

Primeiro exemplo prático

Conhecendo o exemplo

O primeiro exemplo é uma listagem de cards sobre cursos.



Publique seu site na internet

Publique seu site na internet

Uma das tarefas mais comuns que você vai se deparar durante sua carreira de programador front-end é publicar o seu site.



React
Componentização de apps com props+useState

Componentização de apps com Props + useState

No React precisaremos passar informações entre componentes...



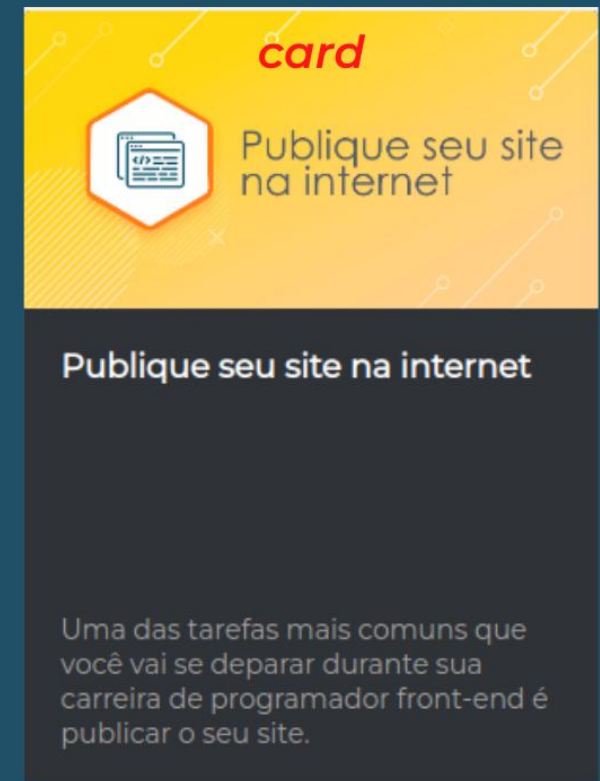
React
Comportamento da página com State Hook e evento onClick

Modificando o comportamento da página com State Hook e evento onClick

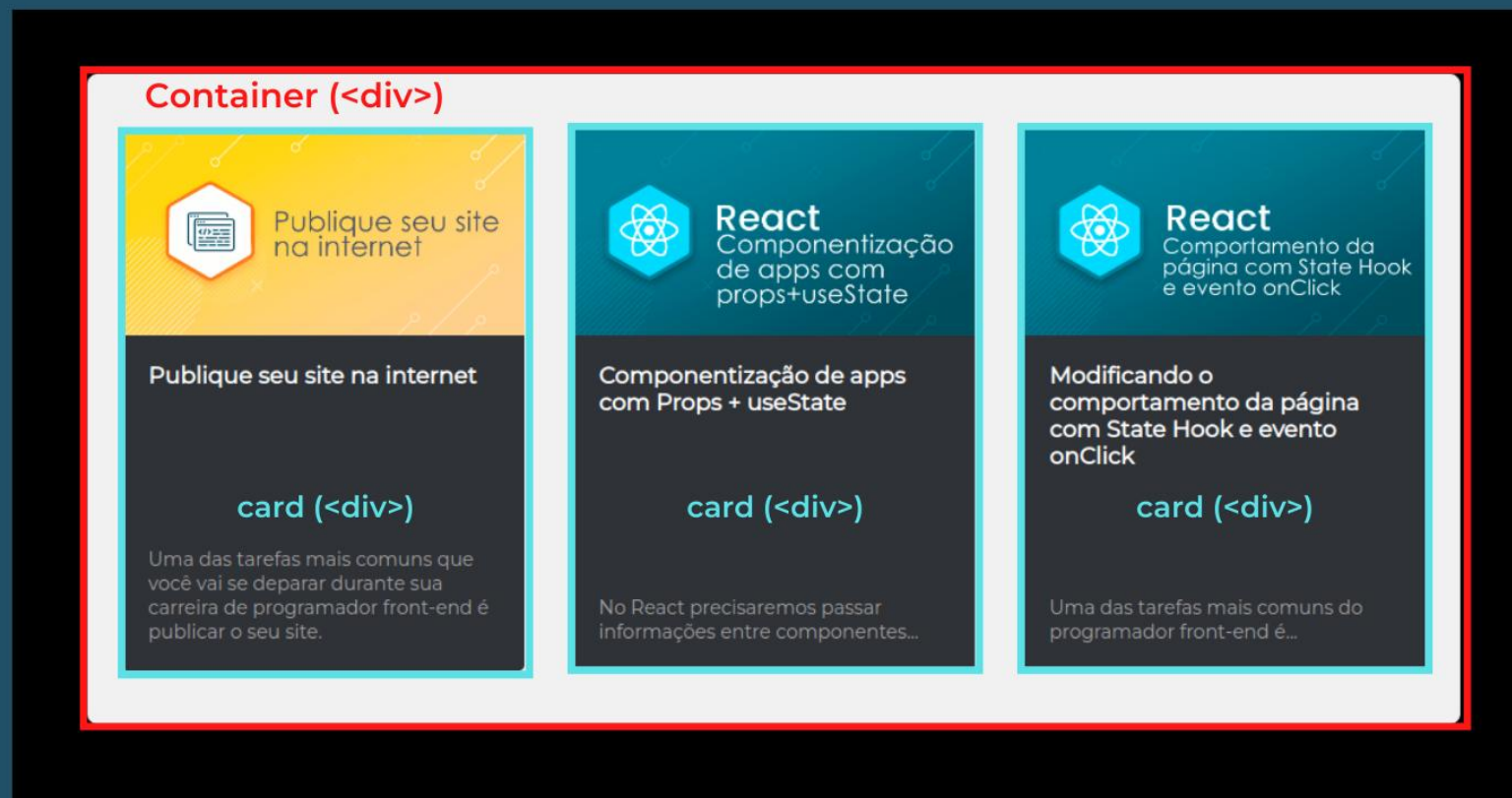
Uma das tarefas mais comuns do programador front-end é...

