

Exercicios CSS

1) Cambia a cor de texto de todos os parágrafos a *azul*.

Solución:

```
p {  
    color: blue;  
}
```

2) Cambia a cor de texto dos parágrafos de clase “destacado” a *azul claro*.

Solución:

```
p.destacado {  
    color: lightblue;  
}
```

3) Selecciona o primeiro parágrafo e cambia a súa fonte a “*Arial, sans-serif*” e o seu tamaño a 18 píxeles.

Solución:

```
p:first-child {  
    font-family: Arial, sans-serif;  
    font-size: 18px;  
}
```

4) Fai que todos os encabezados de primeiro e segundo nivel teñan texto de cor *azul*.

Solución:

```
h1, h2 {  
    color: blue;  
}
```

5) Establece un borde *sólido* de 2 píxeles de cor *vermello* ás imaxes.

Solución:

```
img {  
    border: 2px solid red;  
}
```

O HTML podería ser (empregando picsum, unha web de imaxes):

```
  
>
```

6) Cambia a fonte de todos os elementos da páxina para que teñan un tamaño igual a **15 píxeles**.

Solución:

```
* { /* selector universal */  
  font-size: 15px;  
}
```

7) Selecciona os títulos **h1** e cambiar su cor de texto a **verde** e su tamaño a **24 píxeles**.

Solución:

```
h1 {  
  color: green;  
  font-size: 24px;  
}
```

8) Seleccionar todos os parágrafos e agregar un **espazado de liña** de 1.5.

Solución:

```
p { /* Esta propiedad non está nos apuntes */  
  line-height: 1.5em;  
}
```

9) Seleccionar todos os elementos de texto (parágrafos e título) e cambiar a súa familia de fontes a **"Helvetica, Arial, sans-serif"**.

Solución:

```
h1, p { /* non se nota moito o cambio */  
  font-family: Helvetica, Arial, sans-serif;  
}
```

10) Seleccionar o último parágrafo e darlle un fondo **gris** cun texto de cor **branco** e en **negriña**.

Solución:

```
p:last-child { /* explicar a diferencia con last-of-type */  
  background-color: gray;  
  color: white;  
  font-weight: bold;  
}
```

Podes ver unha explicación en <https://jsfiddle.net/guillermoml/h98zjron/>

11) Poñer todos os elementos do corpo da páxina con cor de letra azul, tamaño de letra 12 puntos e tipo de letra Arial.

Solución:

```
* { /* selector universal */  
  color: blue;  
  font-size: 12pt;  
  font-family: arial;  
}
```

Se estades tentados a empregar o selector `body` comprobade que non afectaría aos elementos `h1`, `h2`...

12) Os parágrafos deben ter unha cor de fondo verde, aliñado á esquerda, tamaño de fonte do 200% do texto por defecto.

Solución:

```
p {  
  background-color: green;  
  text-align: left; /*Por defecto xa están á esquerda */  
  font-size: 2em;  
}
```

13) Poñer as etiquetas `h1` con cor de letra vermello, tamaño de letra un 50% maior que por defecto do navegador e tipo de letra Verdana.

Solución:

```
h1 {  
  color: red;  
  font-size: 1.5em; /* Diminúe o tamaño */  
  font-family: verdana;  
}
```

Veredes que o tamaño de `h1` diminúe porque por defecto é 2em.

14) Nas etiquetas `h2` poñer: cor de letra verde, tamaño de letra un 25% maior que por defecto do navegador, tipo de letra *garamond* e estilo de fonte *cursiva* (itálica).

Solución:

```
h2 {  
  color: green;  
  font-size: 1.25em;  
  font-family: garamond;  
  font-style: italic;  
}
```

Veredes que o tamaño de `h2` diminúe porque por defecto é 1.5em.

15) Poñer as etiquetas **strong** cun ancho de caracteres *normal*, e poñer o texto en *maiúsculas*.

