

10.1. ¿En qué medida crees que es determinante la frecuencia de un sonido?

La frecuencia de un sonido es un factor fundamental que influye directamente en la percepción que tenemos de él. Aquí te presento algunos puntos clave:

Calidad de audio: Una frecuencia de muestreo adecuada (que determina cuántas veces por segundo se mide la señal de audio) es crucial para una buena calidad de sonido. Una frecuencia de muestreo baja puede generar distorsión y pérdida de información.

Uso en sitios web: En el contexto de un sitio web, la frecuencia puede influir en la experiencia del usuario. Por ejemplo, sonidos muy agudos o graves pueden resultar molestos, mientras que una gama de frecuencias equilibrada puede crear una atmósfera más agradable.

10.2. Se tiene un archivo de audio en formato MP3 cuyo tamaño es de 5,2MB. Calcula la duración exacta de este fichero.

No es posible calcular la duración exacta de un archivo MP3 de 5,2 MB sin información adicional.

La duración de un archivo de audio depende de varios factores, como:

Tasa de bits: La cantidad de información por segundo que se utiliza para codificar el audio.

Frecuencia de muestreo: La cantidad de muestras por segundo que se toman de la señal de audio original.

Compresión: El algoritmo de compresión utilizado para reducir el tamaño del archivo.

10.3.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Tres Audios</title>
```

```
</head>
```

```
<body>
```

```
  <audio controls id="audio1">
```

```
    <source src="/audio.mp3" type="audio/mpeg">
```

```
    Tu navegador no soporta este elemento de audio.
```

```
  </audio>
```

```
  <audio controls id="audio2">
```

```
    <source src="/audio.mp3" type="audio/mpeg">
```

Tu navegador no soporta este elemento de audio.

</audio>

<audio controls id="audio3">

<source src="/audio.mp3" type="audio/mpeg">

Tu navegador no soporta este elemento de audio.

</audio>

<script>

```
function disableAudio(audioElement) {  
    audioElement.controls = false;  
}
```

```
document.getElementById("audio1").addEventListener("ended", function () {  
    disableAudio(this);  
});
```

```
document.getElementById("audio2").addEventListener("ended", function () {  
    disableAudio(this);  
});
```

```
document.getElementById("audio3").addEventListener("ended", function () {  
    disableAudio(this);  
});
```

</script>

</body>

</html>

10.4.

Podríamos considerar las siguientes combinaciones y situaciones:

1. Video corto y llamativo en la página de inicio:

- Opción: Formato MP4, resolución 720p, reproducción automática con sonido.
- Justificación: Atrae la atención del usuario de inmediato y proporciona una experiencia visual agradable.

2. Tutorial detallado en una sección de ayuda:

- Opción: Formato WebM (mejor compatibilidad con navegadores), resolución 720p, reproducción manual sin sonido (para que el usuario pueda pausar y rebobinar).
- Justificación: Permite al usuario controlar la reproducción y concentrarse en la información.

3. Video promocional en una página de producto:

- Opción: Formato MP4, resolución 1080p, reproducción automática con sonido (opcionalmente con subtítulos).
- Justificación: Muestra el producto en alta calidad y proporciona una experiencia inmersiva.

10.5.

```
h1 {  
  animation-duration: 6s;  
  animation-name: estiloconanimacion;  
  animation-iteration-count: 5;  
  /* Se repite 5 veces */  
  animation-direction: alternate;  
  /* Se alterna entre hacia adelante y hacia atrás */  
}  
@keyframes estiloconanimacion {  
  from {  
    margin-left: 50%;  
    width: 200%;  
  }  
  50% {  
    font-size: 200%;  
    margin-left: 0;  
    width: 150%;  
  }  
}
```

```
to {  
margin-left: 0;  
width: 100%;  
}  
}
```

Actividad 10.6 y 10.7

```
<!DOCTYPE html>  
  
<html lang="es">  
  
<head>  
  
  <meta charset="UTF-8">  
  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  
  <title>Actividades CSS</title>  
  
  <style>  
  
    /* Actividad 10.6 */  
  
    .rectangulo {  
  
      width: 400px;  
  
      height: 300px;  
  
      background-color: red;  
  
      transition: transform 0.3s, background-color 0.3s;  
  
    }  
  
  
    .rectangulo:hover {  
  
      transform: scale(0.5);  
  
      background-color: blue;  
  
    }  
  
  
  
    /* Actividad 10.7 */  
  
    .cuadrado {  
  
      width: 30px;  
  
      height: 30px;
```

```
        background-color: red;

        margin: 10px;

        display: inline-block;

        transition: all 0.5s ease;
    }

    .cuadrado:hover {

        width: 60px;

        height: 30px;

        background-color: blue;

    }

</style>
</head>
<body>

    <h2>Actividad 10.6</h2>

    <div class="rectangulo"></div>


    <h2>Actividad 10.7</h2>

    <div class="cuadrado"></div>

    <div class="cuadrado"></div>

</body>
</html>
```

10.8

Etiquetas HTML:

- Son la estructura del documento. Elementos como <div>, o <button> pueden ser animados.

Reglas CSS para animaciones:

- transition: Permite animaciones suaves al cambiar propiedades como color, tamaño o posición.
- @keyframes: Se usa para definir animaciones más complejas y personalizadas.

Atributos y propiedades de animación:

- transform: Permite escalar, rotar y mover elementos.
- animation: Define el uso de @keyframes para animaciones detalladas.
- transition-duration, transition-delay, transition-timing-function: Controlan la velocidad y el comportamiento de la animación.

10.9

```
<!DOCTYPE html>
```

```
<html lang="es">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Actividades CSS</title>
```

```
<style>
```

```
dialog {  
    opacity: 0;  
    transform: translateY(-20px);  
    transition: opacity 0.5s ease, transform 0.5s ease;  
}
```

```
dialog[open] {  
    opacity: 1;  
    transform: translateY(0);  
}
```

```
/* Flecha animada */  
.toggle {  
    cursor: pointer;  
    font-size: 24px;  
    display: inline-block;  
    transition: transform 0.3s ease;
```

```
}

.details {
  display: none;
  margin-top: 10px;
}

.expanded {
  transform: rotate(90deg);
}
</style>
<script>
function toggleDetails() {
  var details = document.getElementById("details");
  var arrow = document.getElementById("arrow");
  if (details.style.display === "none" || details.style.display === "") {
    details.style.display = "block";
    arrow.classList.add("expanded");
  } else {
    details.style.display = "none";
    arrow.classList.remove("expanded");
  }
}
</script>
</head>
<body>
<h2>Actividad 10.9</h2>
<span id="arrow" class="toggle" onclick="toggleDetails()">▶</span>
<div id="details" class="details">
  <button onclick="document.getElementById('dialogo').showModal()">Abrir
  diálogo</button>
</div>
</body>
</html>
```

Adrián Peñalver Fernández

```
<dialog id="dialogo">  
  <p>Este es un cuadro de diálogo animado.</p>  
  <button onclick="document.getElementById('dialogo').close()">Cerrar</button>  
</dialog>  
</div>  
</body>  
</html>
```