

## Para saber mais: a diretiva v-for e a propriedade key

Quando Vue está atualizando uma lista de elementos renderizados com v-for, ele usa, por padrão, uma estratégia de patch "no local". Se a ordem dos itens de dados for alterada, ao invés de mover os elementos DOM para coincidir com a ordem dos itens, o Vue simplesmente corrigirá cada elemento no local e verificará se ele reflete o que deve ser renderizado nesse índice específico.

Seu modo padrão é eficiente, mas somente adequado quando a saída de renderização da lista não depende do estado do componente filho ou do estado DOM temporário (por exemplo, valores de entrada do formulário).

Para dar Vue uma dica para que ele possa acompanhar a identidade de cada nó e, portanto, reutilizar e reordenar elementos existentes, você precisará fornecer um atributo de chave exclusivo para cada item. Um valor ideal para chave seria o id exclusivo de cada item.

Vejamos um exemplo presente em nosso componente `Home.vue` :

```
<!-- código anterior omitido -->
```

```
<li class="lista-fotos-item" v-for="foto of fotosComFiltro">
```

```
<!-- código posterior omitido -->
```

COPIAR CÓDIGO

Agora, com a propriedade `key` :

```
<!-- código anterior omitido -->
```

```
<li class="lista-fotos-item" v-for="foto of fotosComFiltro"
```

```
:key="foto._id">
```

```
<!-- código posterior omitido -->
```

[COPIAR CÓDIGO](#)

É recomendável fornecer sempre uma chave com v-for. Inclusive novas versões do Vue.js exibirão um aviso no console caso você não tenha definido uma chave.