Solución:

```
strong {  
    font-weight: normal; /* Por defecto é negriña */  
    text-transform: uppercase;  
}
```

16) Poñer as etiquetas **em** cun estilo de fonte *normal*.

Solución:

```
em {  
    font-style: normal; /* Por defecto é cursiva */  
}
```

17) Poñer *subliñadas* as etiquetas **em**.

Solución:

```
em {  
    text-decoration-line: underline.  
}
```

18) Facer que as ligazóns non visitadas teñan cor de letra *púrpura* e as visitadas que sexa *fucsia*. Ao colocar o rato por encima: letra de cor *lima*, ao facer clic que se poña *amarela*.

Solución:

```
a:link { color: maroon; }  
a:visited { color: fuchsia; } /* Non funciona en modo  
incógnito por temas de privacidade */  
a:hover { color: lime; }  
a:active { color: yellow; }
```

19) Crea una nova clase **eliminado** con estas propiedades: decoración de texto: *tachado*, cor: *morada*. Comproba que funcione.

Solución:

```
.eliminado {  
    text-decoration-line: line-through;  
    color: purple;  
}
```

Nota: Para probar a nova clase `eliminado` podeses incluír no documento HTML o seguinte código: `palabra`

20) Fai desaparecer as dúas primeiras propiedades do código comentando o código correspondente.

```
p {  
    text-decoration-line: line-through;  
    color: purple;  
    text-align: center;  
    font-size: 40px;  
}
```

Solución:

```
p {  
    /*  
    text-decoration-line: line-through;  
    color: purple;  
    */  
    text-align: center;  
    font-size: 40px;  
}
```

21) Dado o seguinte código selecciona o terceiro div de xeito inequívoco e aplícalle unha propiedade calquera.

```
<div class="links" id="links1">Primeiro div</div>  
<div class="links" id="links2">Segundo div</div>  
<div class="links" id="links3">Terceiro div</div>
```

Solución:

Non podes escoller un selector por nome/elemento dado que son todos iguais. Tampouco podes por clase dado que todos teñen a mesma clase e aínda que non fose así se máis adiante se asignase a clase a outro elemento o selector deixaría de ser único.

Pero o id debe ser único nunha páxina (senón sería erróneo) e neste caso se cumpre co cal:

```
#links3 {  
    font-size: 40px;  
}
```

Poderíamos escoller mediante unha pseudoclase, caso de non haber máis elementos (se pon coma exemplo, non é unha pseudoclase que teñamos nos apuntes):

```
div:nth-child(3) {  
    font-size: 40px;  
}
```

Outra opción sería empregar o seguinte (tampouco a vimos na teoría):

```
div:nth-of-type(3) {  
    font-size: 40px;  
}
```

pero non sería tan inequívoco, dado que depende de que non se meta ningún div por medio.

22) Dado o seguinte código escolle o segundo e aplica calquera propiedade.

```
<div class="clase1" id="div1">Primeiro div</div>
<span class="sp1"> elemento 1</span>
<div class="links" id="links2">Segundo div</div>
<span class="sp2"> elemento 2</span>
<div class="links" id="links3">Terceiro div</div>
<span class="sp3"> elemento 3</span>
```

Solución:

Dados os atributos que temos vemos que debemos escoller un parágrafo con clase p2, entón:

```
/* span é o elemento, .sp2 é a clase*/
span.sp2 {
    font-size: 40px;
}
```

Poderíamos escoller mediante unha pseudoclase, caso de non haber máis elementos:

```
span:nth-child(4) {
    font-size: 40px;
}
```

23) Dado o código seguinte fai que nun só selector todos os div, span e parágrafos teñan un tamaño de fonte de 20px.

```
<div class="clase1" id="div1">Primeiro div</div>
<p class="p1">Parágrafo 1</p>
<div class="links" id="links2">Segundo div</div>
<span class="p2"> elemento 2</span>
<div class="links" id="links3">Terceiro div</div>
<span class="sp3"> elemento 3</span>
```

Solución:

Dúas posibles opcións:

```
/* Eiquí traballamos con elementos/nomes */
div, p, span {
    font-size: 20px;
}
```

ou

```
/* Eiquí traballamos con clases */
.clase1, .p1, .links, .p2, .sp3 {
    font-size: 20px;
}
```

Tede en conta que o resultado sobre o código amosado é o mesmo pero non son os mesmos selectores, se houbese máis elementos ambos anacos de código potencialmente seleccionarían distintas cousas dentro da páxina.

