

Desarrollo Basado en Plataformas

Dr. Edward Hinojosa Cárdenas
ehinojosa@unsa.edu.pe

Primer Parcial

- Evaluación: Primer Parcial (Martes 24/04 14:00pm Grupo A – 16:40pm Grupo B). Evaluación Escrita (Todos los temas vistos en teoría, práctica, laboratorio y tareas). Traer tres pliegos de papel, todos los útiles de escritorio, calculadora (si es necesario - no celulares).
- Feriado día martes 1ro de Mayo.
- Corrección y Entrega de Notas en los horarios de laboratorio:
 - Miércoles 02/05 10:40 – Grupo A de Laboratorio
 - Miércoles 02/05 08:50 – Grupo B de Laboratorio
 - Los alumnos del Grupo C pueden asistir a cualquier horario anterior. No se aceptaran reclamos después de esas fechas.

CSS3

- CSS3 no solo cubre diseño y estilos web sino también forma y movimiento.
- La especificación de CSS3 es presentada en módulos que permiten a la tecnología proveer una especificación estándar por cada aspecto involucrado en la presentación visual del documento.

CSS3

- Desde esquinas redondeadas y sombras hasta transformaciones y reposicionamiento de los elementos ya presentados en pantalla.
- Este nivel de cambio convierte CSS3 en una tecnología prácticamente inédita comparada con versiones anteriores.

CSS3

- Las propiedades CSS3 son importantes y deben ser estudiadas una por una, pero para facilitar su aprendizaje vamos a aplicar todas ellas sobre la misma plantilla.
- Por este motivo comenzaremos por crear un documento HTML sencillo con algunos estilos básicos:

CSS3

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Nuevos Estilos CSS3</title>
  <link rel="stylesheet" href="nuevocss3.css">
</head>
<body>
  <header id="principal">
    <span id="titulo">Estilos CSS Web 2.0</span>
  </header>
</body>
</html>
```

CSS3

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
}  
#titulo {  
    font: bold 36px verdana, sans-serif;  
}
```

Estilos CSS Web 2.0

CSS3 – Prefijos Navegadores

- Los prefijos para los navegadores más comunes son los siguientes:
 - **-moz-** para Firefox.
 - **-webkit-** para Safari y Chrome.
 - **-o-** para Opera.
 - **-khtml-** para Konqueror.
 - **-ms-** para Internet Explorer.
 - **-chrome-** específico para Google Chrome.

CSS3 – Border-radius

- Por muchos años diseñadores han sufrido intentando lograr el efecto de esquinas redondeadas en las cajas de sus páginas web.
- La propiedad border-radius en este momento es experimental por lo que debemos usar los prefijos –moz- y -webkit- para que funcionen en navegadores basados en motores Gecko y WebKit, como Firefox, Safari y Google Chrome.

CSS3 – Border-radius

- Si todas las esquinas tienen la misma curvatura podemos utilizar un solo valor. Sin embargo, como ocurre con las propiedades margin y padding, podemos también declarar un valor diferente por cada una:

CSS3 – Border-radius

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-border-radius: 20px 10px 30px 50px;  
    -webkit-border-radius: 20px 10px 30px 50px;  
    border-radius: 20px 10px 30px 50px;  
}  
#titulo {  
    font: bold 36px verdana, sans-serif;  
}
```

CSS3 – Border-radius

Estilos CSS Web 2.0

CSS3 – Border-radius

- Los cuatro valores asignados a la propiedad border-radius representan diferentes ubicaciones.
- Recorriendo la caja en dirección de las agujas del reloj, los valores se aplicarán en el siguiente orden: esquina superior izquierda, esquina superior derecha, esquina inferior derecha y esquina inferior izquierda.

CSS3 – Border-radius

- También podemos dar forma a las esquinas declarando un segundo grupo de valores separados por una barra.
- Los valores a la izquierda de la barra representarán el radio horizontal mientras que los valores a la derecha representan el radio vertical:

CSS3 – Border-radius

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-border-radius: 20px / 10px;  
    -webkit-border-radius: 20px / 10px;  
    border-radius: 20px / 10px;  
}  
#titulo {  
    font: bold 36px verdana, sans-serif;  
}
```


