



Search...

Ctrl+K



Image List

The Image List displays a collection of images in an organized grid.



For Figma - A large UI kit with over 600 handcrafted Material UI, MUI X, Joy UI components 🎨.

ad by MUI

Image lists represent a collection of items in a repeated pattern. They help improve the visual comprehension of the content they hold.

[View as Markdown](#)[Feedback](#)[Bundle size](#)[Source](#)[Material Design](#)[Figma](#)[Sketch](#)

Standard image list



Standard image lists are best for items of equal importance. They have a uniform container size, ratio, and spacing.

[Edit in Chat](#)

JS

TS

[Collapse code](#)

```
import ImageList from '@mui/material/ImageList';
import ImageListItem from '@mui/material/ImageListItem';

export default function StandardImageList() {
  return (
    <ImageList sx={{ width: 500, height: 450 }} cols={3} rowHeight={164}>
      {itemData.map((item) => (
        <ImageListItem key={item.img}>
          <img
            srcSet={`${item.img}?w=164&h=164&fit=crop&auto=format&dpr=2 2x`}
            src={`${item.img}?w=164&h=164&fit=crop&auto=format`}
            alt={item.title}
            loading="lazy"
          />
        </ImageListItem>
      ))}
    </ImageList>
  );
}

const itemData = [
{
  img: 'https://images.unsplash.com/photo-1551963831-b3b1ca40c98e',
  title: 'Breakfast',
},
{
  img: 'https://images.unsplash.com/photo-1551782450-a2132b4ba21d',

```

```
    title: 'Burger',
},
{
  img: 'https://images.unsplash.com/photo-1522770179533-24471fcdba45',
  title: 'Camera',
},
{
  img: 'https://images.unsplash.com/photo-1444418776041-9c7e33cc5a9c',
  title: 'Coffee',
},
{

```

Quilted image list



Quilted image lists emphasize certain items over others in a collection. They create hierarchy using varied container sizes and ratios.



Edit in Chat

JS

TS

Collapse code



```
import ImageList from '@mui/material/ImageList';
import ImageListItem from '@mui/material/ImageListItem';

function srcset(image: string, size: number, rows = 1, cols = 1) {
  return {
```

```

src: `${image}?w=${size * cols}&h=${size * rows}&fit=crop&auto=format` ,
srcSet: `${image}?w=${size * cols}&h=${size * rows}&fit=crop&auto=format&dpr=2 2x` ,
    size * rows
}};

}

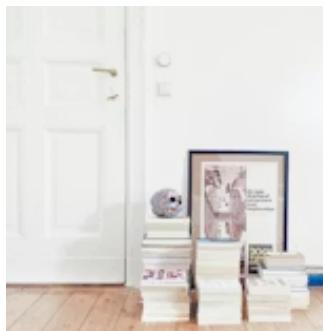
export default function QuiltedImageList() {
  return (
    <ImageList
      sx={{ width: 500, height: 450 }}
      variant="quilted"
      cols={4}
      rowHeight={121}
    >
      {itemData.map((item) => (
        <ImageListItem key={item.img} cols={item.cols || 1} rows={item.rows || 1}>
          <img
            {...srcset(item.img, 121, item.rows, item.cols)}
            alt={item.title}
            loading="lazy"
          />
        </ImageListItem>
      ))}
    </ImageList>
  );
}

const itemData = [
{
  img: 'https://images.unsplash.com/photo-1551963831-b3b1ca40c98e',
  title: 'Breakfast',
  rows: 2
}
]

```

Woven image list

Woven image lists use alternating container ratios to create a rhythmic layout. A woven image list is best for browsing peer content.

