Relacionando um arquivo externo na página



```
<script type="text/javascript" src="/js/app.js"></script>
```



HORA DE PRATICAR



CRIE UM SISTEMA COM HTML, CSS E JAVASCRIPT ONDE O USUÁRIO IRÁ DIGITAR O ANO DE NASCIMENTO E SERÁ EMITIDO UM ALERTA COM A SUA IDADE



CRIE UM SISTEMA PARA CALCULAR O SEU IMC, O USUÁRIO DEVERÁ DIGITAR SEU PESO E ALTURA

$$IMC = \text{peso} / (\text{altura} \times \text{altura})$$



Cálculo IMC

Altura(ex: 1.70)

Peso(ex: 69.2)

Calcular IMC

JAVASCRIPT



API FETCH



API FETCH

```
function buscarCep() {  
  const cep = document.querySelector("#cep").value  
  const url = 'https://brasilapi.com.br/api/cep/v2/'+cep  
  
  fetch(url).then(response => response.json())  
    .then(response => {  
      console.log(response)  
      document.getElementById('estado').value = response.state  
    })  
}
```

