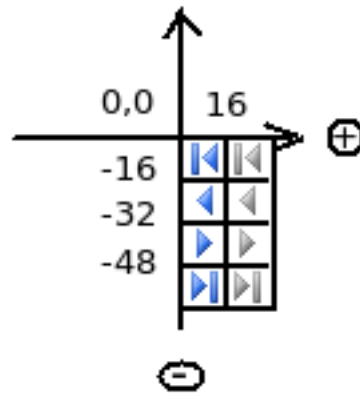


Trabalhando com sprites em CSS

Tenho a imagem, que quero utilizar mas apenas partes no meu código sem ter que cortar a imagem, mas mapeando com o CSS.



	= 0,0
	= 0, -16
	= 0, -48
	= 16, -16

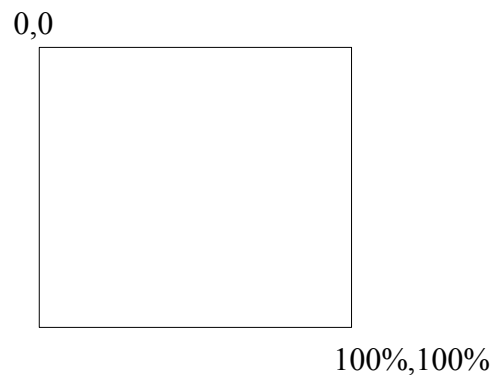


Sprite paging.gif

Lembrando que a propriedade background-position pode receber parâmetros de 3 tipos: posições, porcentagem e unidades.

As posições possíveis são: top left, top center, top right, center left, center center, center right, bottom left, bottom center e bottom right. Veja o gráfico abaixo:

LeftTop	CenterTop	RightTop
LeftCenter	CenterCenter	CenterRight
LeftBottom	CenterBottom	RightBottom



A posição **0% 0%** corresponde a **left top**. A posição **100% 100%** a **right bottom**.

O que pode dar trabalho de entender no início é que as três formas de trabalhar são inteiramente diferentes: com unidades refere-se ao plano cartesiano e ocupa o quarto quadrante (-Y e +X), com valores positivos e negativos.

Usando porcentagem varia de 0 a 100% e isso refere-se apenas ao tamanho da imagem.

Usando posições é simples mas restringe-se apenas às posições definidas, com apenas 9 pontos: (top left), (top center), (top right), (center left), (center center), (center right), (bottom left), (bottom center) e (bottom right).

Exemplo: background-position: left top;

Mas caso precisemos de mais posições, então devemos usar porcentagem ou unidades, como px.

Exemplo usando unidades (pixels):

```
<head>
<style>
.sprite, a.first_e, a.first_d, a.last_e, a.last_d, a.previous_e, a.previous_d, a.next_e, a.next_d
{
    background-Image:url('/paging.gif');
    display:block; /* Usado para quebrar linha line breaks are created with the block value. */
    width: 16px;
    height: 16px;
    /* Aqui, como não foi explicitada a propriedade background-position, é assumido o valor default: 0 0.
    */
}
/* Observe acima, que temos uma classe sprite e fazemos as demais classes herdarem dela várias
propriedades, nas classes abaixo apenas o que não existe na classe sprite */

/* Iniciar em 0,0 e pegar 16x16 */
a.first_e { background-position: 0px 0px; }

/* Iniciar em 16,0 e pegar 16x16 */
a.first_d { background-position: 16px 0px; }

/* Iniciar em 0,-16 e pegar 16x16 */
a.previous_e { background-position: 0px -16px; }

/* Iniciar em -16,-16 e pegar 16x16 */
a.previous_d { background-position: -16px -16px; }

/* Iniciar em 0,-32 e pegar 16x16 */
a.next_e { background-position: 0px -32px; }

/* Iniciar em -16,-32 e pegar 16x16 */
a.next_d { background-position: -16px -32px; }

/* Iniciar em 0,-48 e pegar 16x16 */
a.last_e { background-position: 0px -48px; }
```

```

/* Iniciar em -16,-48 e pegar 16x16 */
a.last_d { background-position: -16px -48px; }
</style>
</head>