[Edit in Chat](#)

JS

TS

[Collapse code](#)

```
import ImageList from '@mui/material/ImageList';
import ImageListItem from '@mui/material/ImageListItem';

export default function WovenImageList() {
  return (
    <ImageList sx={{ width: 500, height: 450 }} variant="woven" cols={3} gap={8}>
      {itemData.map((item) => (
        <ImageListItem key={item.img}>
          <img
            srcSet={`${item.img}?w=161&fit=crop&auto=format&dpr=2 2x`}
            src={`${item.img}?w=161&fit=crop&auto=format`}
            alt={item.title}
            loading="lazy"
          />
        </ImageListItem>
      ))}
    </ImageList>
  );
}

const itemData = [
  {
    img: 'https://images.unsplash.com/photo-1549388604-817d15aa0110',
    title: 'Bed',
  },
  {
    img: 'https://images.unsplash.com/photo-1563298723-dcfbeaa392e3',
    title: 'Kitchen',
  },
]
```

```
{  
  img: 'https://images.unsplash.com/photo-1523413651479-597eb2da0ad6',  
  title: 'Sink',  
},  
{  
  img: 'https://images.unsplash.com/photo-1525097487452-6278ff080c31',  
  title: 'Books',  
},  
{
```

Masonry image list

Masonry image lists use dynamically sized container heights that reflect the aspect ratio of each image. This image list is best used for browsing uncropped peer content.

 Edit in Chat JS TS

Collapse code 



```
import Box from '@mui/material/Box';  
import ImageList from '@mui/material/ImageList';  
import ImageListItem from '@mui/material/ImageListItem';  
  
export default function MasonryImageList() {  
  return (  
    <Box sx={{ width: 500, height: 450, overflowY: 'scroll' }}>  
      <ImageList variant="masonry" cols={3} gap={8}>  
        {itemData.map((item) => (
```

```

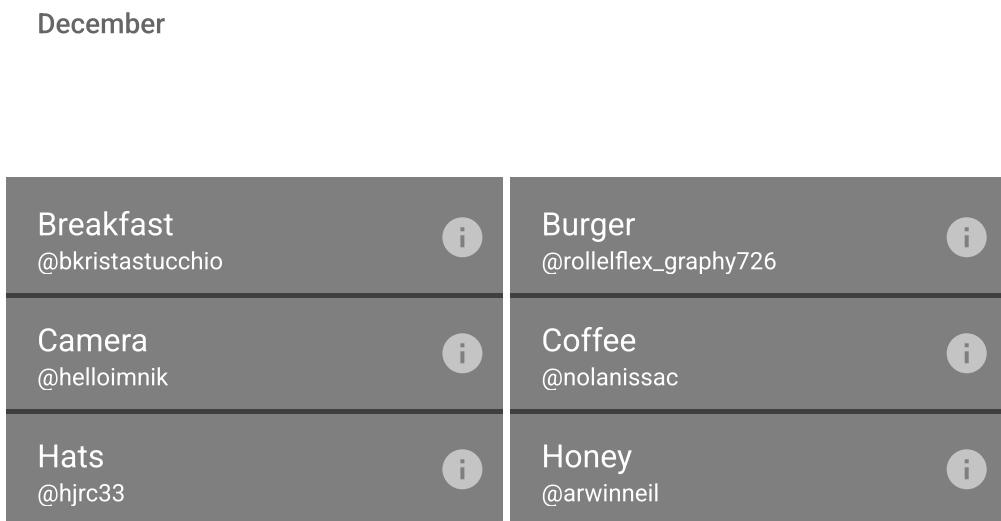
<ImageListItem key={item.img}>
  <img
    srcSet={` ${item.img}?w=248&fit=crop&auto=format&dpr=2 2x`}
    src={` ${item.img}?w=248&fit=crop&auto=format`}
    alt={item.title}
    loading="lazy"
  />
</ImageListItem>
))}>
</ImageList>
</Box>
);
}

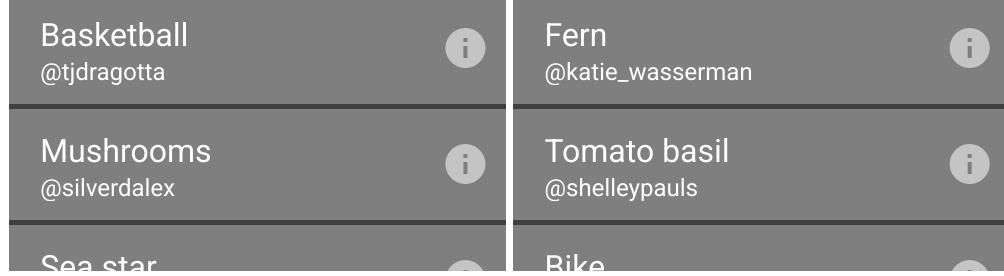
const itemData = [
{
  img: 'https://images.unsplash.com/photo-1549388604-817d15aa0110',
  title: 'Bed',
},
{
  img: 'https://images.unsplash.com/photo-1525097487452-6278ff080c31',
  title: 'Books',
},
{
  img: 'https://images.unsplash.com/photo-1523413651479-597eb2da0ad6',
  title: 'Sink',
},
{
  img: 'https://images.unsplash.com/photo-1563298723-dcfebaa392e3',