Veja o código HTML de um dos cards do exemplo:

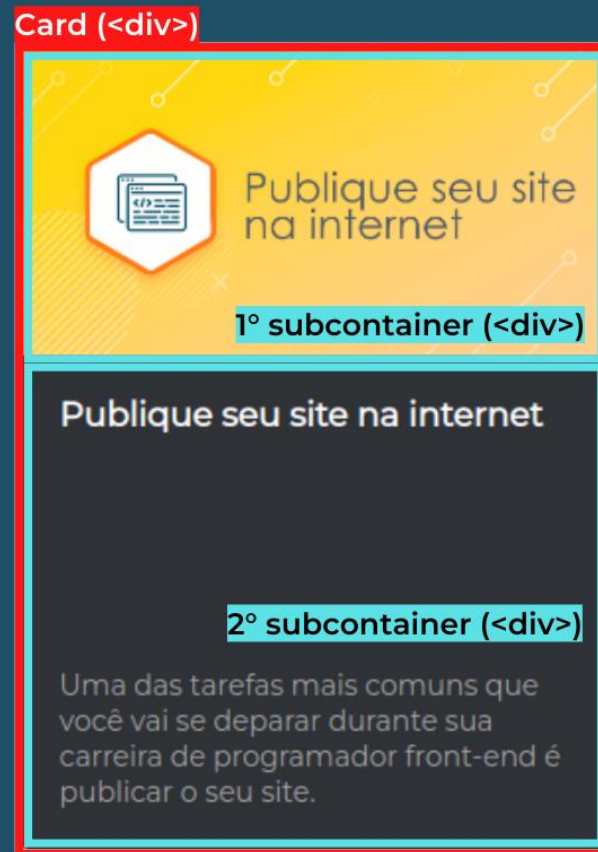
```
<body>
  <div class="container">      index.html
    <div class="card">
      <div class="container-imagem">
        
      </div>
      <div class="container-info">
        <h3>Publique seu site na internet</h3>
        <p>
          Uma das tarefas mais comuns que você vai se deparar durante sua
          carreira de programador front-end é publicar o seu site.
        </p>
      </div>
    </div>
    ...
  </body>
```



