

更进一步

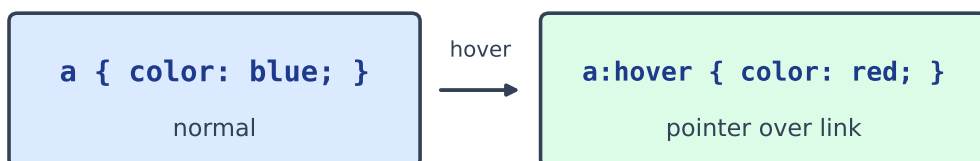
HTML & CSS Reference

:hover 与伪类

伪类 (pseudo-class) 给处于某种状态的元素设置样式。:hover 在鼠标悬停在元素上时生效。加上 transition(过渡), 变化就会平滑发生, 而不是突然跳变。

```
<style>
  .btn { background: #2563eb; color: white; padding: 10px 18px;
        border: none; border-radius: 8px; font-size: 16px;
        transition: background 0.3s; }
  .btn:hover { background: #16a34a; }
</style>
<button class="btn">Hover me</button>
```

- 其他常用伪类::focus(用户点进去的输入框) 和 :first-child。



Exam tip: :hover, :focus, :active are pseudo-classes — write selector:pseudo

:hover styles an element while the pointer is over it

position 定位

定位 (positioning) 让盒子离开原来的位置。relative 轻微移动盒子 (并让它成为子元素的锚点 (anchor)); absolute 从最近的已定位父元素量起, 把盒子精确摆放; fixed 把盒子固定在屏幕上, 页面滚动也不动。

```
<style>
  .card { position: relative; width: 260px; height: 90px;
        background: #dbeafe; border-radius: 10px; padding: 12px; }
```

```

.badge { position: absolute; top: -10px; right: -10px;
        background: #dc2626; color: white; padding: 4px 10px;
        border-radius: 999px; font-size: 13px; }
</style>
<div class="card">
  A product card
  <span class="badge">NEW</span>
</div>

```

- top、right、bottom、left 指定移动多远 —— 它们只对已定位的盒子生效。

媒体查询与响应式

响应式 (responsive) 页面会适应屏幕大小。媒体查询 (media query) 只在屏幕满足某个条件时才应用它的规则, 例如 max-width。

```

<style>
  .banner { background: #2563eb; color: white; padding: 16px; }
  @media (max-width: 600px) {
    .banner { background: #16a34a; }    /* 窄屏变绿 */
  }
</style>
<div class="banner">Resize the window: blue when wide, green when
↪ narrow.</div>

```

- 移动优先 (mobile-first) 的做法: 先写手机样式, 再用 @media (min-width: ...) 补充大屏样式。
- 百分比宽度和 max-width 让盒子能跟着屏幕缩小。

常见错误

- :hover 紧贴选择器, 中间没有空格: 是 .btn:hover, 不是 .btn :hover。
- position: absolute 从最近的**已定位**父元素量起 —— 给那个父元素加 position: relative。
- 媒体查询要包住完整的规则:@media (max-width: 600px) { .a { ... } }。