

## AULA 4 – PSEUDO-ELEMENTOS EM CSS

Exemplo:

```
<body>
  <h1>Personalizando Links</h1>
  <ul>
    <li><a href="https://gustavoguanabara.github.io">Repositório do GitHub</a></li>
    <li><a href="https://youtube.com/cursoemvideo/">Canal do YouTube</a></li>
    <li><a href="https://cursoemvideo.com">Site do CursoemVideo</a></li>
  </ul>
</body>
```

### Personalizando Links

- [Repositório do GitHub](https://gustavoguanabara.github.io)
- [Canal do YouTube](https://youtube.com/cursoemvideo/)
- [Site do CursoemVideo](https://cursoemvideo.com)

Por padrão, sem CSS, os links já visitados ficam marcados de roxo. Os não visitados, de azul.

Criando o estilo:

```
<style>
  body {
    font-family: Arial, Helvetica, sans-serif;
  }

  a {
    color: red;
  }
</style>
```

### Personalizando Links

- [Repositório do GitHub](https://gustavoguanabara.github.io)
- [Canal do YouTube](https://youtube.com/cursoemvideo/)
- [Site do CursoemVideo](https://cursoemvideo.com)

Agora todos links ficam marcados de vermelho, sejam eles visitados previamente ou não

### PSEUDO-CLASSE VISITED:

→ É utilizado para **estilizar links** que o usuário **já visitou**. Isso permite que você defina estilos diferentes para links visitados e não visitados

```
a {
  color: red;
  text-decoration: none;
}

a:visited {
  color: darkred;
}

a:hover {
  text-decoration: underline;
  color: black;
}
```

Esse **text-decoration: none** serve para tirar o sublinhado dos links

Evento quando o mouse passa por cima: vai sublinhar e deixar a cor preta

### Personalizando Links

- [Repositório do GitHub](https://gustavoguanabara.github.io)
- [Canal do YouTube](https://youtube.com/cursoemvideo/)
- [Site do CursoemVideo](https://cursoemvideo.com)

Mouse por cima!

### PSEUDO-CLASSE ACTIVE:

→ É usada para **estilizar um elemento** enquanto ele está sendo **ativamente clicado** pelo usuário.

```
a:active {
  color: yellow;
}
```

## Personalizando Links

- Repositório do GitHub
- Canal do YouTube
- Site do CursoemVideo

### PSEUDO-ELEMENTOS:

→ Eles podem mexer nas classes, nos ids, nos elementos, enfim, no conteúdo como um todo

→ São representados por **duas vezes os dois pontos**

Exemplo: se eu quiser que, ao final de todo o link, tenha no final uma indicação

```
a::after {
  content: ' [link] ';
  color: darkgray;
}
```

**Content** significa conteúdo. **::after** indica que estamos estilizando o conteúdo inserido **depois do conteúdo real** do elemento

## Personalizando Links

- Repositório do GitHub [link]
- Canal do YouTube [link]
- Site do CursoemVideo [link]

→ Ou seja, eu não fiz nenhuma alteração no meu conteúdo, porém, eu adicionei coisas (seja ela antes ou depois)

OBS: eu posso criar alguma classe para somente um dos elementos conter esse conteúdo novo

Por exemplo:

```
<body>
<h1>Personalizando Links</h1>
<ul>
<li><a href="https://gustavoguanabara.github.io" class="especial">Repositório do GitHub</a></li>
<li><a href="https://youtube.com/cursoemvideo/">Canal do YouTube</a></li>
<li><a href="https://cursoemvideo.com">Site do CursoemVideo</a></li>
</ul>
```

```
.especial::after {
  content: ' [link] ';
  color: darkgray;
}
```

## Personalizando Links

- Repositório do GitHub [link]
- Canal do YouTube
- Site do CursoemVideo

Somente minha lista onde eu identifiquei uma classe está com pseudo-elemento!

### RESUMO:

- # = id
- . = class
- : = pseudo-class
- :: = pseudo-elemento
- > = children