```

Image list with title bars

This example demonstrates the use of the `ImageListItemBar` to add an overlay to each item. The overlay can accommodate a `title`, `subtitle` and secondary action - in this example an `IconButton`.





Edit in Chat JS TS

Hide code



```
import ImageList from '@mui/material/ImageList';
import ImageListItem from '@mui/material/ImageListItem';
import ImageListItemBar from '@mui/material/ImageListItemBar';
import ListSubheader from '@mui/material/ListSubheader';
import IconButton from '@mui/material/IconButton';
import InfoIcon from '@mui/icons-material/Info';

export default function TitlebarImageList() {
  return (
    <ImageList sx={{ width: 500, height: 450 }}>
      <ImageListItem key="Subheader" cols={2}>
        <ListSubheader component="div">December</ListSubheader>
      </ImageListItem>
      {itemData.map((item) => (
        <ImageListItem key={item.img}>
          <img
            srcSet={`${item.img}?w=248&fit=crop&auto=format&dpr=2 2x`}
            src={`${item.img}?w=248&fit=crop&auto=format`}
            alt={item.title}
            loading="lazy"
          />
          <ImageListItemBar
            title={item.title}
            subtitle={item.author}
            actionIcon={
              <IconButton
                sx={{ color: 'rgba(255, 255, 255, 0.54)' }}
                aria-label={`info about ${item.title}`}
              >
                <InfoIcon />
              </IconButton>
            }
          />
        </ImageListItem>
      )))
    </ImageList>
  );
}
```

Title bar below image (standard)

The title bar can be placed below the image.

Breakfast
by: @bkristastucchio

Burger
by: @rollelflex_graphy726

Camera
by: @helloimnik

Coffee
by: @nolanissac

Hats
by: @hjrc33

Honey
by: @arwinneil

Basketball
by: @tjdragotta

Fern
by: @katie_wasserman

Mushrooms
by: @silverdalex

Tomato basil
by: @shelleypauls

Sea star
by: @peterlaster

Bike
by: @southside_customs

 Edit in Chat

JS TS

Hide code     

```
import ImageList from '@mui/material/ImageList';
import ImageListItem from '@mui/material/ImageListItem';
import ImageListItemBar from '@mui/material/ImageListItemBar';

export default function TitlebarBelowImageList() {
  return (
    <ImageList sx={{ width: 500, height: 450 }}>
      {itemData.map((item) => (
        <ImageListItem key={item.img}>
          <img
            srcSet={`${item.img}?w=248&fit=crop&auto=format&dpr=2 2x`}
            src={`${item.img}?w=248&fit=crop&auto=format`}
            alt={item.title}
            loading="lazy"
          />
          <ImageListItemBar
            title={item.title}
            subtitle={<span>by: {item.author}</span>}
            position="below"
          />
        </ImageListItem>
      )));
    </ImageList>
  );
}

const itemData = [
```

```
{  
  img: 'https://images.unsplash.com/photo-1551963831-b3b1ca40c98e',  
  title: 'Breakfast',  
  author: '@bkristastucchio',  
},  
{  
  img: 'https://images.unsplash.com/photo-1551782450-a2132b4ba21d',  
  title: 'Burger',  
  author: '@rollelflex_graphy726',  
},  
{
```