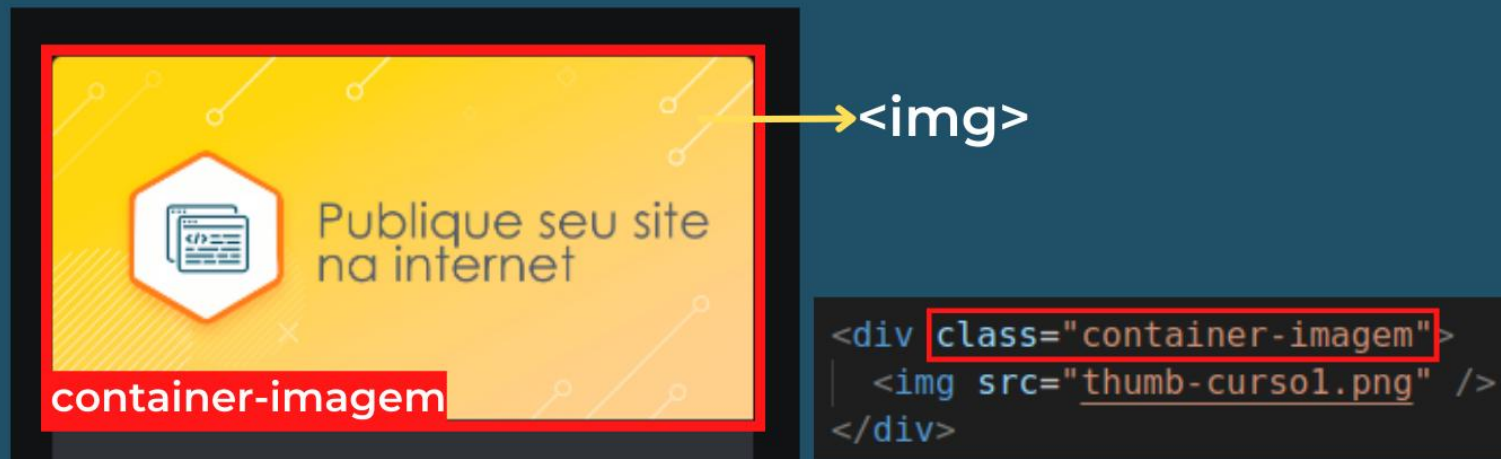
Neste exemplo temos **um container** (`<div>`)
que possui **três cards** (`<div>`).



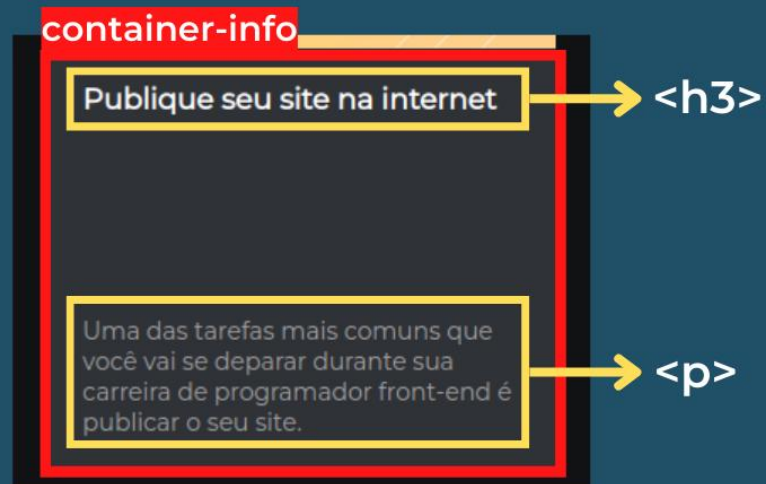
Cada card do exemplo terá **dois subcontainers (<div>)**.



O primeiro subcontainer dos cards terá
uma imagem ().