CSS3 – Border-radius

Estilos CSS Web 2.0

CSS3 – Box-shadow

- Por años diseñadores han combinado imágenes, elementos y algunas propiedades CSS para generar sombras.
- Con la propiedad box-shadow podremos aplicar sombras a nuestras cajas con solo una simple línea de código:

CSS3 – Box-shadow

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-border-radius: 20px;  
    -webkit-border-radius: 20px;  
    border-radius: 20px;  
    -moz-box-shadow: rgb(150,150,150) 5px 5px;  
    -webkit-box-shadow: rgb(150,150,150) 5px 5px;  
    box-shadow: rgb(150,150,150) 5px 5px;  
}  
#titulo {  
    font: bold 36px verdana, sans-serif;  
}
```

CSS3 – Box-shadow

Estilos CSS Web 2.0

CSS3 – Box-shadow

- La propiedad box-shadow necesita al menos tres valores.
- El primero, es el color. Este valor fue construido aquí utilizando la función rgb() y números decimales, pero podemos escribirlo en números hexadecimales también.
- Los siguientes dos valores, expresados en pixeles, establecen el desplazamiento de la sombra. Este desplazamiento puede ser positivo o negativo.

CSS3 – Box-shadow

- Los valores indican, respectivamente, la distancia horizontal y vertical desde la sombra al elemento.
- Valores negativos posicionarán la sombra a la izquierda y arriba del elemento, mientras que valores positivos crearán la sombra a la derecha y debajo del elemento.

CSS3 – Box-shadow

- La sombra que obtuvimos hasta el momento es sólida, sin gradientes o transparencias (no realmente como una sombra suele aparecer).
- Existen algunos parámetros más y cambios que podemos implementar para mejorar la apariencia de la sombra.

CSS3 – Box-shadow

- Un cuarto valor que se puede agregar a la propiedad ya estudiada es la distancia de difuminación.
- Con este efecto ahora la sombra lucirá real. Puede intentar utilizar este nuevo parámetro declarando un valor de 10 píxeles.

```
box-shadow: rgb(150,150,150) 5px 5px 10px;
```


CSS3 – Box-shadow

Estilos CSS Web 2.0

CSS3 – Text-shadow

- Para crear sombras para figuras irregulares como textos, existe una propiedad especial llamada text-shadow:

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-border-radius: 20px;
    -webkit-border-radius: 20px;
    border-radius: 20px;
    -moz-box-shadow: rgb(150,150,150) 5px 5px 10px;
    -webkit-box-shadow: rgb(150,150,150) 5px 5px 10px;
    box-shadow: rgb(150,150,150) 5px 5px 10px;
}
#titulo {
    font: bold 36px verdana, sans-serif;
    text-shadow: rgb(0,0,150) 3px 3px 5px;
}
```

Estilos CSS Web 2.0

CSS3 – Text-shadow

- Los valores para text-shadow son similares a los usados para box-shadow.
- Podemos declarar el color de la sombra, la distancia horizontal y vertical de la sombra con respecto al objeto.

CSS3 – @font-face

- Usuarios regulares de la web a menudo tienen un número limitado de fuentes instaladas en sus ordenadores, usualmente estas fuentes son diferentes de un usuario a otro, y la mayoría de las veces muchos usuarios tendrán fuentes que otros no.
- Por años, los sitios webs solo pudieron utilizar un limitado grupo de fuentes confiables (un grupo básico que prácticamente todos los usuarios tienen instalados) y así presentar la información en pantalla.