Title bar below image (masonry)

| | | |
|------------------|-------------------|------------------|
| swabdesign | Darren Richardson | Jen P. |
| Pavel Nekoranec | Taylor Simpson | Douglas Sheppard |
| Charles Deluvio | Ben Kolde | Fi Bell |
| Christian Mackie | Philipp Berndt | Hutomo Abrianto |

 Edit in Chat  

[Collapse code](#)



```
import Box from '@mui/material/Box';  
import ImageList from '@mui/material/ImageList';  
import ImageListItem from '@mui/material/ImageListItem';  
import ImageListItemBar from '@mui/material/ImageListItemBar';  
  
export default function TitlebarBelowMasonryImageList() {  
  return (  
    <Box sx={{ width: 500, height: 450, overflowY: 'scroll' }}>  
      <ImageList variant="masonry" cols={3} gap={8}>  
        {itemData.map((item) => (  
          <ImageListItem key={item.img}>  
            <img
```

```
        srcSet={`${item.img}?w=248&fit=crop&auto=format&dpr=2 2x`}
        src={`${item.img}?w=248&fit=crop&auto=format`}
        alt={item.title}
        loading="lazy"
      />
      <ImageListItemBar position="below" title={item.author} />
    </ImageListItem>
  )));
</ImageList>
</Box>
);
}

const itemData = [
{
  img: 'https://images.unsplash.com/photo-1549388604-817d15aa0110',
  title: 'Bed',
  author: 'swabdesign',
},
{
  img: 'https://images.unsplash.com/photo-1525097487452-6278ff080c31',
  title: 'Books',
  author: 'Pavel Nekoranec',
},
{
  img: 'https://images.unsplash.com/photo-1523413651479-597eb2da0ad6',

```

Custom image list

In this example the items have a customized titlebar, positioned at the top and with a custom gradient `titleBackground`. The secondary action `IconButton` is positioned on the left. The `gap` prop is used to adjust the gap between items.

☆ Breakfast

☆ Burger

☆ Camera

[Edit in Chat](#)

JS

TS

[Hide code](#)



```
import ImageList from '@mui/material/ImageList';
import ImageListItem from '@mui/material/ImageListItem';
import ImageListItemBar from '@mui/material/ImageListItemBar';
import IconButton from '@mui/material/IconButton';
import StarBorderIcon from '@mui/icons-material/StarBorder';

function srcset(image: string, width: number, height: number, rows = 1, cols = 1) {
  return {
    src: `${image}?w=${width * cols}&h=${height * rows}&fit=crop&auto=format`,
    srcSet: `${image}?w=${width * cols}&h=${height * rows}&fit=crop&auto=format&dpr=2 2x`,
  };
}

export default function CustomImageList() {
  return (
    <ImageList
      sx={{
        width: 500,
        height: 450,
        // Promote the list into its own layer in Chrome. This costs memory, but helps keeping high FP
        transform: 'translateZ(0)',
      }}
      rowHeight={200}
      gap={1}
    >
      {itemData.map((item) => {
        const cols = item.featured ? 2 : 1;
        return (
          <ImageListItem
            key={item.id}
            cols={cols}
            style={{ position: 'relative' }}>
            {item.image}
            <ImageListItemBar
              title={item.title}
              subtitle={item.description}
              actionIcon={item.featured ?  : null}
            />
          </ImageListItem>
        );
      })}
    
  );
}
```

```
const rows = item.featured ? 2 : 1;

return (
  <ImageListItem key={item.img} cols={cols} rows={rows}>
    <img
      {...srcset(item.img, 250, 200, rows, cols)}
      alt={item.title}
      loading="lazy"
    />
```

API



See the documentation below for a complete reference to all of the props and classes available to the components mentioned here.

- [`<ImageList />`](#)
- [`<ImageListItem />`](#)
- [`<ImageListItemBar />`](#)

Edit this page

Was this page helpful?

[Stack](#)

[Click-Away Listener](#) >