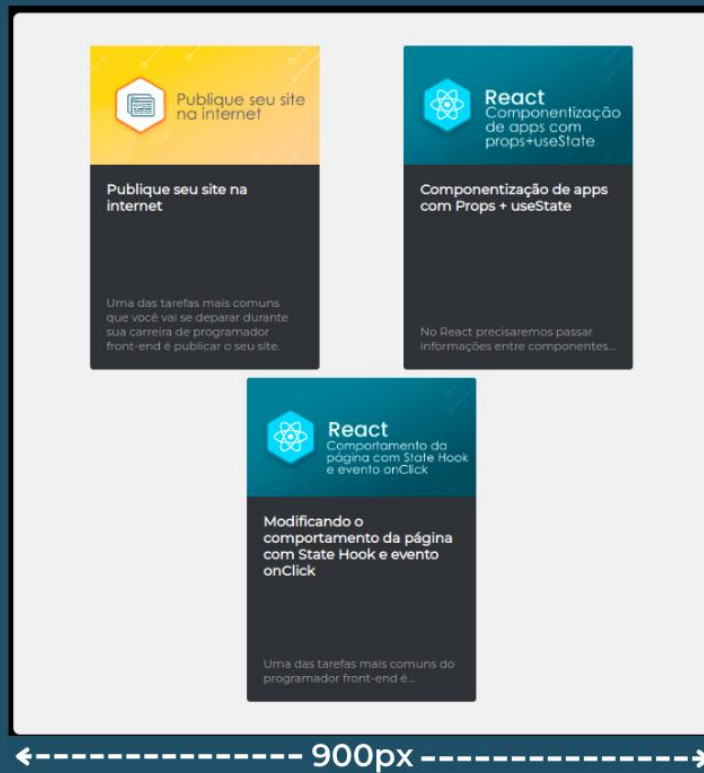


O segundo subcontainer terá um subtítulo (<h3>) e um parágrafo (<p>).

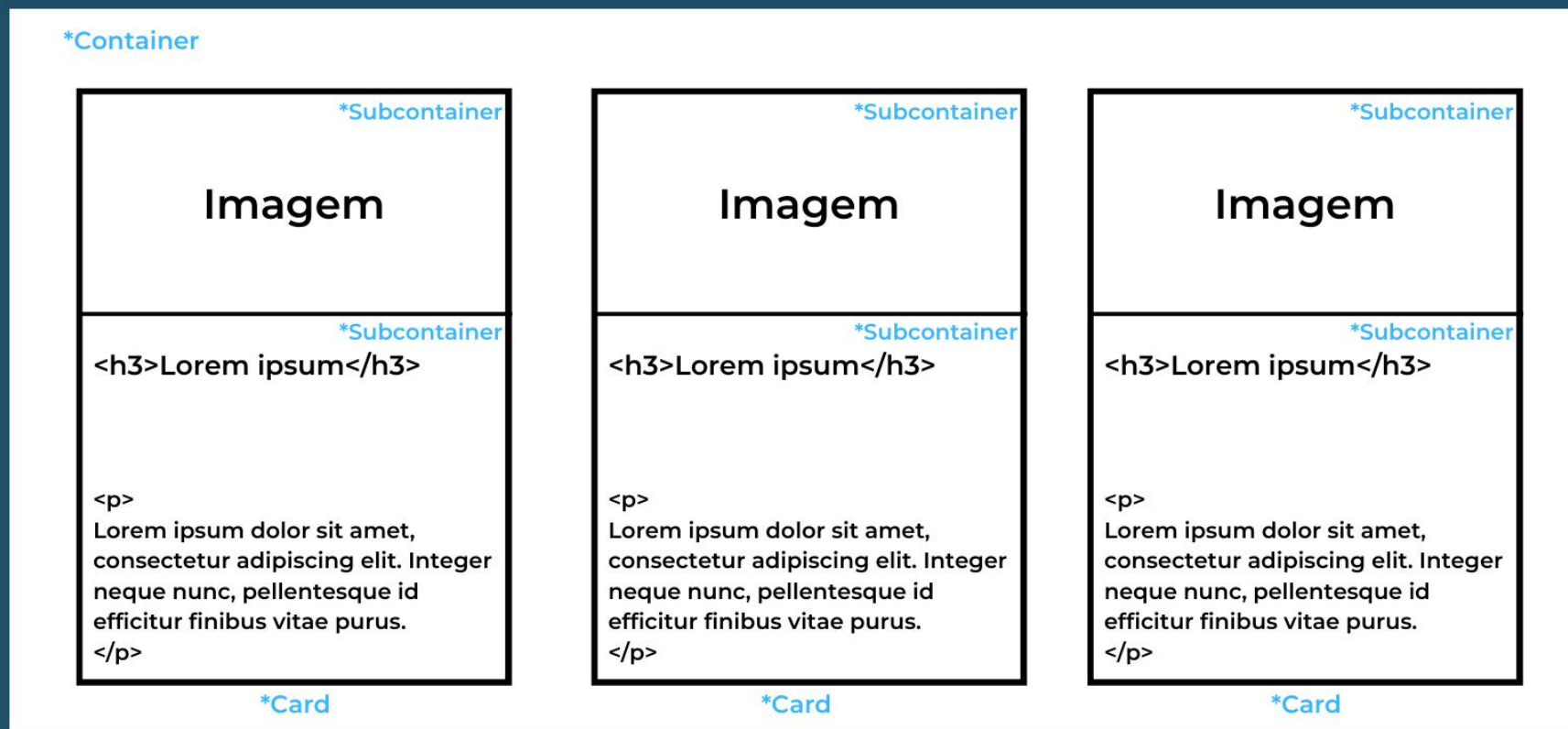


```
<div class="container-info">
  <h3>Publique seu site na internet</h3>
  <p>
    Uma das tarefas mais comuns que você vai se deparar durante sua carreira
    de programador front-end é publicar o seu site.
  </p>
</div>
```

