

## Práctica 2 Entornos Gráficos - CSS

### Integrantes:

Barroso Bollero Agustin - 52818

Caracchi Victoria - 53482

Ponce Lautaro - 52898

Reschini Enrico - 52973

## Ejercitación 1:

1. ¿ Qué es CSS y para qué se usa?

CSS son las siglas de Cascade Style Sheet que traducido significa hojas de estilo en cascada. Las hojas de estilo son una tecnología que nos permite controlar la apariencia de una página web. CSS describe cómo los elementos dispuestos en la página son presentados al usuario permitiéndonos especificar estilos como el tamaño, fuentes, color, espaciado entre textos y recuadros así como el lugar donde disponer texto e imágenes en la página.

2. CSS utiliza reglas para las declaraciones de estilo, ¿cómo funcionan?

La regla tiene dos partes: un selector y la declaración. A su vez la declaración está compuesta por una propiedad y el valor que se le asigne. El selector funciona como enlace entre el documento y el estilo, especificando los elementos que se van a ver afectados por esa declaración. La declaración es la parte de la regla que establece cuál será el efecto.

Por ejemplo, si declaramos la siguiente regla:

```
p {color:yellow; }
```

p es el selector y color:yellow la declaración, de esta manera la regla dice que a todos los elementos p se le asignan las propiedades de la declaración.

3. ¿ Cuáles son las tres formas de dar estilo a un documento?

- Utilizando una hoja de estilo externa, la cual vinculamos al documento a través del elemento <link>, el cual va dentro de la sección <head> de nuestro html.
- Utilizando el elemento <style> en el interior del documento al que se quiere dar estilo (generalmente se sitúa en la sección <head>). De esta forma los estilos se reconocerán antes que la página se cargue por completo.
- Utilizando estilos directamente sobre aquellos elementos que lo permiten a través del atributo <style> dentro del <body>. De esta manera se pierden las ventajas de las hojas de estilo, ya que se mezcla el contenido con su presentación.

4. ¿ Cuáles son los distintos tipos de selectores más utilizados? Ejemplifique cada uno.

- Selector universal \* : sirve para apuntar a todos los elementos HTML del documento.  
Ejemplo:  

```
* { color:red; }
```
- Selector de clase: Se crea una clase css en la hoja de estilo y luego, mediante el atributo class en el html indicamos que un elemento pertenece a una clase, para que tome sus propiedades. Los elementos pueden tener más de una clase.  
Ejemplo:

En html: `<h1 class='titulo'>TITULO</h1>`

En css (las clases se declaran con un punto en el inicio):

```
.titulo {  
    color:red; }
```

- Selector id: se utiliza igual que el selector de clase, pero precedido por un # y a diferencia de las clases estos se utilizan en un único elemento.

Ejemplo:

En html: `<h2 id='titulo'>TITULO</h2>`

En css (las clases se declaran con un punto en el inicio):

```
#titulo {  
    margin:0px; }
```

- Selectores de etiqueta: también denominados selectores de tipo, son aquellos en los que se indica el nombre de la etiqueta HTML a la que se le asignará una propiedad CSS.

Ejemplo:

```
h1 { color:black; }
```

- Selectores de atributo: permiten apuntar a elementos HTML que contengan un determinado atributo. Se definen, como mínimo, definiendo entre corchetes el nombre del atributo al que queremos señalar, con la nomenclatura *[nombre-atributo]*.

Ejemplo:

```
[href]{  
    text-decoration: none;  
    color: orange;  
}
```

## 5. ¿Qué es una pseudo-clase? Cuáles son las más utilizadas aplicadas a vínculos?

Las pseudo-clases son palabras clave que añadimos a nuestros selectores CSS para especificar un estado sobre el elemento seleccionado.

Las más utilizadas aplicadas a vínculos son:

- `:link` esta refiere a un vínculo que no ha sido visitado, el cual aparecerá en azul.
- `:visited` esta refiere a un vínculo que ha sido visitado, el cual aparecerá en rojo.

## 6. ¿Qué es la herencia?

Para entender el concepto de herencia primero se debe entender el concepto de elemento padre e hijo. Un ejemplo de los mismos son los elementos que utilizamos para declarar listas donde, `<ol>` es el elemento padre de `<li>`, los cuales son en consecuencia sus elementos hijos.

En CSS herencia, significa que algunas propiedades CSS heredan por defecto los valores establecidos en el elemento padre, pero otras no.

Por ejemplo, si para un elemento se establece el color (color), cada elemento que se encuentre dentro de él también se mostrará de ese color, a menos que les se haya aplicado un color diferente.

## 7. ¿En qué consiste el proceso denominado cascada?

En un primer nivel de simplicidad, la cascada en las hojas de estilo significa que el orden de las reglas importa en CSS: cuando dos reglas tienen la misma especificidad, se aplica la que aparece en último lugar en el CSS.

