

### **Problema 1:**

Crear una página HTML y definir tres títulos empleando los elementos h1, h2 y h3. Definir el mismo color para el texto de cada bloque y colores diferentes para el fondo de los mismos (cada elemento HTML debe tener definida la propiedad style).

### **Problema 2:**

Definir un estilo diferente para los elementos h1, h2, h3, h4, h5 y h6. Mostrar mensajes utilizando estos elementos a las que se le han definido estilos (por ahora solo conocemos el color de texto y fondo).

### **Problema 3:**

Definir reglas para los elementos HTML: h1, h2, h3, h4, h5 y h6. Inicializar la propiedad font-size con valores decrecientes para cada uno de los elementos (40,30,25,20,15 y 10 píxeles). Inicializar la propiedad font-family para los tres primeros elementos con los valores: Arial, Arial Black y Arial Narrow.

### **Problema 4:**

Definir la misma fuente, color y tamaño de fuente para los elementos p (párrafo) y h6 (tener en cuenta que cuando vea la página el texto que se encuentra en el elemento h6 difiere del texto que se encuentra dentro del párrafo por la propiedad font-weight (ya que el elemento h6 es de tipo bold por defecto y el elemento p tiene por defecto normal)).

### **Problema 5:**

Definir para el elemento <h6> el mismo tamaño de fuente (12px) que el elemento <p> (párrafo) pero color azul para el título y gris claro (color:#aaaaaa) para el párrafo. Agrupar las dos elementos para la definición de la fuente y posteriormente agregar a cada elemento el valor respectivo para el color del texto.

### **Problema 6:**

Definir para el elemento párrafo (p) el color verde puro, texto centrado. Imprimir varios párrafos para ver los resultados según la regla de estilo definida. ¿Qué valor debemos definir para que el texto aparezca con color rojo oscuro?

### **Problema 7:**

Definir los elementos de cabecera h1, h2 y h3 con valores decrecientes para las propiedades:

- letter-spacing
- word-spacing

Luego inicializar la propiedad text-transform para que el texto siempre salga en mayúsculas.

### Problema 8:

Confeccione una página que defina una regla para el elemento body e inicialice las propiedades color y font-family. Luego defina reglas de estilo para los elementos h1, h2 y h3 que redefinan la fuente con los valores: Times New Roman, Courier y Arial respectivamente. Imprima tres títulos, cada uno con los elementos h1, h2 y h3. Por último, imprime un párrafo.

### Problema 9:

Definir distintas escalas de grises para los textos contenidos por el elemento "em" que se encuentren contenidos por los elementos h1, h2, h3, h4, h5 y h6.

La escala de grises que podemos utilizar es:

```
color:#dddddd; si está contenido en un h1
color:#cccccc; si está contenido en un h2
color:#bbbbbb; etc.
color:#aaaaaa;
color:#999999;
color:#888888;
```

Aparecen los bloques de más claro a más oscuro (cuando los tres valores son iguales (rojo, verde, azul) estamos en presencia de un gris (salvo el #000000 y el #ffffff)).

### Problema 10:

Crear una página web que contenga una cabecera de nivel 1 (h1), luego una cabecera de nivel 2 (h2) y un conjunto de párrafos. Definir reglas de estilo para los tres elementos h1, h2 y p.

### Problema 11:

En la página html definir estas dos clases:

```
.subrayado {
  color:#00aa00;
  text-decoration:underline;
}
.tachado {
  color:#00aa00;
  text-decoration:line-through;
}
```

Luego definir un título con nivel h1 (subrayar todo el título). Seguidamente un párrafo que tenga algunas palabras subrayadas y otras tachadas (utilizar el elemento span para asignar estilos a una palabra).

### Problema 12:

Crear una página HTML en la que definiremos estas dos reglas

```
.pregunta {  
    font-family:verdana;  
    font-size:14px;  
    font-style:italic;  
    color:#0000aa;  
}  
.respuesta {  
    font-family:verdana;  
    font-size:12px;  
    font-style:normal;  
    text-align:justify;  
    color:#555555;  
}
```

La página debe contener 3 preguntas y 3 respuestas. Cada pregunta y respuesta habrá que disponerla en un párrafo distinto. Asignar los estilos .pregunta y .respuesta

### Problema 13:

Definir tres estilos de tipo id (cabecera, cuerpo y pie), luego definir en el archivo HTML tres áreas (div) inicializando el atributo id con los nombres de reglas de estilo creados.

