

## Lista de imagem

As listas de imagem exibem uma coleção de imagens em uma grade de forma organizada.

As listas de imagens representam uma coleção de itens em um padrão repetido. Elas ajudam a melhorar a compreensão visual do conteúdo que elas contêm.

```
{{"component": "modules/components/ComponentLinkHeader.js"}}
```

### Lista de imagem padrão

Listas de imagem padrão são as melhores para itens de igual importância. Elas tem um tamanho uniforme no contêiner, proporção e espaçamento.

```
{{"demo": "StandardImageList.js"}}
```

### Lista de imagem com barras de título

Listas de imagens classificadas enfatizam certos itens sobre outros em uma coleção. Criam uma hierarquia usando tamanhos de contêiner variados e proporção.

```
{{"demo": "QuiltedImageList.js"}}
```

### Lista de imagem entrelaçada

Listas de imagens entrelaçadas usam contêiner de proporção alternados para criar um leiaute rítmico. Uma lista de imagens entrelaçadas é melhor para navegar por conteúdo em pares.

```
{{"demo": "WovenImageList.js"}}
```

### Lista de imagem alternada

Listas de imagens alternadas usam alturas de tamanho dinamicamente dimensionado do contêiner que reflete a proporção de cada imagem. Esta lista de imagens é a melhor utilizada para navegação de conteúdo de pares não cortados.

```
{{"demo": "MasonryImageList.js"}}
```

### Lista de imagem com barra de título

Este exemplo demonstra o uso da `ImageListItemBar` para adicionar uma sobreposição a cada item. A sobreposição pode acomodar um `title`, `subtitle` e ação secundária - neste exemplo utilizamos um `IconButton`.

```
{{"demo": "TitlebarImageList.js"}}
```

#### Barra de título abaixo da imagem (padrão)

A barra de título pode ser colocada abaixo da imagem.

```
{{"demo": "TitlebarBelowImageList.js"}}
```

#### Barra de título abaixo da imagem (alternada)

```
{{"demo": "TitlebarBelowMasonryImageList.js"}}
```

## Lista de imagem customizada

Neste exemplo, os itens tem uma barra de título customizada, posicionada no topo e com um gradiente customizado com `titleBackground`. The secondary action `IconButton` is positioned on the left. The `gap` prop is used to adjust the gap between items.

```
{{"demo": "CustomImageList.js", "defaultCodeOpen": false}}
```