Flexbox Sample

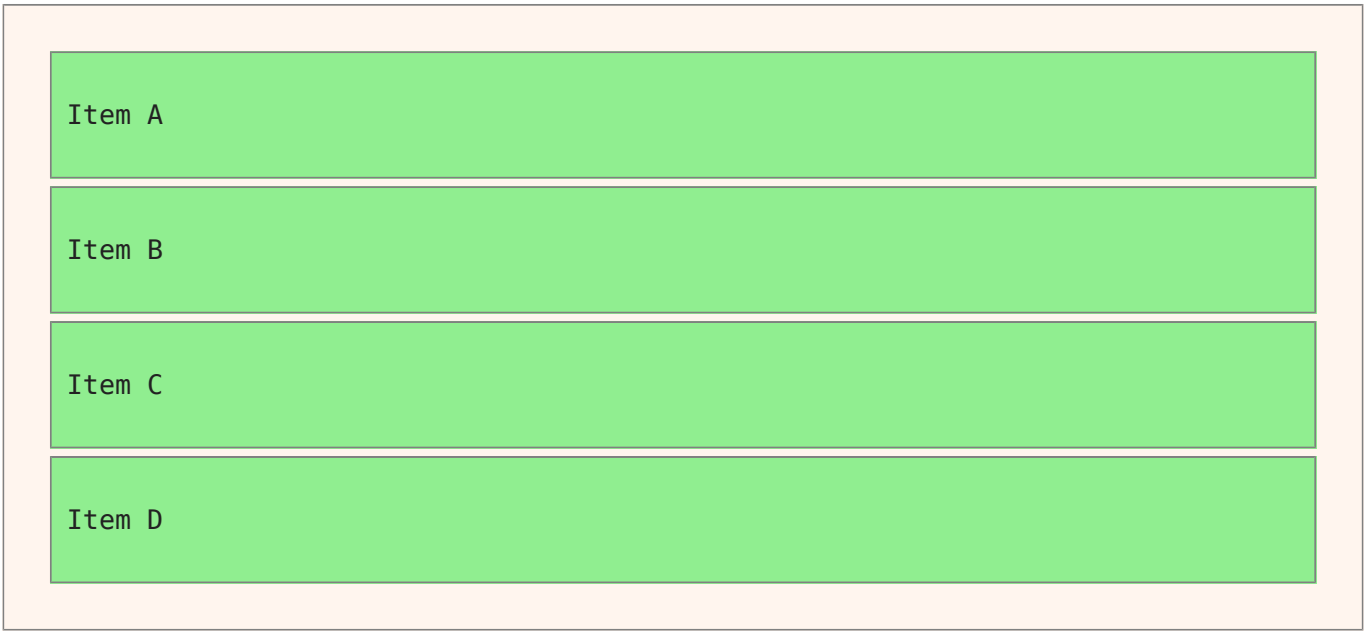
Create a flexbox layout

Flexbox layout is an easy way to arrange multiple items inside a container.

To create a flex container, simply apply `display: flex` to the desired container.

