

Actividad: **Diseño de reloj analógico**

Utilizando lenguajes de marcación HTML y CSS, crear un reloj analógico formado por un solo elemento estructura, teniendo en cuenta:

- Un fondo cuadrado opaco con profundidad 0, ocupando el 60% de la ventana.
 - Tres imágenes superpuestas equivalentes a las tres manecillas del reloj (profundidad)
 - Crear un evento que dispare una transformación del segundero (z-index mayor)
 - Efectuar un disparo de evento en cascada de una transformación de las capas inferiores (z-index menor) por cada vuelta de la imagen superior.
 - Añadir una propiedad que efectúe las tres transformaciones de forma lineal (incluyendo z-index mayor)
 - Cambiar el evento de disparo manual por un evento programado.
 - Capturar la hora del sistema, y aplicando a las tres imágenes (capas)
 - Añadir un intervalo Delay que lleve el elemento a la esquina inferior derecha, aplicando una transformación de tamaño (10%)
 - Una vez situado el elemento según punto anterior, disparar un evento que vuelva a situarlo en el centro de la pantalla y con las dimensiones iniciales.
 - Explicar brevemente el proceso de cada etapa del diseño.
-

Desarrollo de Aplicaciones Web - DISEÑO DE INTERFACES WEB

Actividad: **Diseño de reloj analógico**

viernes, 30 de noviembre de 2018

- **Un fondo cuadrado opaco con profundidad 0, ocupando el 60% de la ventana.**

```
.reloj {
  height: 60%;
}
#esfera, #segundero, #minutero, #horario{
  position: absolute;
  transition: transform 1s linear;
}
#esfera{
  z-index: 0;
}
```

La clase reloj es la que le da el tamaño a el fondo, y el *z-index: 0*; es el que le da profundidad. La imagen es png por lo cual es un fondo opaco en si.

- **Tres imágenes superpuestas equivalentes a las tres manecillas del reloj (profundidad)**

```
#horario{
  z-index: 1;
}
#minutero{
  z-index: 2;
}
#segundero{
  z-index: 3;
}
#milisegundero{
  z-index: 4;
}
```

Le añado a las agujas por su id la profundidad de cada una.

- **Crear un evento que dispare una transformación del segundero (z-index mayor)**

```
window.onload = function() {
  empezar.addEventListener('click', iniciarReloj);
}
```

empezar es un boton e **iniciarReloj()** crea un intervalo que valla moviendo cada segundo las cuatro agujas una proporción equivalente a los segundos que hay en un día partido entre 360 grados de la esfera, para así que todas se muevan a la vez su cantidad proporcional. Le tengo metido en los estilos una transición cada vez que ocurre una transformación se realice durante un segundo, que es a la velocidad que tengo el intervalo para que las agujas se vean linealmente, sin saltos.

```
#esfera, #milisegundero, #segundero, #minutero, #horario{
  position: absolute;
  transition: transform 1s linear;
}
```

- **Efectuar un dispararo de evento en cascada de una transformación de las capas inferiores (z-index menor) por cada vuelta de la imagen superior.**

Como he explicado arriba, para que se viera lineal he hecho que cada aguja se valla moviendo cada segundo su parte proporcional.

- **Añadir una propiedad que efectúe las tres transformaciones de forma lineal (incluyendo z-index mayor)**

```
#esfera, #milisegundero, #segundero, #minutero, #horario{  
  position: absolute;  
  transition: transform 1s linear;  
}
```

Añade una transición cada vez que se realiza una transformación, aplicándole un 1 segundo hasta que finaliza, que es el tiempo en el que se actualiza el reloj interno de mi reloj.

- **Cambiar el evento de disparo manual por un evento programado.**

```
window.onload = function() {  
  iniciarReloj();  
};
```

La función iniciarReloj inicializa el reloj y lo pone en la esquina inferior derecha tras 5 segundos.

- **Capturar la hora del sistema, y aplicando a las tres imágenes (capas)**

```
function iniciarReloj () {  
  parar();  
  var date = new Date();  
  var ms = 0;  
  var seg = (360/60)*date.getSeconds();  
  var min = (360/60)*(date.getMinutes()+date.getSeconds()/60);  
  var hora = (360/12)*(date.getHours()+date.getMinutes()/60);  
  interval = setInterval(function(){  
    ms+=360;  
    seg+=6;  
    min+=0.1;  
    hora+=0.00836820084;  
    milisegundero.style.transform = `rotate(${ms}deg)`;  
    segundero.style.transform = `rotate(${seg}deg)`;  
    minutero.style.transform = `rotate(${min}deg)`;  
    horario.style.transform = `rotate(${hora}deg)`;  
  }, 1000);  
  timeout = setTimeout(hacerPeque, 5000);  
}
```

Esa función captura la fecha del sistema.

- **Añadir un intervalo Delay que lleve el elemento a la esquina inferior derecha, aplicando una transformación de tamaño (10%)**

```
timeout = setTimeout(hacerPeque, 5000);
```

La función de arriba también lanza un Timeout que hace mas pequeño el reloj y lo pone debajo a la derecha.

- **Una vez situado el elemento según punto anterior, disparar un evento que vuelva a situarlo en el centro de la pantalla y con las dimensiones iniciales.**

```
agrandar.addEventListener('click', function(){  
    if (segundero.getAttribute('class') == 'reloj') {  
        hacerPeque();  
    } else {  
        hacerGrande();  
    }  
});
```

Ese botón pone el reloj grande o pequeño según su estado actual.

- **Explicar brevemente el proceso de cada etapa del diseño.**

He ido explicando la mayoría de etapas en cada apartado, por lo cual explicar los temas no tocados.

1) Para colocar el reloj abajo a la derecha he usado:

```
.pequeño {  
    height: 10%;  
    right: 0px;  
    bottom: 0px;  
}
```