24) Dado o seguinte código, de que cor ser verá o parágrafo e o elemento span?.

```
<html>
<head>
  <style>
    #azul {color: blue;}
    .orange {color: orange;}
    .verde {color: green;}
    p {color: red;}
  </style>
</head>
<body>
  <p class="orange">
    Este parágrafo ten un
    <span id="azul" class="verde" style="color:grey;">
      elemento span, de que cor vai ser?
    </span>
    trata de averigualo antes de poñer o código.
  </p>
</body>
</html>
```

Solución:

Parágrafo:

- Vemos que o elemento `<p>` está afectado por:
 - selector de elemento `p {color: red;}`
 - selector de clase `.orange {color: orange;}`

O selector de clase domina sobre o máis xenérico de elemento, co cal será laranja.

Span:

- O elemento span está afectado:
 - Polo selector do parágrafo onde está incluído.
 - Polo selector de id `#azul {color: blue;}`
 - Polo selector de clase `.verde {color: green;}`
 - Polo estilo en liña `style="color:grey;"`
- Dado que:
 - estilo do pai ten menos especificidade ca estilo de clase
 - estilo de clase ten menos especificidade ca estilo de id
 - estilo de id ten menos especificidade ca estilo en liña

ao final ese anaco de texto estará en gris.

25) Dado o seguinte código, de que cor será o encabeçamento h1?

```
<html>
  <head>
    <style>
      h1.encabezamento1 {
        color: purple;
      }
      .encabezamento1 {
        color: green;
      }
      h1 {
        color: grey;
      }
    </style>
  </head>
  <body>
    <h1 class="encabezamento1">Encabezamento de nivel 1</h1>
  </body>
</html>
```

Solución:

Repasando a especificidade:

- O selector de elemento vale 1
 - `h1 { color: grey; }`
- O selector de clase vale 10
 - `.encabezamento1 { color:green; }`
- O selector de clase para o elemento h1 vale 10 + 1
 - `h1.encabezamento1 { color: purple; }`

Polo que o encabezamento será morado (purple).

26) Cal destas propiedades terá maior especificidade?.

```
h1 {color: red; } /* Vermello */  
h1 {color: rgb(255, 0, 0); } /* Vermello */  
h1 {color: #FF0000; } /* Vermello */
```

Solución:

Ningunha, son todas iguais, o que se está a poñer son distintas maneiras de expresar a cor vermella, pero non ten nada que ver coa especificidade, que é de valor 1 en todos os casos.

27) Dado o seguinte código, como poñerías un fondo púrpura e un texto de cor branca.

```
<div class="div01">  
    Neque porro quisquam est qui dolorem ipsum quia dolor sit  
    amet, consectetur, adipisci velit  
</div>
```

Solución:

Dado que o div ten unha clase o máis aconsellable é dar as propiedades á clase:

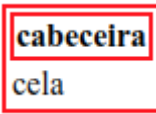
```
.div01 {  
    color: white;  
    background-color: purple;  
}
```

Como non se especifica tamén valería poñer o valor ao elemento:

```
div {  
    color: white;  
    background-color: purple;  
}
```

28) Como aplicas o borde á táboa e a cabeceira pero non ás celas? (como na imaxe).

```
<table>  
  <tr>  
    <th>cabeceira</th>  
  </tr>  
  <tr>  
    <td>cela</td>  
  </tr>  
</table>
```



Solución:

Aplicando a propiedade border especificamente a eses elementos en concreto:

```
table, th {  
    border: 2px solid red;  
}
```

Se quixeramos aplicalo a todo deberíamos engadir td á listaxe.