CSS3 – @font-face

- La propiedad @font-face permite a los diseñadores proveer un archivo conteniendo una fuente específica para mostrar sus textos en la página.
- Ahora podemos incluir cualquier fuente que necesitemos con solo proveer el archivo adecuado:

<https://www.ffonts.net>

CSS3 – @font-face

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-border-radius: 20px;
    -webkit-border-radius: 20px;
    border-radius: 20px;
    -moz-box-shadow: rgb(150,150,150) 5px 5px 10px;
    -webkit-box-shadow: rgb(150,150,150) 5px 5px 10px;
    box-shadow: rgb(150,150,150) 5px 5px 10px;
}
#titulo {
    font: bold 36px MiNuevaFuente, verdana, sans-serif;
    text-shadow: rgb(0,0,150) 3px 3px 5px;
}
@font-face {
    font-family: MiNuevaFuente;
    src: url(font.otf);
}
```

CSS3 – @font-face

ESTILOS CSS WEB 2.0

CSS3 – Gradiente lineal

- Una propiedad background con algunos pocos parámetros es suficiente para convertir su documento en una página web con aspecto profesional.
- Los gradientes son configurados como fondos, por lo que podemos usar las propiedades background o background-image para declararlos. La sintaxis para los valores declarados en estas propiedades es linear-gradient(posición inicio, color inicial, color final).

CSS3 – Gradiente lineal

- Los atributos de la función `linear-gradient()` indican el punto de comienzo y los colores usados para crear el gradiente. El primer valor puede ser especificado en píxeles, porcentaje o usando las palabras clave `top`, `bottom`, `left` y `right`.
- El punto de comienzo puede ser reemplazado por un ángulo para declarar una dirección específica del gradiente. También podemos declarar los puntos de terminación para cada color.

CSS3 – Gradiente lineal

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-border-radius: 20px;
    -webkit-border-radius: 20px;
    border-radius: 20px;
    -moz-box-shadow: rgb(150,150,150) 5px 5px 10px;
    -webkit-box-shadow: rgb(150,150,150) 5px 5px 10px;
    box-shadow: rgb(150,150,150) 5px 5px 10px;
    background: -webkit-linear-gradient(top, #FFFFFF, #006699);
    background: -moz-linear-gradient(top, #FFFFFF, #006699);
}
#titulo {
    font: bold 36px MiNuevaFuente, verdana, sans-serif;
    text-shadow: rgb(0,0,150) 3px 3px 5px;
}
@font-face {
    font-family: 'MiNuevaFuente';
    src: url('font.otf');
}
```

CSS3 – Gradiente lineal

ESTILOS CSS WEB 2.0

CSS3 – Gradiente lineal

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-border-radius: 20px;
    -webkit-border-radius: 20px;
    border-radius: 20px;
    -moz-box-shadow: rgb(150,150,150) 5px 5px 10px;
    -webkit-box-shadow: rgb(150,150,150) 5px 5px 10px;
    box-shadow: rgb(150,150,150) 5px 5px 10px;
    background: -webkit-linear-gradient(30deg, #FFFFFF, #006699);
    background: -moz-linear-gradient(30deg, #FFFFFF, #006699);
}
#titulo {
    font: bold 36px MiNuevaFuente, verdana, sans-serif;
    text-shadow: rgb(0,0,150) 3px 3px 5px;
}
@font-face {
    font-family: 'MiNuevaFuente';
    src: url('font.otf');
}
```

CSS3 – Gradiente lineal

ESTILOS CSS WEB 2.0

CSS3 – Gradiente lineal

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-border-radius: 20px;
    -webkit-border-radius: 20px;
    border-radius: 20px;
    -moz-box-shadow: rgb(150,150,150) 5px 5px 10px;
    -webkit-box-shadow: rgb(150,150,150) 5px 5px 10px;
    box-shadow: rgb(150,150,150) 5px 5px 10px;
    background: -webkit-linear-gradient(top, #FFFFFF 50%, #006699 80%);
    background: -moz-linear-gradient(top, #FFFFFF 50%, #006699 80%);
}
#titulo {
    font: bold 36px MiNuevaFuente, verdana, sans-serif;
    text-shadow: rgb(0,0,150) 3px 3px 5px;
}
@font-face {
    font-family: 'MiNuevaFuente';
    src: url('font.otf');
}
```