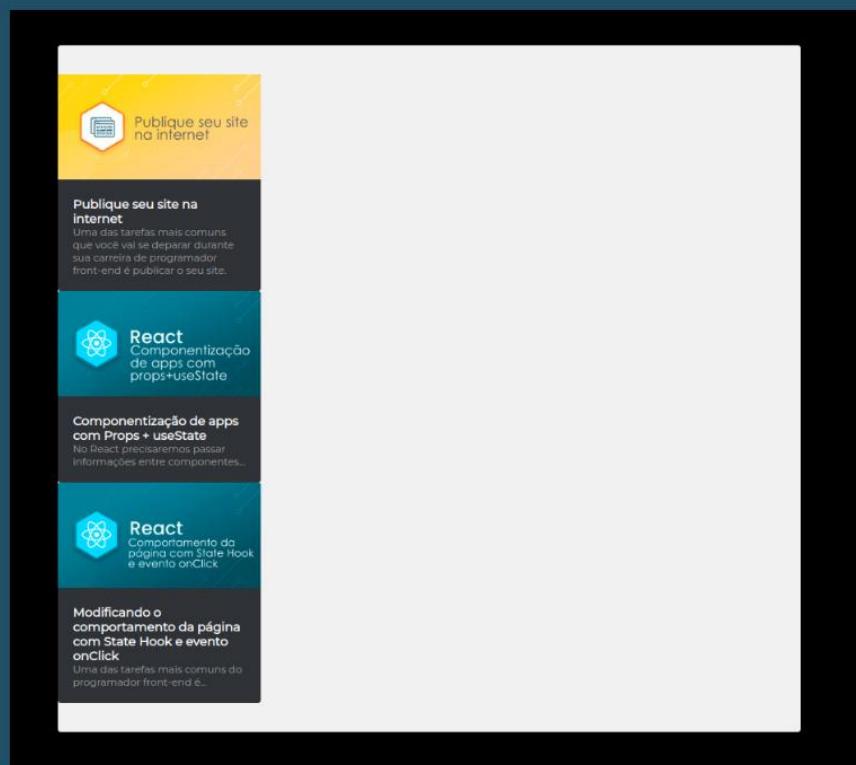
Os cards devem começar a quebrar linha em resoluções a partir de 900px.



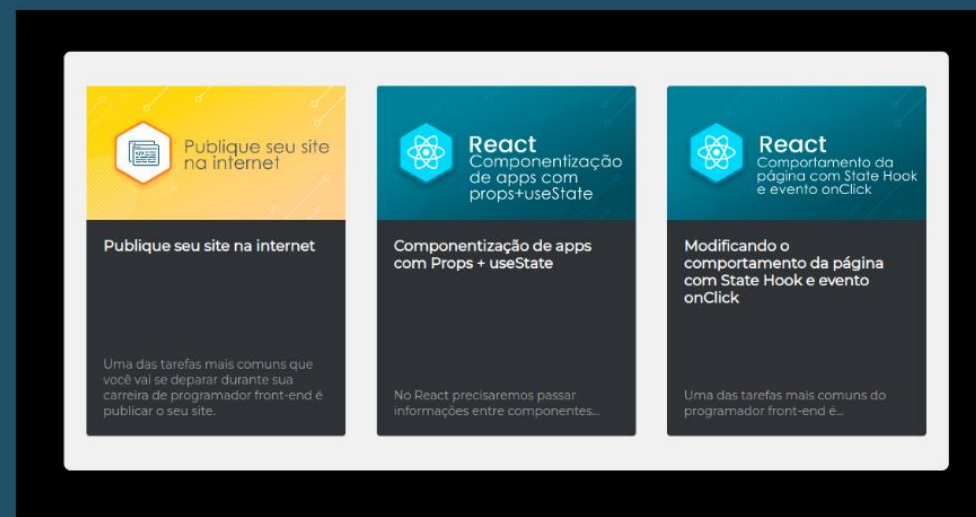
Veja o mesmo exemplo em uma versão wireframe.