29) Temos un contedor div e aínda que o tamaño óptimo é de 640px queremos que o texto se adapte ao ancho do navegador, como debemos facer?

```
<div>
    Texto relativamente longo que imos supoñer que ocupa
    640px en situacións ideais.
</div>
```

Solución:

Se aplicamos directamente a anchura con `width` o problema é que se o tamaño do navegador é inferior a ese tamaño aparecerá unha barra de scroll.

Para estes casos o ideal é empregar `max-width` que cumpre co que pide o enunciado.

```
div {
    max-width: 640px;
}
```

30) Como podemos fazer para aplicar unha pequena sangría aos parágrafos?, algo deste estilo:

```
<h1>Texto 1</h1>
<p>
Texto a aplicar a sangría.
</p>
<h1>Texto 2</h1>
<p>
Texto a aplicar a sangría.
</p>
```

Texto 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Texto 2

Bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla,
bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla,
bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla,
bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla, bla,

Solución:

O que fazemos é aplicar un borde esquerdo ao parágrafo:

```
p {
    margin: 0 0 0 40px; /* Marxe no lado esquerdo */
}
```

31) Queremos que a táboa teña a cabeceira centrada e o resto de celas estean aliñadas á esquerda, que CSS debemos empregar?

Dato1	Dato2
valor	100,00

Solución:

Por defecto as táboas xa empregan esa aliñación. Se queremos facelo explícito:

```
th { text-align: center; }
td { text-align: left; }
```

32) Aplica un borde á táboa e a todas as súas celas. Asegúrate que o borde sexa único (non dobre).

Cabeceira	Cabeceira
Cela	Cela

Solución:

Tal cal está nos exemplos dos apuntes, hai que aplicar o borde á táboa, as cabeceiras e celas. A maiores hai que empregar a propiedade `border-collapse` para fundir as liñas.

```
table, td, th {  
    border: 1px solid;  
    border-collapse: collapse;  
}
```

33) Agora buscamos que a táboa teña o texto centrado tanto horizontal coma verticalmente.

Cabeceira	Cabeceira con moito texto vertical
Cela	Cela con moito texto que fai que á da esquerda deba centrarse.

Solución:

Empregamos

```
table { /* Aplicado a toda a táboa */  
    text-align: center;  
    vertical-align: middle;  
}
```

Tamén valería:

```
td { /* Aplicado a celas, as cabeceiras xa o están */  
    text-align: center;  
    vertical-align: middle; /* Por defecto xa está */  
}
```

34) Reproduce este texto empregando as propiedades correspondentes:

Este anaco de cor branco e fondo azul, *este en cursiva*, **este en negriña** e ~~este tachado~~.

Solución:

Empregamos:

```
<span style="color: white; background-color: blue;">Este anaco de cor branco e fondo azul</span>, <span style="font-style: italic;">este en cursiva</span>, <span style="font-weight: bold;">este en negriña</span> e <span style="text-decoration-line: line-through;">este tachado</span>.
```

Tede en conta:

- O que se pide é un texto continuo que vai cambiando de propiedades co cal temos que empregar `` dado que `<p>` provoca cambio de liña.
- Podemos empregar estilos inline ou darlle un id a cada elemento `` e aplicar as propiedades a cada id.

35) Fai que apareza en maiúsculas este texto:

```
<span class="maiusculas">texto en minúsculas</span>
```

Solución:

Empregamos

```
.maiusculas {  
    text-transform: uppercase;  
}
```

36) Facer que o texto sexa un 300% máis grande que o estándar:

```
<span class="agrandar">texto a agrandar</span>
```

Solución:

Empregamos

```
.agrandar {  
    font-size: 300%;  
}
```

ou

```
.agrandar {  
    font-size: 3em;  
}
```

37) Pon este texto cunha fonte de tamaño fixo:

```
<pre>
  <code class="codigofonte">
    #!/usr/bin/python3
    print("Hello, World!")
  </code>
</pre>
```

Solución:

Temos que empregar fontes de tipo monospace:

```
.codigofonte {
  font-family: "New Courier", Courier, Monospace;
}
```

Poñemos dende a tipografía máis específica á máis xeral.