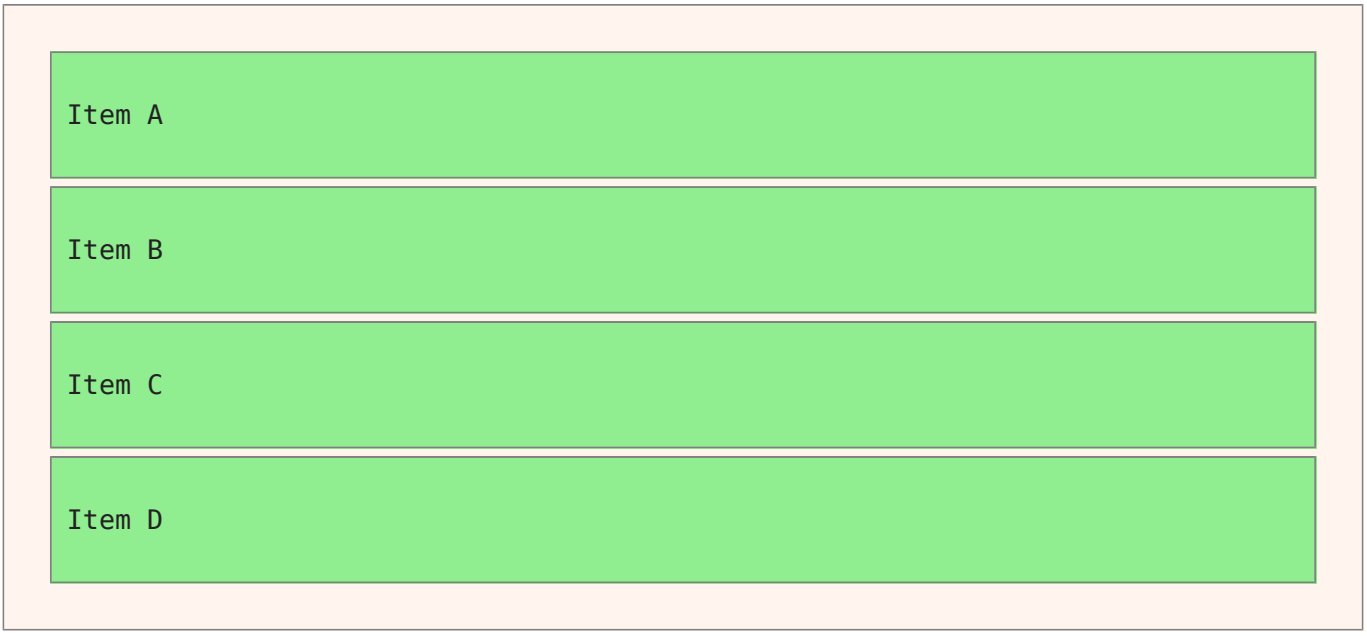
```
.flex { display: flex; }
```

The children of a flex container are called flex items. They will flow horizontally by default, so they will be positioned in a row:



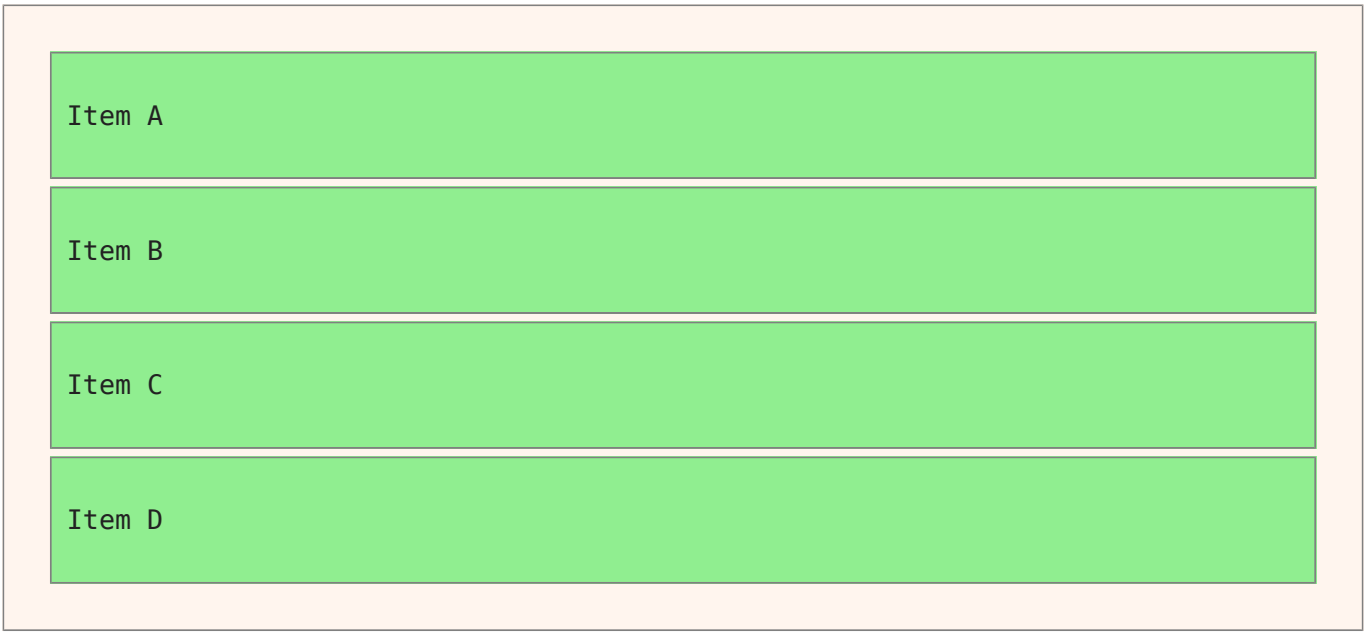
The flow direction can be modified using the `flex-direction` property. Setting `flex-direction: column` will cause the flex items to flow vertically in a column:

```
.flex { display: flex; flex-direction: column; }
```