Neste exemplo o **Flexbox** será **utilizado** para posicionar os elementos.



Exemplo sem Flexbox

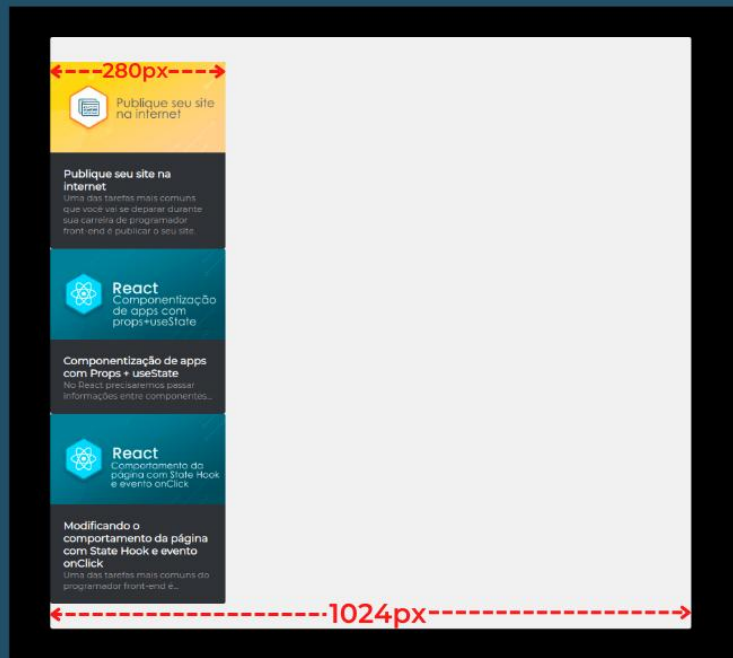


Exemplo com Flexbox

Primeiro exemplo prático

Entendendo o Flexbox do exemplo


Primeiro, veja o exemplo **sem** o código **Flexbox** aplicado.



```
.container {  
  max-width: 1024px;  
  margin: 0 auto;  
  padding: 40px 0px;  
  background-color: #f1f1f1;  
  border-radius: 8px;  
}  
  
.card {  
  width: 280px;  
  background-color: #2f3237;  
  border-radius: 4px;  
}  
  
.container-imagem {  
  height: 150px;  
}  
  
.container-imagem img {  
  width: 100%;  
  height: 100%;  
}  
  
.container-info {  
  padding: 20px;  
}
```

Note que definimos uma **largura de 1024px** para o container e **280px** para os cards.

O objetivo aqui é que no **final desta aula**, o exemplo **seja apresentado** da seguinte forma:



Publique seu site na internet

Publique seu site na internet

Uma das tarefas mais comuns que você vai se deparar durante sua carreira de programador front-end é publicar o seu site.



React
Componentização de apps com props+useState

Componentização de apps com Props + useState

No React precisaremos passar informações entre componentes...



React
Comportamento da página com State Hook e evento onClick

Modificando o comportamento da página com State Hook e evento onClick

Uma das tarefas mais comuns do programador front-end é...

Para isso, começamos **aplicando** o código **Flexbox** no **container**.



```
.container {  
  max-width: 1024px;  
  margin: 0 auto;  
  padding: 40px 0px;  
  background-color: #f1f1f1;  
  border-radius: 8px;  
  display: flex;  
  justify-content: space-evenly;  
}
```

Os cards ficarão em **linha** (*display: flex*) e **espaçados igualmente** entre si (*justify-content: space-evenly*).

Veja que os **elementos** (subcontainers) dos cards já são apresentados **em coluna**.



```
.card {  
  width: 280px;  
  background-color: #2f3237;  
  border-radius: 4px;  
}
```

Esse código não
receberá o Flexbox

Dessa forma, **não precisaremos** aplicar o
Flexbox nos cards.

Definimos que a imagem **ocupe 100%** do espaço do **1º subcontainer**.



```
.container-imagem {  
  height: 150px;  
}  
.container-imagem img {  
  width: 100%;  
  height: 100%;  
}
```

Dessa forma, **não precisaremos** aplicar o Flexbox no 1º subcontainer.