Código HTML para os seguintes exercicios de selectores

```
<!DOCTYPE html>
<html>
  <head>
    <title>Exercicios de selectores</title>
    <meta charset="utf-8"/>
  </head>
  <body>
    <div class="numero1">
      <div class="numero2">
        <span>Texto1</span>
        <div class="numero2">
          <span class="span">Texto2</span> e
        <span>Texto3</span>
        </div>
        <div class="numero2">Texto3.1</div>
      </div>
      <div>Texto4</div>
      <div class="numero3">Texto5</div>
    </div>
    <div>
      <p class="numero2">Texto6 <span>Texto7</span></p>
    </div>
  </body>
</html>
```

38) Emprega selectores para seleccionar "Texto1"

Solución

```
.numero1 > .numero2 > span { /* Opción 1 */  
    font-size: 1.5em;  
}
```

Esta primeira opción funciona porque seleccionamos un elemento `span` que sexa fillo dun elemento coa clase `numero2` que á súa vez sexa fillo doutro elemento cunha clase `numero1`.

```
.numero1 > .numero2 > :first-child { /* Opción 2 */  
    color: blue;  
}
```

A segunda opción escolle o primeiro fillo dun elemento coa clase `numero2` que á súa vez sexa fillo doutro elemento cunha clase `numero1` e coincide que o elemento `span` é o primeiro elemento contido nese `div`.

39) Emprega selectores para seleccionar "Texto2" e "Texto3" pero non a conxunción "e".

Solución

```
.numero1 > .numero2 > .numero2 > span {  
    color: green;  
}
```

Aquí encadeamos 3 elementos coas súas respectivas clases, e despois collemos os fillos que sexan de tipo `span`, e neste caso hai 2 elementos. A conxunción "e" queda fora porque non está dentro dun `span`.

40) Emprega selectores para seleccionar a cadea "Texto2 e Texto3"

Solución

```
.numero1 > .numero2 > span + div {  
    background-color: yellow;  
}
```

Este exemplo é máis retorcido, dado que para seleccionar todo o que hai no `div` temos que dirixirnos ao `span` que contén "Texto1" e dicir que colla o elemento `div` inmediatamente posterior que estea á súa altura (irmán).

```
<div class="numero1">  
    <div class="numero2">  
        <span>Texto1</span>  
        <div class="numero2">  
            <span class="span">Texto2</span> e <span>Texto3</span>  
        </div>  
    </div>
```

Temos estas outras posibilidades:

```
.numero1 > .numero2 :nth-child(2)
```

collemos o segundo elemento dentro dese `div`.

Ou se quixeramos ser máis específicos:

```
.numero1 > .numero2 div:nth-child(2)
```

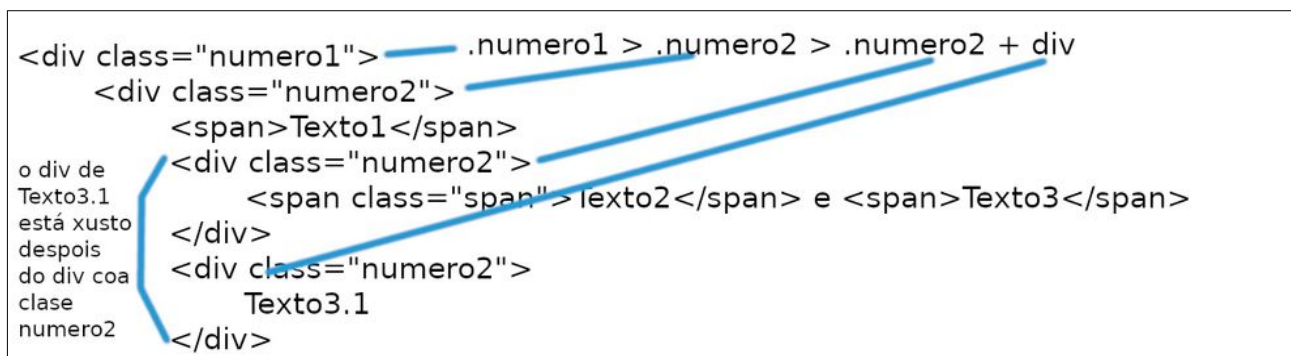
collemos o segundo elemento dentro dese `div` sempre que sexa un `div`.