### Problema 14:

Inicializar los elementos HTML h1, h2 y h3 con bordes de 2 pixeles, color azul y diferentes estilos para cada uno. Crear una página HTML que los utilice. Por ejemplo:

```
h1 {  
    border-width:2px;  
    border-style:solid;  
    border-color:#0000ff;  
}
```

### Problema 15:

Crear una clase .titulosecundario que inicialice el borde superior e inferior con ancho de 2 pixeles y los bordes laterales con cero pixeles. Queda a tu elección el estilo y color.

### Problema 16:

Definir una clase titulo que fije los padding de izquierda y derecha con 20 pixeles y el superior e inferior en 10. Escribir un texto dentro del elemento h1.

### Problema 17:

Disponer títulos de primer y segundo nivel con un margen de 5 píxeles en la parte superior e inferior del elemento HTML.

### Problema 18:

Confeccionar dos listas de lenguajes de programación. Agrupar los lenguajes estructurados (Pascal, C) y los lenguajes orientados a objetos (C++, Java, C#) Aplicar estilos distintos a cada lista.

### Problema 19:

Crear dos listas y e inicializar la propiedad **list-style-position** con los valores inside y outside respectivamente y analizar en que difieren.

### Problema 20:

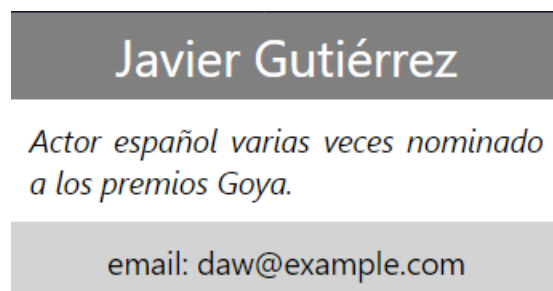
Defina cuatro propiedades para el selector universal y luego implemente una página HTML que pruebe su funcionamiento.

### Problema 21:

Definir la propiedad **font-size** con un valor distinto cuando la flecha del mouse está sobre el link y pintar al mismo tiempo el fondo de color amarillo.

### Problema 22:

Una **“.card”** (tarjeta) es una construcción casi estándar por su uso. Construir la tarjeta de la imagen.



La página, de título Tarjeta, usa como fuente [Arial, sans-serif] y el tamaño por defecto será 14px (ponlo como propiedades de html). Como ves la “card” tiene tres partes cuyas class son: titulo, body, foot. Las características son:

**título:** blanco sobre fondo gris, con un padding de 8 píxeles y centrado. La fuente es 1,5 veces el tamaño por defecto (unidades rem)

**body:** itálica, texto justificado, y paddign de 8px

**pie:** texto negro sobre fondo gris claro, centrado y con padding de 8px.

### Problema 23:

Hacer un menú de hipervínculos que se muestren en forma vertical.

**Problema 24:**

Modificar la hoja de estilo del menú desarrollado en el problema anterior para que cada opción aparezca una línea de color negro que la recuadre.

**Problema 25:**

Implementar un menú horizontal con la lista de los problemas anteriores.

**Problema 26:**

Confeccionar un menú horizontal y un menú vertical a la izquierda con la misma lista de los problemas anteriores.

**Problema 27:**

Crear una página HTML que muestre dos párrafos, a uno de ellos aplicarle el siguiente estilo:

```
.parrafoestrecho {  
  width:200px;  
  text-align:justify;  
}
```

Recordar que para aplicar esta clase a un párrafo debemos inicializar la propiedad class del elemento p (ej. <p class="margenestrecho">bla bla</p>).

**Problema 28:**

Crea un div y dispón una serie de párrafos. Define como padding 0.5rem. Luego prueba distintos tamaños de fuente para el elemento raíz "html" y ve como varía el padding del div.

**Problema 29:**

Utilizando la sintaxis simplificada para indicar las proporciones de rojo, verde y azul. pintar el fondo de 16 párrafos. Comenzar por el valor #000 hasta el #fff.

Hacer que los párrafos estén sin margen entre sí. También variar el color de la fuente para que permita su lectura.

**Problema 30:**

Disponer una lista ordenada con los nombres de los valores que puede adoptar la propiedad cursor. Luego definir dicho cursor para cada valor de propiedad.

Recordemos que la propiedad cursor puede adoptar los siguientes valores:

crosshair, default, pointer, move, e-resize, ne-resize, nw-resize, n-resize, se-resize, sw-resize, s-resize, w-resize, text, wait, help y auto.

**Problema 31:**

Crear un formulario que solicite el ingreso del nombre de una persona y su edad.

