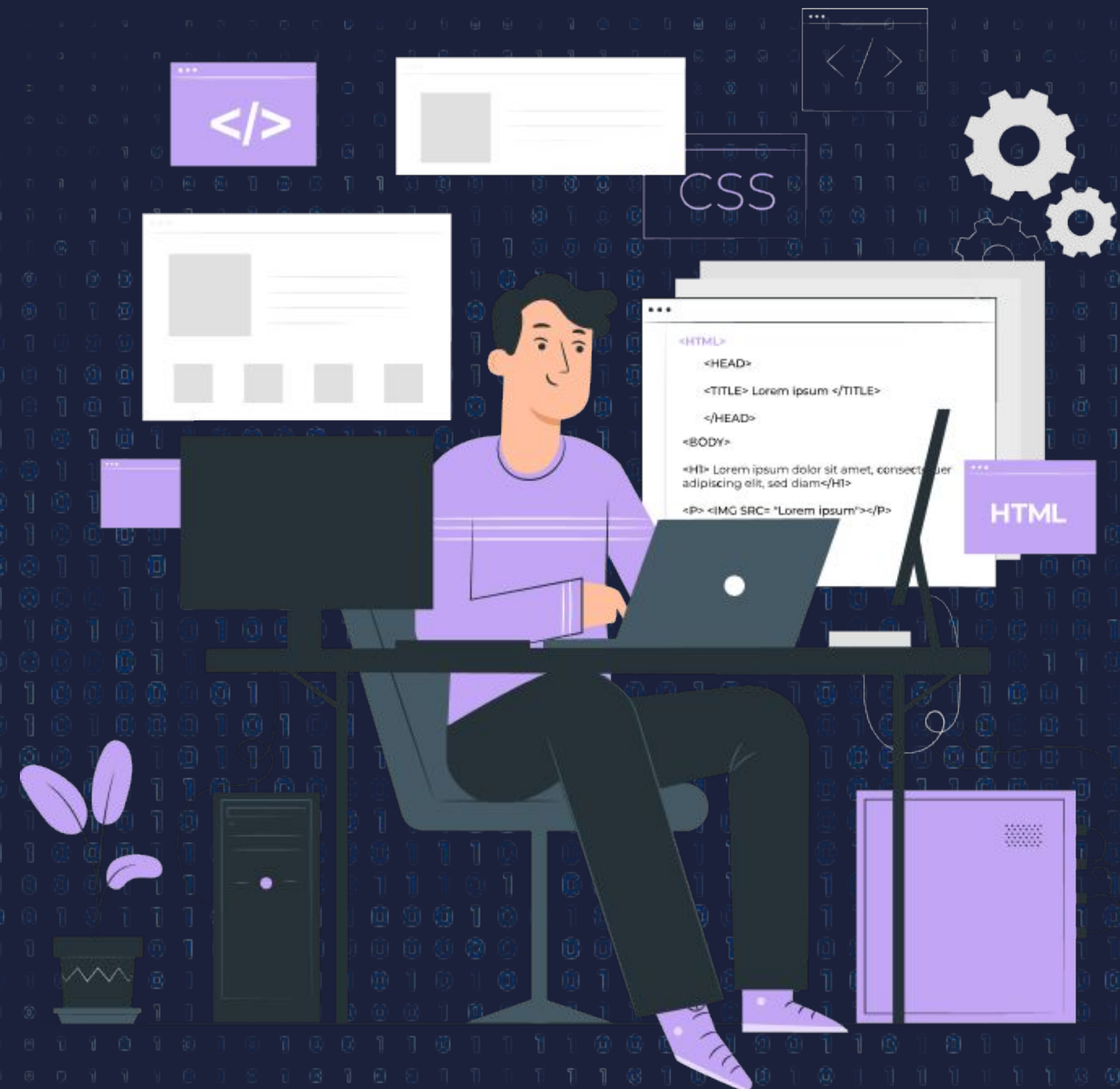




Css gradient





Topics

- Css gradient.
- Css linear gradient.
- Css radial gradient.



CSS Gradient

CSS gradients let you display smooth transitions between two or more specified colors



CSS Linear Gradient

It includes the smooth color transitions to going up, down, left, right, and diagonally.
The minimum two-color required to create a linear gradient.

Syntax:

```
background-image: linear-gradient(direction, color-stop1, color-stop2, ...);
```

Direction parameter: It can take values such as "to right", "to left", "to top", "to bottom", "to top left", "to top right", "to bottom left", or "to bottom right".

Color-stop parameters: specify the colors and positions of the gradient stops, You can specify as many color stops as you like, separated by commas.