41) Emprega selectores para seleccionar a cadea "Texto3.1"

Solución

```
.numero1 > .numero2 > .numero2 + div {  
    font-style: italic;  
}
```

Moi similar ao anterior, nos situamos no `div` que contén `Texto2` e `Texto3` e collemos o `div` inmediatamente posterior empregando `+ div`



Dado o código HTML que temos tamén sería posible esta solución:

```
.numero1 > .numero2 :last-child {  
    font-style: italic;  
}
```

Dado que collemos o derradeiro elemento dentro do `<div class="numero2">` que o contén.

Outra posibilidade:

```
.numero1 > .numero2 div:last-child
```

igual ca a anterior, soamente un pouco máis específico porque indicamos que o último elemento debe ser un `div`.

E tamén:

```
.numero1 > .numero2 :nth-child(3)
```

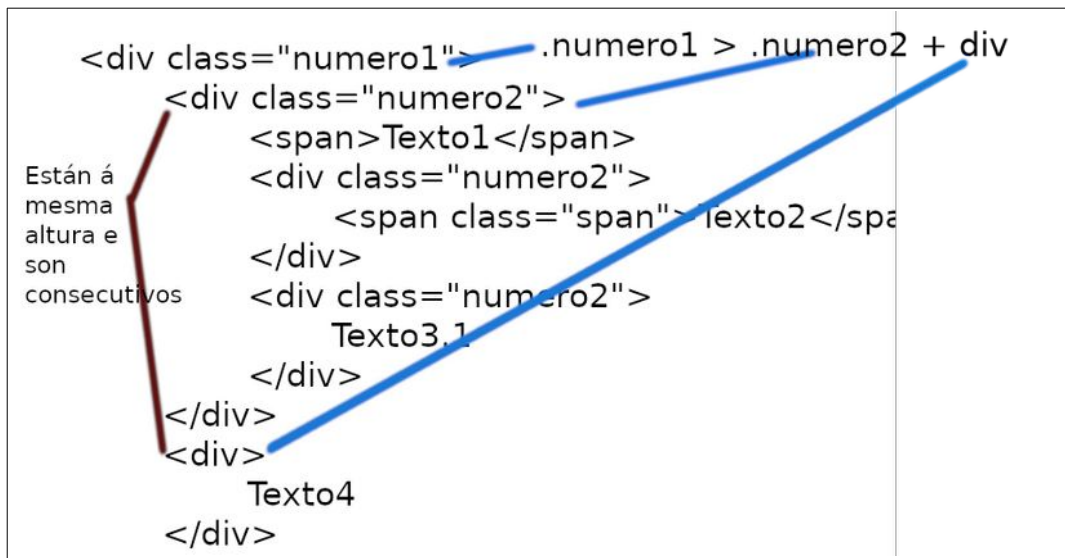
collemos o terceiro elemento dentro dese `div`.

42) Emprega selectores para seleccionar a cadea "Texto4"

Solución

```
.numero1 > .numero2 + div {  
    text-decoration-line: line-through;  
}
```

Texto4 está nun div (sen clases) que está dentro do div con clase `numero1`. Entón collemos o div máis externo con clase `numero2` e empregamos `+` para coller o div irmán inmediatamente posterior.



Outra posibilidade sería:

```
.numero1 > :nth-child(2)
```

o que estamos a indicar é que dun elemento con clase `numero1` collamos o segundo fillo.