CSS3 – Gradiente lineal

ESTILOS CSS WEB 2.0

CSS3 – Gradiente radial

- La sintaxis estándar para los gradientes radiales solo difiere en unos pocos aspectos con respecto a la anterior. Debemos usar la función `radial-gradient()` y un nuevo atributo para la forma.
- La posición de comienzo es el origen y puede ser declarada en pixeles, porcentaje o una combinación de las palabras clave `center`, `top`, `bottom`, `left` y `right`. Existen dos posibles valores para la forma (`circle` y `ellipse`) y la terminación para el color indica el color y la posición donde las transiciones comienzan.

CSS3 – Gradiente radial

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-border-radius: 20px;
    -webkit-border-radius: 20px;
    border-radius: 20px;
    -moz-box-shadow: rgb(150,150,150) 5px 5px 10px;
    -webkit-box-shadow: rgb(150,150,150) 5px 5px 10px;
    box-shadow: rgb(150,150,150) 5px 5px 10px;
    background: -webkit-radial-gradient(center, circle, #FFFFFF 0%, #006699 200%);
    background: -moz-radial-gradient(center, circle, #FFFFFF 0%, #006699 200%);
}
#titulo {
    font: bold 36px MiNuevaFuente, verdana, sans-serif;
    text-shadow: rgb(0,0,150) 3px 3px 5px;
}
@font-face {
    font-family: 'MiNuevaFuente';
    src: url('font.otf');
}
```

CSS3 – Gradiente radial

ESTILOS CSS WEB 2.0

CSS3 – RGBA

- Hasta este momento los colores fueron declarados como sólidos utilizando valores hexadecimales o la función `rgb()` para decimales.
- CSS3 ha agregado una nueva función llamada `rgba()` que simplifica la asignación de colores y transparencias. Esta función además resuelve un problema previo provocado por la propiedad `opacity`.

CSS3 – RGBA

- La función `rgba()` tiene cuatro atributos. Los primeros tres son similares a los usados en `rgb()` y simplemente declaran los valores para los colores rojo, verde y azul en números decimales del 0 al 255.
- El último, en cambio, corresponde a la nueva capacidad de opacidad. Este valor se debe encontrar dentro de un rango que va de 0 a 1, con 0 como totalmente transparente y 1 como totalmente opaco.

CSS3 – RGBA

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-border-radius: 20px;
    -webkit-border-radius: 20px;
    border-radius: 20px;
    -moz-box-shadow: rgb(150,150,150) 5px 5px 10px;
    -webkit-box-shadow: rgb(150,150,150) 5px 5px 10px;
    box-shadow: rgb(150,150,150) 5px 5px 10px;
    background: -webkit-radial-gradient(center, circle, #FFFFFF 0%, #006699 200%);
    background: -moz-radial-gradient(center, circle, #FFFFFF 0%, #006699 200%);
}
#titulo {
    font: bold 36px MiNuevaFuente, verdana, sans-serif;
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;
}
@font-face {
    font-family: 'MiNuevaFuente';
    src: url('font.otf');
}
```

ESTILOS CSS WEB 2.0

CSS3 – Outline

- Esta propiedad era usada para crear un segundo borde, y ahora ese borde puede ser mostrado alejado del borde real del elemento.
- La propiedad outline tiene similares características y usa los mismos parámetros que border. La propiedad outline-offset solo necesita un valor en pixeles.