Es decir si por ejemplo para un mismo elemento, primero reclamamos una regla que indique que va en color verde y luego una que indique que va en amarillo, el elemento tomará el color amarillo ya que es el último que se declaró.

## Ejercitación 2:

```
p#normal {
    font-family: arial,helvetica;
    font-size: 11px;
    font-weight: bold;
}
*#destacado {
    border-style: solid;
    border-color: blue;
    border-width: 2px;
}
#distinto {
    background-color: #9EC7EB;
    color: red;
}
<p id="normal">Este es un párrafo</p>

<p id="destacado">Este es otro párrafo</p>

<table id="destacado"><tr><td>Esta es una tabla</td></tr></table>

<p id="distinto">Este es el último párrafo</p>
```

Este código primeramente muestra una regla cuyo selector es “p#normal”, indicando los estilos descritos a continuación se aplicarán solo a los elementos <p> (párrafo) que tengan el atributo id= “normal”. Luego, a través de los 3 renglones respectivos que se encuentran entre las llaves, dichas declaraciones definen la fuente de texto (arial o en su defecto helvetica), el tamaño del texto (11 píxeles) y se aplica un efecto “negrita” en el mismo.

En la regla número 2, observamos el selector “\*#destacado”, es decir, las declaraciones posteriores se aplicarán a cualquier elemento con el atributo id=“destacado”. Luego, se establece un borde sólido alrededor del elemento, se escoge el color azul para el borde y se fija el grosor del borde en 2 píxeles. ( a través de border-style, border-color y border-width respectivamente).

En última regla vemos que su selector se llama “#distinto”, lo que selecciona a cualquier elemento con el atributo id=“distinto”. El efecto que produce esta regla se

relaciona con el cambio de color de fondo (especificado con el código #9EC7EB) y cambio de color del texto (color rojo).

Por último y una vez hablado sobre las reglas previamente definidas, podemos decir que:

`<p id="normal">Este es un párrafo</p>`. Este párrafo tendrá el estilo definido en la regla “`p#normal`”, es decir, texto Arial o Helvetica, tamaño 11px y en negrita.

`<p id="destacado">Este es otro párrafo</p>`. Este párrafo aplicará la regla “`*#destacado`”, por lo que tendrá un borde azul sólido de 2px (con el estilo de texto por defecto del navegador).

`<table id="destacado"><tr><td>Esta es una tabla</td></tr></table>`. Esta tabla también tendrá un borde azul de 2 píxeles.

`<p id="distinto">Este es el último párrafo</p>`. Este párrafo tendrá el fondo de color #9EC7EB y el texto en rojo, como se define en la regla “`#distinto`”.

### Ejercitación 3:

```
p.quitar {
    color: red;
}
*.desarrollo {
    font-size: 8px;
}
.importante {
    font-size: 20px;
}
<p class="desarrollo">
    En este primer párrafo trataremos lo siguiente:
    <br />xxxxxxxxxxxxxxxxxxxxxxxxxxxx
</p>

<p class="quitar">
    Este párrafo debe ser quitado de la obra...
    <br />xxxxxxxxxxxxxxxxxxxxxxxxxxxx
</p>

<p >
    En este otro párrafo trataremos otro tema:<br />
    xxxxxxxxxxxxxxxxxxxxxxxxxxxx
</p>

<p class="importante">
    Y este es el párrafo más importante de la obra...
    <br />xxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
</ p>
```

```
<h1 class="quitar">Este encabezado también debe ser quitado de la obra</h1>
```

```
<p class="quitar importante">Se pueden aplicar varias clases a la vez</p>
```

Primeramente observamos la regla cuyo selector es “p.quitar”. Esto significa que el estilo es para los elementos <p> (párrafos) que tienen la clase “quitar”. El mismo tiene un renglón dentro de lo que es su declaración que establece que el color del texto será rojo.

Luego tenemos la segunda regla con el selector “\*.desarrollo”, lo que quiere decir que sus declaraciones se aplicarán a todos aquellos elementos que tengan la clase “desarrollo”. El renglón `font-size: 8px;` define el tamaño de fuente en 8 píxeles.

La tercera regla tiene el selector “.importante”. En este caso se las declaraciones que se definan se aplicarán a todos los elementos que tengan la clase “importante”. De esta forma, nos damos cuenta que el \* (utilizado en la regla anterior) es opcional, ya que si sólo usamos `.nombredeclase`, ya nos estamos refiriendo a “todos aquellos elementos que..”. Luego `font-size: 20px;` define el tamaño de fuente en 20 píxeles.

