更进一步

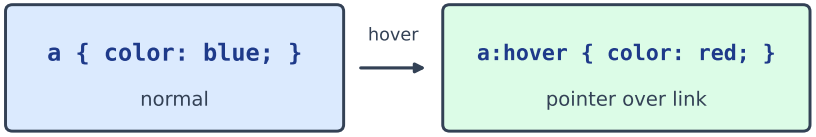
HTML & CSS Reference

Hover & pseudo-classes

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>
```

- 其他有用的伪类: `:focus` (获得焦点的元素, 例如按 Tab 键或点击后) 和`:first-child`。



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

:hover 在指针悬停时样式化元素

Positioning

定位定位将方框移出正常位置。`relative` 偏移方框 (并将其作为子元素的锚点锚点); `absolute` 从最近的定位祖先精确放置方框; `fixed` 通常在页面滚动时将其固定在视口上。变换过的祖先则建立其包含块。

```
<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。

Media queries & responsive design

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; }   /* narrow screens turn green */
  }
</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**定位**祖先测量 — 需指定预期的包含祖先 `position: relative`。若无包含祖先, 则使用初始包含块。
- 媒体查询要包住完整的规则:`@media (max-width: 600px) { .a { ... } }`。

exercises.lol exercises.com.cn