CSS3 – Outline

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    outline: 2px dashed #000099;  
    outline-offset: 15px;  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.otf');  
}
```

CSS3 – Outline

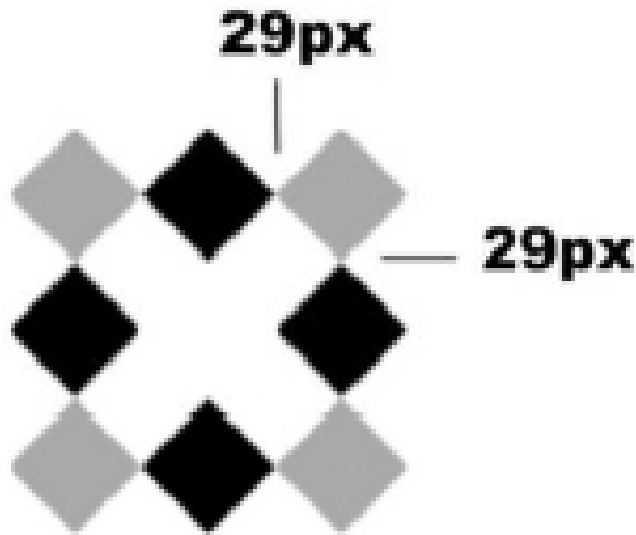
ESTILOS CSS WEB 2.0

CSS3 – Border-image

- Los posibles efectos logrados por las propiedades border y outline están limitados a líneas simples y solo algunas opciones de configuración.
- La nueva propiedad border-image fue incorporada para superar estas limitaciones y dejar en manos del diseñador la calidad y variedad de bordes disponibles ofreciendo la alternativa de utilizar imágenes propias.

CSS3 – Border-image

- La propiedad border-image toma una imagen y la utiliza como patrón. De acuerdo a los valores otorgados, la imagen es cortada como un pastel, las partes obtenidas son luego ubicadas alrededor del objeto para construir el borde.



CSS3 – Border-image

- La propiedad border-image toma una imagen y la utiliza como patrón.
- De acuerdo a los valores otorgados, la imagen es cortada como un pastel, las partes obtenidas son luego ubicadas alrededor del objeto para construir el borde.

CSS3 – Border-image

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
  
    border: 29px solid transparent;  
    -moz-border-image: url('diamonds.png') 29 stretch;  
    -webkit-border-image: url('diamonds.png') 29 stretch;  
    border-image: url('diamonds.png') 29 stretch;  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.ttf');  
}
```

CSS3 – Border-image



CSS3 – Border-image

- Estamos definiendo un borde de 29 pixeles para la caja de nuestra cabecera y luego cargando la imagen diamonds.png para construir ese borde.
- El valor 29 en la propiedad border-image declara el tamaño de las piezas y stretch es uno de los métodos disponibles para distribuir estas piezas alrededor de la caja.
- Existen tres valores posibles para el último atributo.

CSS3 – Border-image

- La palabra clave repeat repetirá las piezas tomadas de la imagen todas las veces que sea necesario para cubrir el lado del elemento. En este caso, el tamaño de las piezas es preservado y la imagen será cortada si no existe más espacio para ubicarla.
- La palabra clave round considerará qué tan largo es el lado a ser cubierto y ajustará el tamaño de las piezas para asegurarse que cubren todo el lado y ninguna pieza es cortada.
- Finalmente, la palabra clave stretch estira solo una pieza para cubrir el lado completo.

CSS3 – Transform y transition

- La propiedad transform puede operar cuatro transformaciones básicas en un elemento: scale (escalar), rotate (rotar), skew (inclinarse) y translate (trasladar o mover).
- Veremos como funciona cada uno de ellas.

CSS3 – Transform: scale

- La función scale recibe dos parámetros: el valor X para la escala horizontal y el valor Y para la escala vertical. Si solo un valor es provisto el mismo valor es aplicado a ambos parámetros.
- Números enteros y decimales pueden ser declarados para la escala. Esta escala es calculada por medio de una matriz. Los valores entre 0 y 1 reducirán el elemento, un valor de 1 mantendrá las proporciones originales y valores mayores que 1 aumentarán las dimensiones del elemento de manera incremental.