```

```

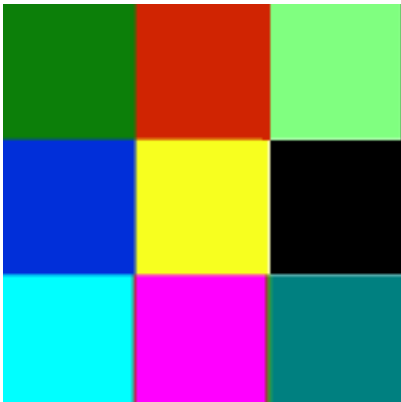
<body>
<!-- No caso sobrescrevemos a âncora-->
<a class="first_e"></a>
<a class="first_d"></a>
<a class="previous_e"></a>
<a class="previous_d"></a>
<a class="next_e"></a>
<a class="next_d"></a>
<a class="last_e"></a>
<a class="last_d"></a>

</body>

```

Exemplo usando as nove posições

Com esta imagem:



```

<head>
<style>
.sprite, a.lt, a.ct, a.rt, a.lc, a.cc, a.rc, a.lb, a.cb , a.rb
{
  background-Image:url('./posicoes.png');
  display:block;
  width: 50px;
  height: 50px;
  /* Aqui, como não foi explicitada a propriedade background-position, é assumido o valor default: 0 0.
  */
}
/* Observe acima, que temos uma classe sprite e fazemos as demais classes herdarem dela várias
propriedades, nas classes abaixo apenas o que não existe na classe sprite */

```

```
/* Usando posições só é adequado quando temos 9 posições: três acima, três ao centro e três embaixo
(). */
```

```
a.lt { background-position: top left; }
a.ct { background-position: top center; }
a.rt { background-position: top right; } /* 100% 0% */

a.lc { background-position: left center; }
a.cc { background-position: center; } /* mesmo que: center center */
a.rc { background-position: right center; }

a.lb { background-position: left bottom; }
a.cb { background-position: center bottom; }
a.rb { background-position: right bottom; }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<!-- No caso sobrescrevemos a âncora-->
```

```
<a class="lt"></a>
```

```
<a class="ct"></a>
```

```
<a class="rt"></a>
```

```
<a class="lc"></a>
```

```
<a class="cc"></a>
```

```
<a class="rc"></a>
```

```
<a class="lb"></a>
```

```
<a class="cb"></a>
```

```
<a class="rb"></a>
```

```
</body>
```

Exemplo usando porcentagem

Usando a mesma imagem anterior.

```
<head>
```

```
<style>
```

```
.sprite, a.lt, a.ct, a.rt, a.lc, a.cc, a.rc, a.lb, a.cb , a.rb
{
```

```
background-Image:url('./posicoes.png');
```

```
display:block;
```

```
width: 50px;
```

```
height: 50px;
```

```
/* Aqui, como não foi explicitada a propriedade background-position, é assumido o valor default: 0 0.
```

```

*/
}
/* Observe acima, que temos uma classe sprite e fazemos as demais classes herdarem dela várias
propriedades, nas classes abaixo apenas o que não existe na classe sprite */

/* Usando posições só é adequado quando temos 9 posições: três acima, três ao centro e três embaixo
().
background-position: { { percentage | length | left | center | right } 1 or 2 values | inherit } ;
lt = left top, ct = center top e rt = right top */

a.lt { background-position: 0% 0%; }
a.ct { background-position: 50% 0%; }
a.rt { background-position: 100% 0%; }

a.lc { background-position: 0% 50%; }
a.cc { background-position: 50% 50%; }
a.rc { background-position: 100% 50%; }

a.lb { background-position: 0% 100%; }
a.cb { background-position: 50% 100%; }
a.rb { background-position: 100% 100%; }

</style>
</head>

<body>
<!-- No caso sobrescrevemos a âncora-->
<a class="lt"></a>
<a class="ct"></a>
<a class="rt"></a>

<a class="lc"></a>
<a class="cc"></a>
<a class="rc"></a>

<a class="lb"></a>
<a class="cb"></a>
<a class="rb"></a>
</body>

```