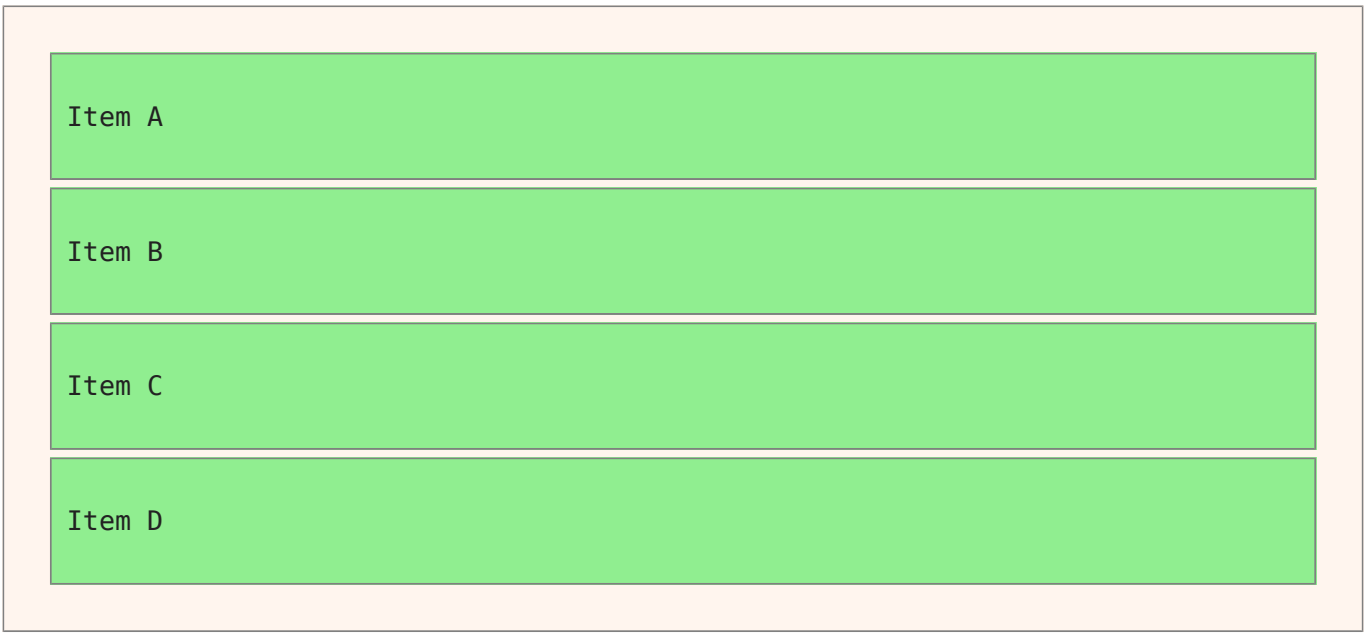
Setting `flex-direction: row-reverse` will cause the flex items to flow in a row in reverse order:

```
.flex { display: flex; flex-direction: row-reverse; }
```



As we saw in the example above, `flex-direction: row-reverse` reverses the item order and aligns the items to the right. Let's align them to the left instead:

```
.flex { display: flex; flex-direction: row-reverse; justify-content: flex-end; }
```



If the flex items are arranged in a column, they can be aligned to the bottom of the flex container using `justify-content: flex-end`:

```
.flex { display: flex; flex-direction: column; justify-content: flex-end; }
```

Item A

Item B

Item C

Item D

Flex items can be centered by using both `align-items` and `justify-content`:

```
.flex { display: flex; flex-direction: row; align-items: center; justify-content: center; }
```

Item A

Item B

Item C

Item D

Flex items can also be justified.

```
.flex { display: flex; justify-content: space-between; }
```

Item A

Item B

Item C

Item D

Growing flex items

Flex items grow to fill the row when the `flex-grow` property is set to a positive value:

```
.flex { display: flex; } .flex div { flex-grow: 1; }
```

Item A

Item B

Item C

Item D

Different values of `flex-grow` for items in the same row cause different amounts of growth:

```
.flex { display: flex; } .flex div { flex-grow: 1; } .flex div:nth-of-type(3) { flex-grow: 3; }
```

Item A

Item B

Item C

Item D

Wrapping flex items in rows or columns

You can wrap flex items in multiple rows by specifying `flex-wrap: wrap` or `flex-flow: row wrap`, which is a shorthand for `flex-direction: row; flex-wrap: wrap`:

```
.flex { display: flex; flex-flow: row wrap; }
```