CSS3 – Transform: scale

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-transform: scale(2);  
    -webkit-transform: scale(2);  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.ttf');  
}
```

CSS3 – Transform: scale

Estilos CSS Web 2.0

CSS3 – Transform: scale

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-transform: scale(1,-1);  
    -webkit-transform: scale(1,-1);  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.ttf');  
}
```

CSS3 – Transform: scale

Estijos C22 M6P S'O

CSS3 – Transform: scale

- El primer valor, 1, mantiene la proporción original para la dimensión horizontal de la caja. El segundo valor también mantiene la proporción original, pero invierte el elemento verticalmente para producir el efecto espejo.
- Existen también otras dos funciones similares a scale pero restringidas a la dimensión horizontal o vertical: scaleX y scaleY. Estas funciones, por supuesto, utilizan un solo parámetro.

CSS3 – Transform: rotate

- La función rotate rota el elemento en la dirección de las agujas de un reloj. El valor debe ser especificado en grados usando la unidad “deg”.
- Si un valor negativo es declarado, solo cambiará la dirección en la cual el elemento es rotado.

CSS3 – Transform: rotate

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-transform: rotate(10deg);  
    -webkit-transform: rotate(10deg);  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.ttf');  
}
```

CSS3 – Transform: rotate

Estilos CSS Web 2.0

CSS3 – Transform: skew

- Esta función cambia la simetría del elemento en grados y en ambas dimensiones.
- La función skew usa dos parámetros, pero a diferencia de otras funciones, cada parámetro de esta función solo afecta una dimensión (los parámetros actúan de forma independiente).
- En el ejemplo anterior, realizamos una operación transform a la caja de la cabecera para inclinarla. Solo declaramos el primer parámetro, por lo que solo la dimensión horizontal de la caja será modificada. Si usáramos los dos parámetros, podríamos alterar ambas dimensiones del objeto.

CSS3 – Transform: skew

- La función skew usa dos parámetros, pero a diferencia de otras funciones, cada parámetro de esta función solo afecta una dimensión (los parámetros actúan de forma independiente).
- En el ejemplo anterior, realizamos una operación transform a la caja de la cabecera para inclinarla. Solo declaramos el primer parámetro, por lo que solo la dimensión horizontal de la caja será modificada. Si usáramos los dos parámetros, podríamos alterar ambas dimensiones del objeto. Como alternativa podemos utilizar funciones diferentes para cada una de ellas: skewX y skewY.

CSS3 – Transform: skew

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-transform: skew(20deg);  
    -webkit-transform: skew(20deg);  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.ttf');  
}
```

CSS3 – Transform: skew

Estilos CSS Web 2.0

CSS3 – Transform: translate

- La función translate mueve o desplaza el elemento en la pantalla a una nueva posición.
- La función translate considera la pantalla como una grilla de pixeles, con la posición original del elemento usada como un punto de referencia. La esquina superior izquierda del elemento es la posición 0,0, por lo que valores negativos moverán al objeto hacia la izquierda o hacia arriba de la posición original, y valores positivos lo harán hacia la derecha o hacia abajo.

CSS3 – Transform: translate

- Movimos la caja de la cabecera hacia la derecha unos 100 pixeles desde su posición original.
- Dos valores pueden ser declarados en esta función si queremos mover el elemento horizontal y verticalmente, o podemos usar funciones independientes llamadas `translateX` y `translateY`.