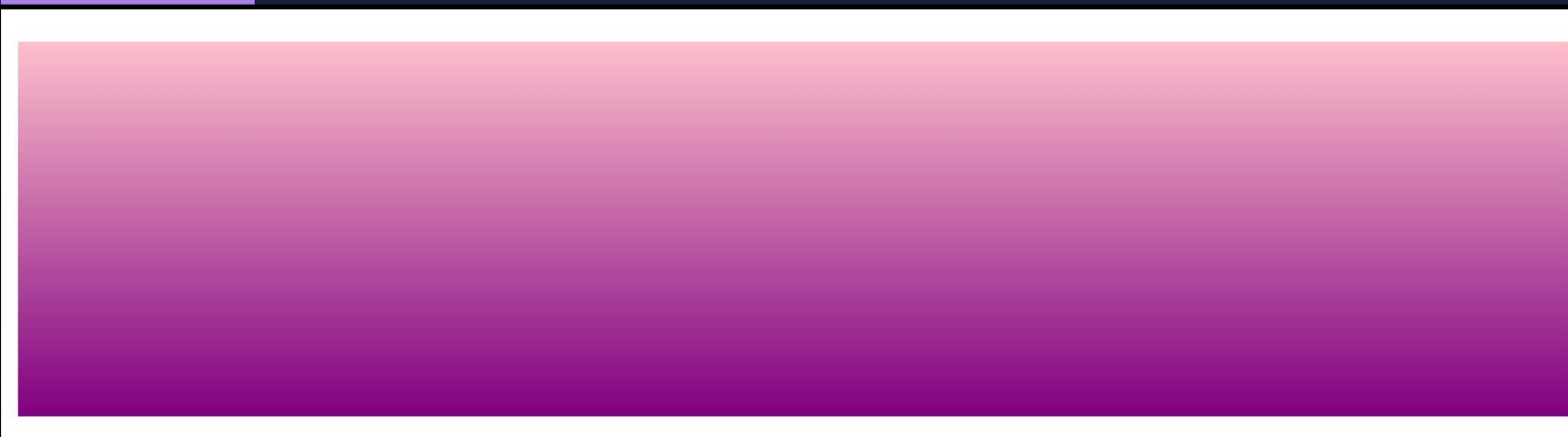
Examples

Top to bottom:

CSS

```
.box {  
  height: 50px;  
  background-image: linear-gradient(pink,  
  purple);  
  
  // background-image: linear-gradient(to bottom,  
  pink, purple);  
}
```

Output





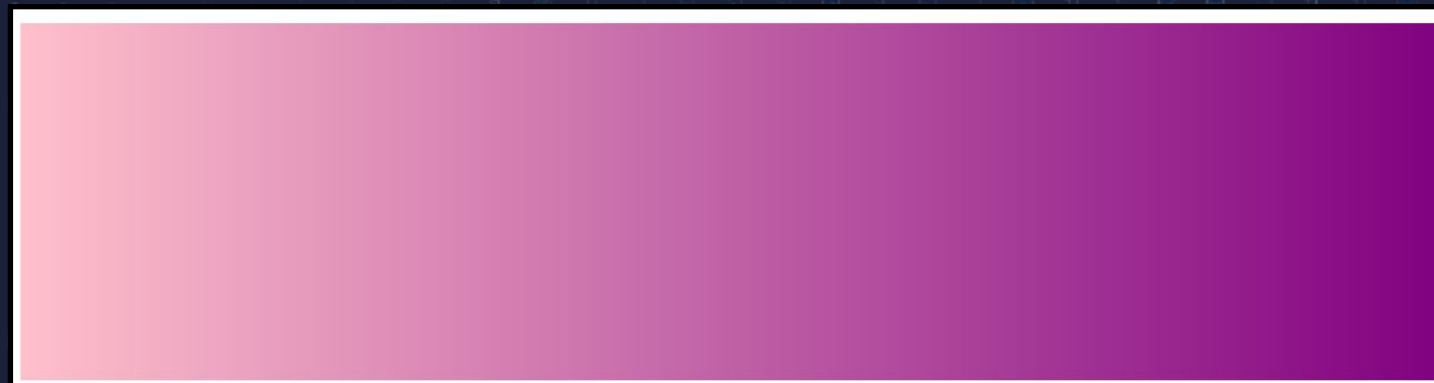
Examples

Left to right:

CSS

```
.box {  
  height: 50px;  
  background-image: linear-gradient(to right,  
pink, purple);  
}
```