Por último y una vez hablado sobre las reglas previamente definidas, podemos decir que:

```
<p class="desarrollo">
  En este primer párrafo trataremos lo siguiente:
  <br />xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
</p>
```

Este párrafo tiene la clase “desarrollo”, por lo que el tamaño del texto será de 8 píxeles debido a la regla 2.

```
<p class="quitar">
  Este párrafo debe ser quitado de la obra...
  <br />xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
</p>
```

Este párrafo tiene la clase quitar, por lo que el color del texto será rojo, según la regla p.quitar.

```
<p>
  En este otro párrafo trataremos otro tema:
  <br /> xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
</p>
```

Este párrafo no tiene ninguna clase asignada, por lo que no se le aplica ninguna regla de estilo y seguirá el estilo por defecto del navegador.

```
<p class="importante">
  Y este es el párrafo más importante de la obra...
```

```
<br />xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
</ p>
```

Este párrafo tiene la clase importante, por lo que el texto será más grande (tamaño 20px).

```
<h1 class="quitar">
    Este encabezado también debe ser quitado de la obra
</h1>
```

Este encabezado tiene la clase “quitar”. Sin embargo la regla “p.quitar” solo afecta a los párrafos y no hay una regla específica para encabezados, por lo que el encabezado no aplicará los estilos de color rojo.

```
<p class="quitar importante">
    Se pueden aplicar varias clases a la vez
</p>
```

Este párrafo tiene dos clases: “quitar” e “importante”. De esta forma, se aplicarán los estilos de ambas: el color del texto será rojo y el tamaño del texto será de 20 píxeles.

## Ejercitación 4:

```
<body>
<p class="contenido"
style="font-weight:
normal;">
Este es un texto
.....</p>
<table>
<tr>
<td>Y esta es una
tabla.....</td>
</tr>
<tr>
<td><a
href="http://www.google.com.
ar">con un
enlace</a></td>
</tr>
</table>
</body>
```

```
<body class="contenido">
<p> Este es un
texto.....</p>
<table>
<tr>
<td>Y esta es una
tabla.....</td>
</tr>
<tr>
<td><a
href="http://www.google.com.
ar">con
un enlace</a></td>
</tr>
</table>
</body>
```

En el primer ejemplo, se le atribuye la clase contenido únicamente al párrafo escrito dentro de la primer etiqueta <p> por lo tanto el párrafo tomará las propiedades de esa clase y además mediante el atributo style en la declaración también se tomarán las propiedades allí agregadas, mientras que en el segundo ejemplo todo el cuerpo de la página web tomará las propiedades de la clase contenido, ya que el atributo está puesto dentro de la etiqueta <body>.

## Ejercitación 5:

En cada caso, declarar una regla CSS que produzca el siguiente efecto:

1. Los textos enfatizados dentro de cualquier título deben ser rojos.

```
h1 em, h2 em, h3 em, h4 em, h5 em, h6 em {  
    color: red;}
```

2. Cualquier elemento que tenga asignado el atributo "href" y que esté dentro de cualquier párrafo que a su vez esté dentro de un bloque debe ser color negro.

```
div p a[href] {  
    color: black; }
```

3. El texto de las listas no ordenadas que estén dentro del bloque identificado como "ultimo" debe ser amarillo pero si es un enlace a otra página debe ser azul.

```
#ultimo ul{  
    color:yellow;}  
  
#ultimo ul a[href]{  
    color:blue;}
```

4. Los elementos identificados como "importante" dentro de cualquier bloque deben ser verdes, pero sí están dentro de un título deben ser rojos.

```
div #importante{
```

```
color:green;}
```

```
h1 #importante , h2 #importante , h3 #importante , h4  
#importante , h5 #importante , h6 #importante {  
color:red;}
```

5. Todos los elementos h1 que especifique el atributo title, cualquiera que sea su valor, deben ser azules.

```
h1[title]{  
color:blue;}
```

6. El color de los enlaces en las listas ordenadas debe ser azul para los enlaces aún no visitados, y violeta para los ya visitados y, además, no deben aparecer subrayados.

```
ol a{  
text-decoration: none;}
```

```
ol a:link {  
color: blue; }
```

```
ol a:visited {  
color: violet; }
```

## Ejercitación 6:

Para obtener la misma salida en el navegador, deberíamos hacer 2 cosas. Primero, cambiar la etiqueta id=título por id=encabezado y además, al elemento de id=pie, asignarle la clase “estilopie”.



## Ejercitación 7:

Choose a level

✓ 1 A

✓ 2 A

✓ 3 #id

✓ 4 A B

✓ 5 #id A

✓ 6 .classname

✓ 7 A.className

✓ 8 Put your back into it!

✓ 9 A, B

✓ 10 \*

✓ 11 A \*

✓ 12 A + B

✓ 13 A ~ B

✓ 14 A > B

✓ 15 :first-child

✓ 16 :only-child

✓ 17 :last-child

✓ 18 :nth-child(A)

✓ 19 :nth-last-child(A)

✓ 20 :first-of-type

✓ 21 :nth-of-type(A)

✓ 22 :nth-of-type(An+B)

✓ 23 :only-of-type

✓ 24 :last-of-type

✓ 25 :empty

✓ 26 :not(X)

✓ 27 [attribute]

✓ 28 A[attribute]

✓ 29 [attribute="value"]

✓ 30 [attribute^="value"]

✓ 31 [attribute\$="value"]

✓ 32 [attribute\*="value"]