CSS3 – Transform: translate

```
body {
    text-align: center;
}
#principal {
    display: block;
    width: 500px;
    margin: 50px auto;
    padding: 15px;
    text-align: center;
    border: 1px solid #999999;
    background: #DDDDDD;
    -moz-transform: translate(100px);
    -webkit-transform: translate(100px);
}
#titulo {
    font: bold 36px MiNuevaFuente, verdana, sans-serif;
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;
}
@font-face {
    font-family: 'MiNuevaFuente';
    src: url('font.ttf');
}
```

CSS3 – Transform: translate

Estilos CSS Web 2.0

CSS3 – Transformando todo al mismo tiempo

- A veces podría resultar útil realizar sobre un elemento varias transformaciones al mismo tiempo.
- Para obtener una propiedad transform combinada, solo tenemos que separar cada función a aplicar con un espacio.
- Una de las cosas que debe recordar en este caso es que el orden es importante. Esto es debido a que algunas funciones mueven el punto original y el centro del objeto, cambiando de este modo los parámetros que el resto de las funciones utilizarán para operar.

CSS3 – Transformando todo al mismo tiempo

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
    -moz-transform: translateY(100px) rotate(45deg) scaleX(0.3);  
    -webkit-transform: translateY(100px) rotate(45deg) scaleX(0.3);  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.ttf');  
}
```

CSS3 – Transformando todo al mismo tiempo



CSS3 – Transformando todo al mismo tiempo

- A veces podría resultar útil realizar sobre un elemento varias transformaciones al mismo tiempo.
- Para obtener una propiedad transform combinada, solo tenemos que separar cada función a aplicar con un espacio.
- Una de las cosas que debe recordar en este caso es que el orden es importante. Esto es debido a que algunas funciones mueven el punto original y el centro del objeto, cambiando de este modo los parámetros que el resto de las funciones utilizarán para operar.

CSS3 – Transformaciones dinámicas

- Lo que hemos aprendido hasta el momento en este capítulo cambiará la forma de la web, pero la mantendrá tan estática como siempre.
- Sin embargo, podemos aprovecharnos de la combinación de transformaciones y pseudo clases para convertir nuestra página en una aplicación dinámica:

CSS3 – Transformaciones dinámicas

- La pseudo-clase :hover de CSS coincide cuando el usuario interactúa con un elemento con un dispositivo señalador, pero no necesariamente lo activa.
- Generalmente se activa cuando el usuario se desplaza sobre un elemento con el cursor (puntero del mouse).

CSS3 – Transformaciones dinámicas

```
body {  
    text-align: center;  
}  
#principal {  
    display: block;  
    width: 500px;  
    margin: 50px auto;  
    padding: 15px;  
    text-align: center;  
    border: 1px solid #999999;  
    background: #DDDDDD;  
}  
#principal:hover{  
    -moz-transform: rotate(5deg);  
    -webkit-transform: rotate(5deg);  
}  
#titulo {  
    font: bold 36px MiNuevaFuente, verdana, sans-serif;  
    text-shadow: rgba(0,0,0,0.5) 3px 3px 5px;  
}  
@font-face {  
    font-family: 'MiNuevaFuente';  
    src: url('font.ttf');  
}
```

CSS3 – Transformaciones dinámicas

Estilos CSS Web 2.0

Práctica 4 – Grupo A

- Cree la siguiente página web.
- Revisión media hora antes de acabar la clase.
- Cada vez que se llame a un alumno del grupo, todos los integrantes tendrán 3 puntos menos.

**Apellidos de todos los
Alumnos del Grupo**

**APELLIDOS DE TODOS
LOS ALUMNOS DEL
GRUPO**

**Apellidos de todos los
Alumnos del Grupo**

**APELLIDOS DE TODOS LOS
ALUMNOS DEL GRUPO**

Práctica 4 – Grupo B

- Cree la siguiente página web.
- Revisión media hora antes de acabar la clase.
- Cada vez que se llame a un alumno del grupo, todos los integrantes tendrán 3 puntos menos.

**APELLIDOS DE TODOS
LOS ALUMNOS DEL
GRUPO**

*Apellidos de todos los
Alumnos del Grupo*

**APELLIDOS DE TODOS LOS
ALUMNOS DEL GRUPO**

**APELLIDOS DE TODOS LOS
ALUMNOS DEL GRUPO**

Gracias

Dr. Edward Hinojosa Cárdenas
ehinojosa@unsa.edu.pe