Item A

Item B

Item C

Item D

Item E

Item F

Item G

Item H

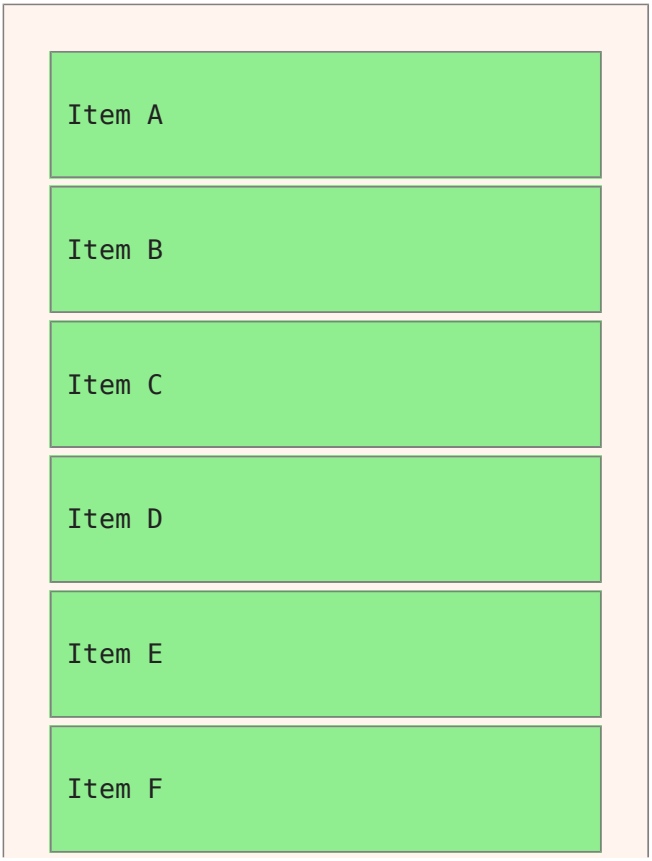
The flow of the flex items can also be reversed:

```
.flex { display: flex; flex-flow: row-reverse wrap; }
```

Item A

Flex items can also be wrapped into columns:

```
.flex { display: flex; flex-flow: column wrap; align-content: flex-start; }
```



When no height is set on the flex container the columns wrap at the bottom edge of the page:

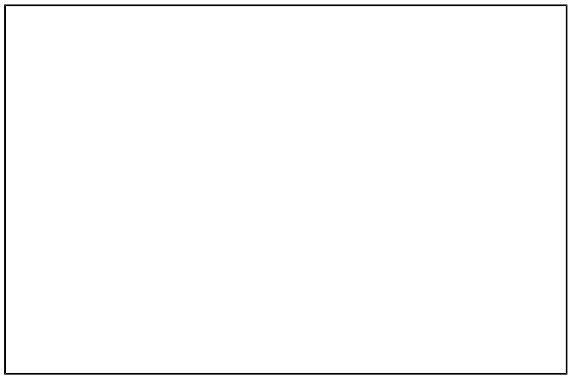
```
.flex { display: flex; flex-flow: column wrap; align-content: flex-start; }
```

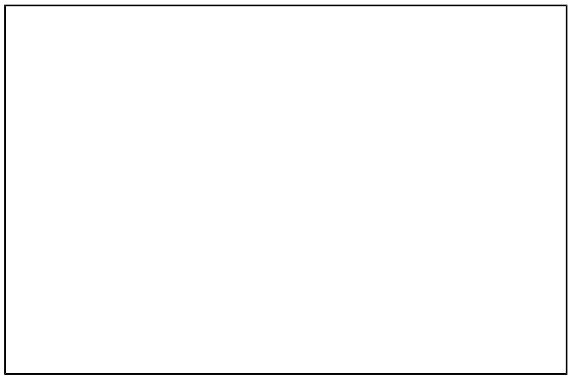
Flex items and pagination

When PDFreactor paginates documents with flex layouts, the flex items are paginated according to the wrap layout mode specified for the flex container.

If the container specifies `flex-flow: row wrap` and a page break occurs between two rows, the rows are continued on the next page:

```
.flex { display: flex; flex-flow: row wrap; }
```





If the container specifies `flex-flow: column wrap` and a page break occurs between two columns, the columns are continued on the next page:

```
.flex { display: flex; flex-flow: column wrap; }
```

Item K