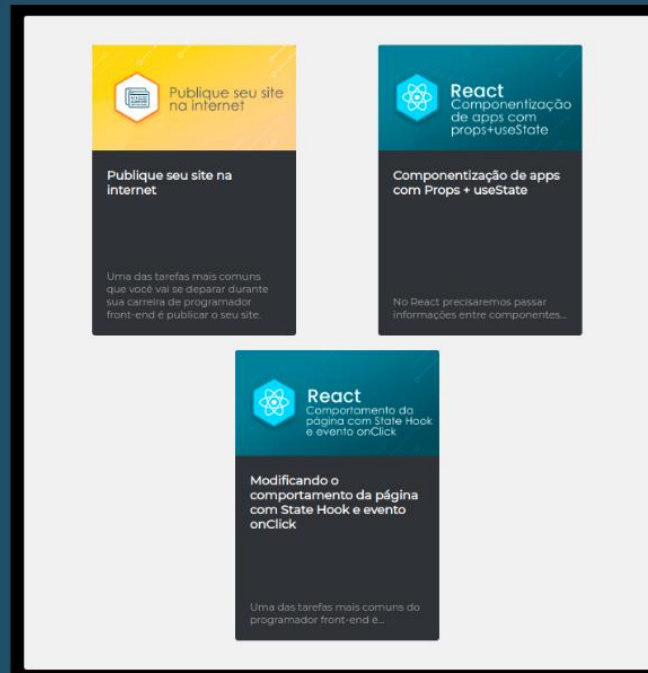
Em seguida, definimos uma **altura** (250px) e o **espaçamento** entre os elementos do 2º subcontainer.



```
.container-info {  
  padding: 20px;  
  height: 250px;  
  display: flex;  
  flex-direction: column;  
  justify-content: space-between;  
}
```

Os elementos ficarão em **coluna** (flex-direction: column) e totalmente **espaçados entre si** (justify-content: space-between).

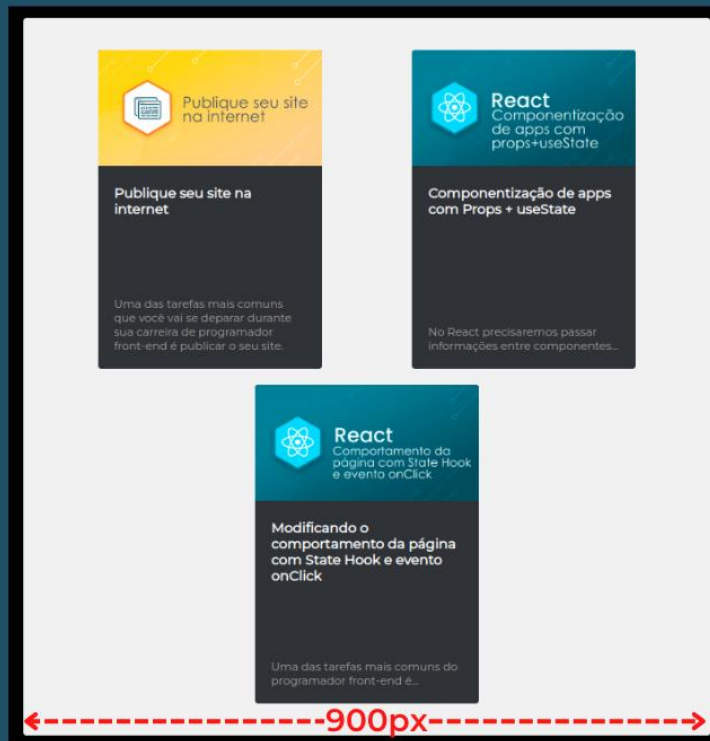
Por fim, definimos a **quebra de linha** do container.



```
.container {  
  max-width: 1024px;  
  margin: 0 auto;  
  padding: 40px 0px;  
  background-color: #f1f1f1;  
  border-radius: 8px;  
  display: flex;  
  justify-content: space-evenly;  
  flex-wrap: wrap;  
  gap: 20px;  
}
```

Para **espaçar** os cards após a **quebra de linha**, aplicamos o *gap: 20px*.

