

# Informe sobre Unidades de medida responsivas

## Programación

En esta tarea se debe entregar el informe sobre su investigación de Unidades de medidas responsivas

### 1) Conceptos de cada unidad de medida y aplicación.

- **Píxeles (px):** El píxel es una unidad de medida absoluta que representa un punto en la pantalla. Es una de las unidades más comunes para definir tamaños en CSS, ya que ofrece un control preciso y consistente en diferentes dispositivos. Se utiliza principalmente para definir tamaños de fuentes, márgenes, padding, bordes, y otros elementos que requieren precisión y uniformidad en diferentes pantallas.
- **Porcentaje (%):** El porcentaje es una unidad de medida relativa que se basa en un valor de referencia, usualmente el tamaño del contenedor padre o la dimensión del elemento en el que se aplica. Por ejemplo, si se establece el ancho de un elemento en 50%, este tomará el 50% del ancho de su contenedor. Es útil para crear diseños flexibles y responsivos, ya que los elementos pueden ajustarse dinámicamente al tamaño del contenedor en diferentes pantallas. Se utiliza comúnmente para definir anchos y márgenes de elementos.
- **Viewport Width (vw) y Viewport Height (vh):** 1vw es igual al 1% del ancho del viewport (la ventana gráfica o el área visible de la página). 1vh es igual al 1% de la altura del viewport. Las unidades vw y vh son ideales para ajustar el tamaño de los elementos según las dimensiones del

viewport, independientemente del tamaño del contenedor padre. Se usan comúnmente para crear elementos que deben mantener una proporción con el tamaño de la ventana gráfica, como fondos de pantalla, secciones de página o tipografía.

- **Relative to Font Size (em y rem):**

- em: Unidad relativa al tamaño de fuente del elemento padre inmediato. Si el tamaño de la fuente del padre es 16px, 1em será igual a 16px.
- rem: Unidad relativa al tamaño de fuente del elemento raíz (html). A diferencia de em, no depende del tamaño de fuente de los elementos padres, lo que la hace más predecible y consistente.

Se utilizan para definir tamaños de fuente, márgenes, padding, y otros valores que deben escalar de manera proporcional al tamaño de la fuente o mantenerse consistentes en toda la página.

- **Min, Max, y Clamp:**

- min(): Devuelve el menor valor de una lista de expresiones.
  - max(): Devuelve el mayor valor de una lista de expresiones.
  - clamp(): Establece un valor dentro de un rango especificado con un valor mínimo y máximo.
- Se utiliza para crear diseños que se adapten a diferentes tamaños de pantalla, pero que mantengan proporciones adecuadas.

Útil para asegurar que los tamaños de los elementos no se vuelvan demasiado pequeños o grandes en diferentes tamaños de pantalla, garantizando un diseño más adaptable.

## 2) Ventajas y desventajas de cada una.

### **Píxeles (px):**

Ventajas: Ofrece un control preciso sobre el tamaño de los elementos. Consistente en todas las pantallas y dispositivos.

Nombre: Giovanni Pauletto

Desventajas: No es flexible para diseños responsivos. Puede requerir ajustes adicionales para escalar correctamente en pantallas de alta resolución (Retina).

### **Porcentaje (%):**

Ventajas: Facilita la creación de diseños fluidos y adaptables. Relativo al contenedor, lo que lo hace flexible para diferentes dispositivos.

Desventajas: Puede ser complicado de manejar en diseños complejos con múltiples anidaciones. Dependencia total del tamaño del contenedor, lo que puede generar problemas en algunos layouts.

### **Viewport Width (vw) y Viewport Height (vh):**

Ventajas: Ajusta los elementos directamente con el tamaño de la pantalla, lo que es ideal para layouts responsivos. No depende de contenedores padres, lo que facilita el control total del diseño en base al viewport.

Desventajas: Puede resultar en tamaños de elementos excesivamente grandes o pequeños en pantallas extremas. No siempre es ideal para todos los elementos, especialmente en layouts donde se requiere consistencia en relación a otros elementos.

### **em y rem (Relative to Font Size):**

Ventajas: em permite crear tamaños que escalen proporcionalmente dentro de un contenedor. rem ofrece consistencia en el diseño, ya que es independiente de la cascada de estilos y se basa en la raíz del documento.

Desventajas: em puede volverse complicado en layouts anidados, donde los tamaños se escalan de manera imprevista. rem puede requerir un mayor control sobre el tamaño de fuente base para lograr los resultados deseados.

### **Min, Max, y Clamp:**

Ventajas: min() y max() permiten establecer valores flexibles que se adaptan a diferentes condiciones.

Nombre: Giovanni Pauletto

clamp() asegura que los tamaños se mantengan dentro de un rango deseado, lo que previene diseños disfuncionales en pantallas extremas.

Desventajas: Pueden ser complejas de entender y aplicar correctamente, especialmente en diseños muy personalizados. Requieren una buena planificación para que los valores mínimos, máximos y la función clamp sean efectivos.

### 3) Ejemplos de código utilizando lo aprendido.

#### **Píxeles (px):**

```
.header {  
    font-size: 20px; /* Tamaño fijo y consistente en cualquier dispositivo */  
    padding: 10px 15px; /* Márgenes definidos en píxeles */  
}
```

#### **em y rem (Relative to Font Size):**

```
body {  
    font-size: 16px;  
}  
  
h1 {  
    font-size: 2rem; /* 32px, relativo al tamaño de fuente del elemento raíz */  
}  
  
p {  
    font-size: 1.5em; /* 24px si el tamaño de fuente del padre es 16px */  
}
```

#### **Viewport Width (vw) y Viewport Height (vh):**

```
.hero {
```

Nombre: Giovanni Pauletto

```
height: 100vh; /* El elemento ocupará el 100% de la altura del viewport */

width: 100vw; /* El elemento ocupará el 100% del ancho del viewport */

}

.text-large {

    font-size: 5vw; /* El tamaño de la fuente será el 5% del ancho del viewport */

}
```

### Porcentaje (%):

```
.container {

    width: 100%;

}

.item {

    width: 50%; /* El elemento tomará el 50% del ancho del contenedor */

    padding: 5%;

}
```

### Min, Max, y Clamp:

```
.box {

    width: max(200px, 50%); /* El ancho será el mayor entre 200px y 50% del contenedor */

    padding: min(5vw, 20px); /* El padding será el menor entre 5vw y 20px */

    font-size: clamp(1rem, 2vw, 3rem); /* La fuente será entre 1rem y 3rem, adaptándose con 2vw */

}
```



UNITED  
INNOVATE