Output





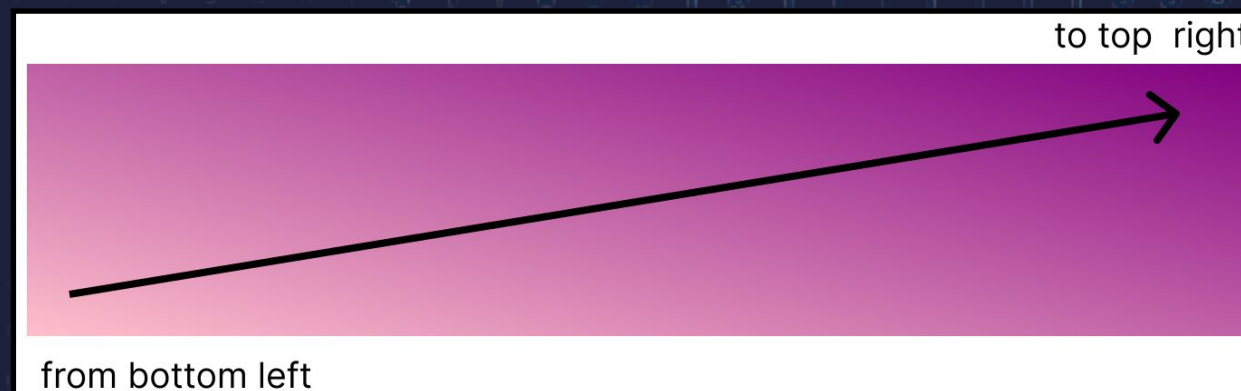
Examples

Diagonal : gradient that transitions smoothly between two or more colors diagonally instead of in a straight line. It can take the value such as “to top left”, “to bottom left”, “to bottom right”, “to top right”.

CSS

```
.box {  
  height: 50px;  
  /* from bottom left to top right */  
  background-image: linear-gradient(to top  
right, pink, violet, purple);  
}
```

Output





CSS radial gradient

Radial gradient color transition start from the center.

```
background-image: radial-gradient(shape size at position, start-color, ..., last-color);
```

shape size parameter : Specifies the shape and side of the gradient, It can take values such as "circle", "ellipse", "closest-side", "closest-corner", "farthest-side", or farthest-corner.

You can also specify the size of the shape by using a length value or a percentage value.



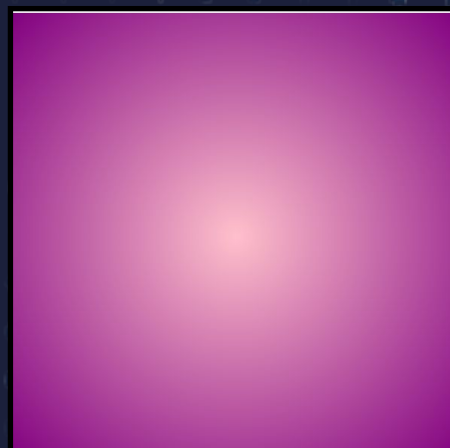
Examples

CSS radial-gradient (evenly spaced color stops) : The fade happens at an equal rate, by default, its shape is an eclipse.

CSS

```
.box {  
  height: 100px;  
  width: 100px;  
  background-image: radial-gradient(pink,  
  purple);  
}
```

Output





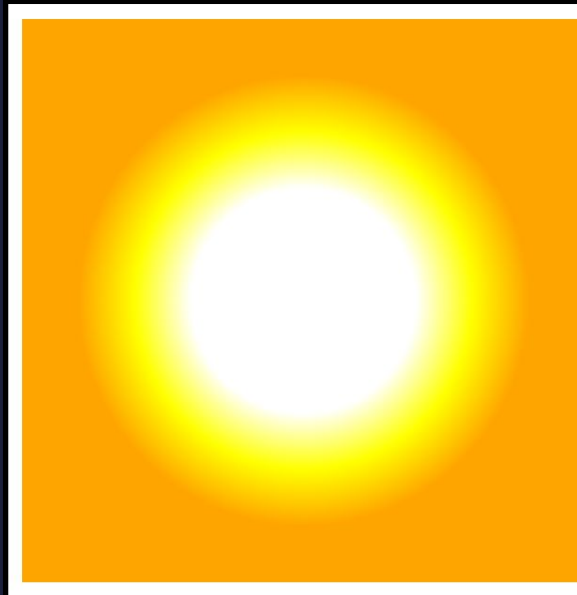
Examples

CSS radial-gradient (differently spaced color stops): Can adjust the spacing between color stops in a CSS radial gradient by specifying position values for each color stop.