Aplicar el estilo 'campo' a los dos controles de tipo text.

```
.campo {  
  color:#666;  
  background-color:#ffa;  
}
```

### **Problema 32:**

Definir una tabla de varias filas y columnas. Aplicar estilos a los elementos th y td solamente las contenidas en el tbody.

### **Problema 33:**

Disponer dos párrafos, desplazar el segundo párrafo hasta que quede casi superpuesto al primer párrafo.

### **Problema 34:**

Disponer dos div con coordenadas absolutas, uno que represente la cabecera de la página (50 píxeles de alto) y otro la primera columna de la izquierda.

### **Problema 35:**

Mostrar 3 div con posiciones absolutas. Mostrarlos uno arriba de otro. Hacerlos de distintos tamaños. Mostrar más arriba el más pequeño y así sucesivamente.

### **Problema 36:**

Crear una página que disponga dos div con posicionamiento fijo. Uno a la derecha y otro en la parte inferior de la página.

```
#barrainferior{  
  position:fixed;  
  bottom:0px;  
  width:100%;  
  height:50px;  
  background-color:black;  
  color:white;  
}
```

```
#barralateralderecha{  
  position:fixed;  
  right:0px;  
  top:0px;  
  width:200px;  
  height:100%;  
  background-color:#eee;  
}
```

**Problema 37:**

Crear una página con dos columnas y disponer en la primera columna un menú de opciones (como el visto en conceptos anteriores)

**Problema 38:**

Inicializar la siguiente regla:

```
.letragrande {  
    float:left;  
    font-size:300%;  
}
```

Confeccionar una página en la que la primera letra de cada párrafo aparezca con el triple de tamaño y además el texto envuelva a la letra.

**Problema 39:**

Crear una página que tenga una cabecera, luego 2 columnas y finalmente un pie de página.

**Problema 40:**

Modifica el problema visto anteriormente para mostrar dos columnas, cabecera y pie de página. Luego asigna a la propiedad width, de la regla contenedor, el valor:

```
width:780px;  
margin: 0 auto;
```

¿Cuál es el resultado?

**Problema 41:**

Crear una página con tres columnas, cabecera y pie de página.

**Problema 42:**

Modifica nuevamente la sintaxis para disponer tres columnas, cabecera y pie de página. Luego define la propiedad width del contenedor con 780px.

**Problema 43:**

Crear una lista ordenada "ol" con tres item. En uno de esos tres item definir en forma anidada otra lista pero no ordenada "ul". Crear dos reglas de estilo distintas para los item que descienden de la lista ordenada y los que descienden de la lista no ordenada.

**Problema 44:**

Confeccionar una página que muestre un título y un conjunto de párrafos. Mediante un selector de hermano adyacente proceder a definir la propiedad text-indent para el primer párrafo.

**Problema 45:**

Desarrollar un formulario que solicite el ingreso del nombre, mail y comentarios. Mediante un selector de atributo acceder a los elementos de tipo "input" y en aquellos que tenga definida la propiedad "type" y almacene el valor "text" proceder a fijar un color de fondo amarillo.

**Problema 46:**

Confeccionar una página que muestre un conjunto de párrafos. Luego mediante la pseudoclase first-child acceder al primer párrafo y aplicarle un borde superior de 1px de color negro. De forma similar acceder al último párrafo de la lista y aplicarle un borde inferior de 1px de color negro.

**Problema 47:**

Confeccionar una página que muestre dos listas ordenadas. En la primera mostrar los elementos pares subrayados y los impares tachados. En la segunda activar color verde para la fuente de los items múltiplos de cuatro.

**Problema 48:**

Confeccionar una página que contenga unos párrafos y en su interior varios elementos em y strong. Aplicar un estilo diferente al primer strong.

**Problema 49:**

Desarrollar una página que muestre una tabla HTML con un conjunto de filas y columnas. Dejar algunas celdas de la tabla vacía. Utilizando la pseudo-clase empty definir color de fondo amarillo para las celdas vacías.

**Problema 50:**

Definir una lista ordenada con los días de la semana. Mostrar el primer carácter de cada día subrayado.

**Problema 51:**

Confeccionar una página y su correspondiente configuración de estilo para que todos los párrafos comiencen con un guión.

**Problema 52:**

En CSS3 disponemos una propiedad llamada "column-count" que nos permite definir la cantidad de columnas que se debe mostrar un texto contenido en un "div".

Definir una serie de media queries para que se muestre una sola columna si el ancho del dispositivo es menor a 768px, 2 columnas si es menor a 1200px y 3 columnas en caso de ser mayor o igual a 1200 píxeles.