43) Emprega selectores para seleccionar a cadea "Texto5"

Solución

```
.numero1 > .numero3 {  
    text-transform: uppercase;  
}
```

Aproveitamos que *Texto 5* está dentro dun div coa clase `numero3` para escoller de xeito sinxelo o elemento con esa clase dentro doutro con clase `numero1`.

Igual que en casos anteriores, tamén podemos empregar:

```
.numero1 > :last-child
```

e

```
.numero1 > :nth-child(3)
```



```

<div class="numero1">
  <div class="numero2">
    <span>Texto1</span>
    <div class="numero2">
      <span class="span">Texto2</span>
    </div>
    <div class="numero2">
      Texto3.1
    </div>
  </div>
  <div>
    Texto4
  </div>
  <div class="numero3">
    Texto5
  </div>
</div>

```

.numero1 > .numero3

44) Emprega selectores para seleccionar a cadea "Texto6 e Texto7"

Solución

```

div > p {
  background-color: lightblue;
}

```

Esa cadea de texto está dentro dun parágrafo que á súa vez está dentro dun div.

```

<div>
  <p class="numero2">
    Texto6 <span>Texto7</span>
  </p>
</div>

```

div > p

Tamén valería:

```
p.numero2
```

e por suposto, dado que non hai outro parágrafo en toda a páxina:

```
p
```

e outras posibilidades coma:

```
body > div:last-child
```

45) Emprega selectores para seleccionar a cadea "Texto7" exclusivamente

Solución

```
div > p > span {  
    font-size: 1.5em;  
}
```

igual que o caso anterior, pero ademais Texto7 está dentro dun `span` co cal temos que engadir o `span` que é fillo do parágrafo.

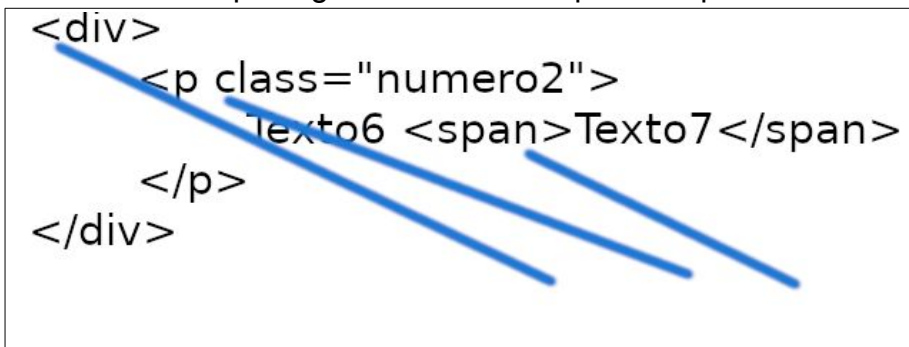
Dado que só hai un parágrafo poderíamos simplificalo a:

```
p > span
```

Outra opción sería:

```
div p span {  
    font-size: 1.5em;  
}
```

é dicir en vez de escoller os fillos escollemos os descendentes (fillos, netos...) e neste caso non hai coincidencias que fagan escoller outra parte da páxina.



```
<div>  
  <p class="numero2">  
    Texto6 <span>Texto7</span>  
  </p>  
</div>
```

O diagrama mostra a estrutura de HTML anterior. Trés liñas azuis diagonais indican a selección de descendentes: a primeira liña conecta o `<div>` co `<p>`, a segunda liña conecta o `<p>` co ``, e a terceira liña conecta o `` co `Texto7`.

As solucións dos anteriores selectores as podes ver en <https://jsfiddle.net/guillermoml/br0wve8d/>

46) Asegúrate mediante CSS de que todos os encabezamentos comecen por unha letra maiúscula

Solución

```
h1::first-letter, h2::first-letter, h3::first-letter,  
h4::first-letter, h5::first-letter, h6::first-letter {  
    text-transform: uppercase;  
}
```

Combinamos o pseudoelemento `::first-letter` cos distintos encabezamentos (h1-h6) para poñer a primeira letra de cada un deles en maiúscula