CSS

```
.box {  
  height: 100px;  
  width: 100px;  
  
  background-image: radial-gradient(white 20px, yellow 30px,  
orange 40px);  
  
  /* background-image: radial-gradient(white 20%, yellow 30%,  
orange 40%); */  
}
```

Output





Set Shape

The shape parameter defines the shape. The default value is an ellipse. Here are the different values you can use for the "**shape size**" parameter:

- "circle"
- "ellipse"
- "closest-side"
- "closest-corner"
- "farthest-side"
- "farthest-corner"



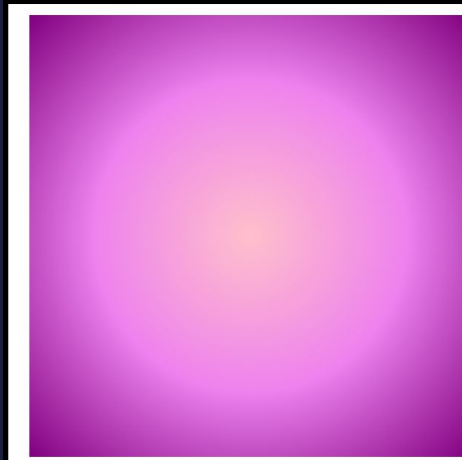
Radial gradient examples with shape parameter

Circle: This value creates a circular shape for the gradient

CSS

```
.box {  
  height: 100px;  
  width: 100px;  
  background-image: radial-gradient(circle, pink, violet,  
  purple);  
}
```

Output





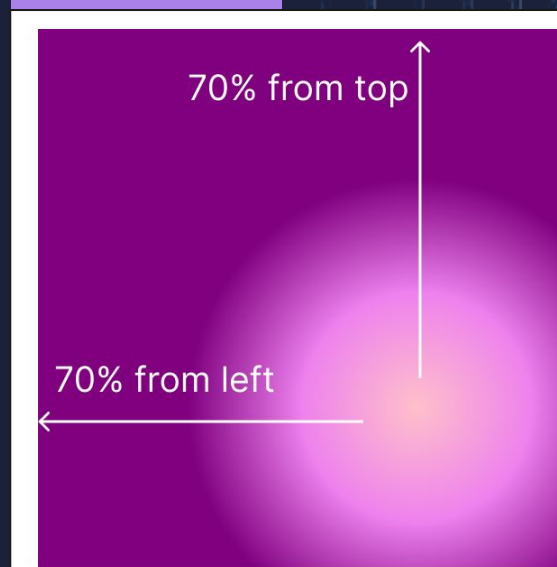
Radial gradient examples with shape parameter

closest-corner: This value creates a shape that is a circle or ellipse, depending on the aspect ratio of the element, with its size set to the smallest distance from the center of the gradient to the corners of the element.

CSS

```
.box {  
  height: 100px;  
  width: 100px;  
  background-image: radial-gradient(closest-corner at 70% 55%,  
pink, violet, purple);  
}
```

Output





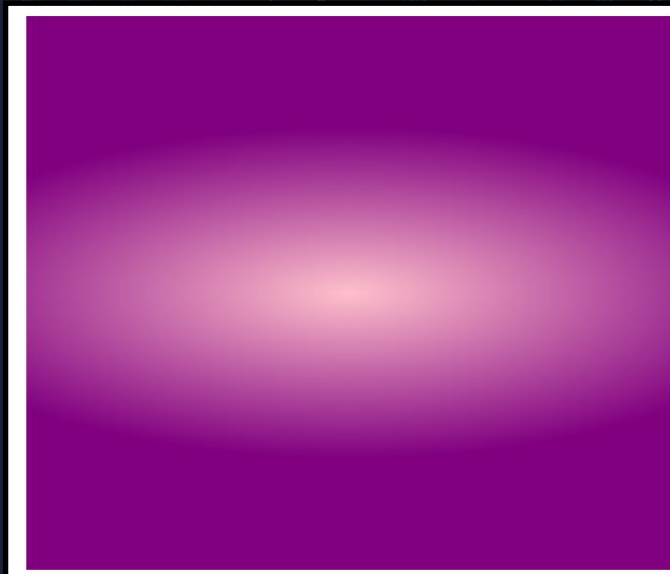
Radial gradient examples with shape parameter

Set the size of the shape by using a length or percentage value.

CSS

```
.box {  
  height: 100px;  
  width: 100px;  
  background-image: radial-gradient(70% 30%, pink, purple);  
/*  
  background-image: radial-gradient(70px 30px, pink, purple);  
*/  
}
```

Output





▶ THANK YOU ◀