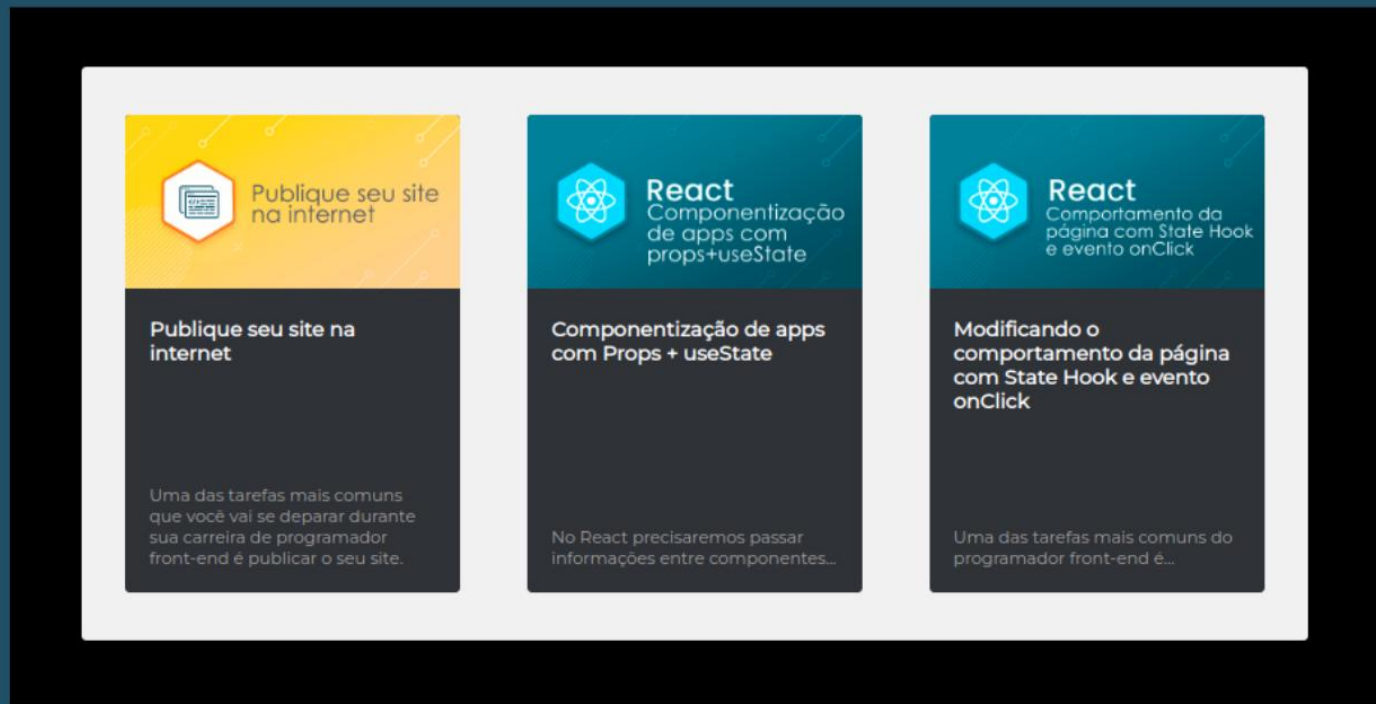
Considerando a **largura** (840px) e o espaço do **gap** (60px) dos cards, a **quebra de linha** ocorrerá em **900px**.



```
.container {  
  max-width: 1024px;  
  margin: 0 auto;  
  padding: 40px 0px;  
  background-color: #f1f1f1;  
  border-radius: 8px;  
  display: flex;  
  justify-content: space-evenly;  
  flex-wrap: wrap;  
  gap: 20px; → 20 x 3 = 60px  
}
```

```
.card {  
  width: 280px; → 280 x 3 = 840px  
  background-color: #2f3237;  
  border-radius: 4px;  
}
```

Veja o resultado do exemplo com o Flexbox aplicado.



```
.container {  
  max-width: 1024px;  
  margin: 0 auto;  
  padding: 40px 0px;  
  background-color: #f1f1f1;  
  border-radius: 8px;  
  display: flex;  
  justify-content: space-evenly;  
  flex-wrap: wrap;  
  gap: 20px;  
}  
  
.card {  
  width: 280px;  
  background-color: #2f3237;  
  border-radius: 4px;  
}  
  
.container-imagem {  
  height: 150px;  
}  
  
.container-imagem img {  
  width: 100%;  
  height: 100%;  
}  
  
.container-info {  
  padding: 20px;  
  height: 250px;  
  display: flex;  
  flex-direction: column;  
  justify-content: space-between;  
}
```


Veja o exemplo em telas menores.

Ao reduzir a largura da tela, o segundo card também irá quebrar linha.

