

## 229 Preguntas de CSS para entrevistas técnicas

Escucha, me presento, por si no me conoces, soy Víctor Robles y me dedico al desarrollo web desde hace muchos años.

En este libro he recopilado muchas preguntas de CSS que, según mi experiencia, se hacen en entrevistas técnicas para puestos de desarrollo web frontend...

Y quizás leerlo antes de tu próxima entrevista puede ayudarte a superarla, quien sabe.

Incluso, si estás estudiando desarrollo web en un ciclo formativo o en la universidad te las puedes encontrar algún examen de CSS.

Vamos con ellas...

### 1. ¿Qué es un selector en CSS?

Un selector en CSS es un patrón que se utiliza para seleccionar y aplicar estilos a uno o más elementos del documento HTML.

---

### 2. ¿Cómo se define un selector de etiqueta en CSS?

Un selector de etiqueta selecciona todos los elementos de un tipo específico usando el nombre de la etiqueta: `p { color: blue; }` selecciona todos los párrafos `<p>`.

---

### 3. ¿Qué es un selector de ID en CSS y cómo se usa?

El selector de ID aplica estilos a un elemento único identificado por un atributo id. Se usa #id:

```
#miID { color: red; }
```

---

### 4. ¿Cómo se utiliza un selector de clase en CSS?

Un selector de clase selecciona elementos con un atributo class específico y se escribe con un punto:

```
.miClase { font-size: 14px; }
```

---

### 5. ¿Qué es un selector de atributo en CSS?

Un selector de atributo selecciona elementos en función de la presencia o el valor de un atributo:

```
[type="text"] { border: 1px solid black; }
```

---

### 6. ¿Cómo funciona el selector de hijo directo (>) en CSS?

El selector > aplica estilos a los elementos que son hijos directos de un padre:

```
div > p { color: green; }
```

---

## **7. ¿Qué diferencia hay entre un selector de hijo directo y un selector descendiente?**

El selector de hijo directo (>) aplica estilos solo a los hijos inmediatos, mientras que el selector descendiente (espacio) afecta a todos los descendientes sin importar la jerarquía.

---

## **8. ¿Cómo seleccionas todos los elementos de un tipo con CSS?**

Usando el selector universal \*, puedes seleccionar todos los elementos:

```
* { margin: 0; }
```

---

## **9. ¿Qué es un grupo de selectores en CSS?**

Un grupo de selectores aplica el mismo estilo a múltiples elementos separados por comas:

```
h1, h2, h3 { font-family: Arial; }
```

---

## **10. ¿Qué es la prioridad en CSS y cómo se calcula?**

La prioridad determina qué reglas de CSS se aplican cuando múltiples coinciden. Se calcula con la especificidad, donde los selectores de ID tienen más peso que clases, y estas más que etiquetas.

---

## **11. ¿Cómo se seleccionan los elementos con varios atributos?**

Se usan múltiples selectores de atributos:

```
input[type="text"][disabled] { background: gray; }
```

---

## **12. ¿Qué es un pseudo-clase en CSS?**

Una pseudo-clase selecciona elementos en un estado específico, como :hover para un elemento sobre el que pasa el cursor.

---

## **13. ¿Cómo seleccionas un elemento con varios nombres de clase?**

Usas un selector que combine las clases:

```
.clase1.clase2 { color: purple; }
```

---

## **14. ¿Qué es un pseudo-elemento en CSS?**

Un pseudo-elemento selecciona y aplica estilos a partes específicas de un elemento, como ::before y ::after.

---

## **15. ¿Qué significa la especificidad de un selector de ID?**

Un selector de ID tiene una especificidad alta y

generalmente sobrescribe estilos aplicados por clases o etiquetas, con un valor específico de 100.

---

## **16. ¿Cómo se selecciona el último hijo de un elemento con CSS?**

Usando la pseudo-clase :last-child:  
`ul li:last-child { font-weight: bold; }`

---

## **17. ¿Qué es el selector descendiente en CSS y cómo funciona?**

El selector descendiente aplica estilos a todos los elementos que están dentro de otro, sin importar el nivel:  
`div p { color: blue; }`

---

## **18. ¿Qué diferencia hay entre el selector de clase y el selector de ID?**

El selector de clase puede aplicarse a múltiples elementos, mientras que el de ID se utiliza para un único elemento por su identificador único.

---

## **19. ¿Qué es el selector de combinador adyacente en CSS?**

El selector + aplica estilos al primer elemento que sigue inmediatamente a otro:

```
h1 + p { font-size: 14px; }
```

---

## **20. ¿Cómo sobrescribir estilos aplicados por selectores con menor especificidad?**

Puedes usar la regla !important o un selector más específico para sobrescribir estilos:

```
p { color: blue !important; }
```

---

## **21. ¿Cómo se especifica una fuente en CSS?**

Se utiliza la propiedad font-family, proporcionando una lista de fuentes separadas por comas.

Ejemplo: font-family: Arial, sans-serif;

---

## **22. ¿Qué son las fuentes personalizadas en CSS3?**

Son fuentes externas que se pueden cargar en una página web usando @font-face o servicios como Google Fonts.

---

### 23. ¿Cómo se utiliza @font-face para cargar una fuente personalizada?

Se define una regla @font-face con el nombre de la fuente y su ubicación:

```
@font-face {  
  
    font-family: 'MiFuente';  
  
    src: url('mi-fuente.woff2') format('woff2');  
  
}
```

---

### 24. ¿Qué es el sistema de fuentes de respaldo en CSS?

Es una lista de fuentes alternativas especificadas en font-family que se usan si la fuente principal no está disponible.

Ejemplo: font-family: Verdana, Arial, sans-serif;

---

### 25. ¿Cómo se define el tamaño de una fuente en CSS?

El tamaño de la fuente se establece con la propiedad font-size y puede usar valores absolutos (px) o relativos (em, rem).

Ejemplo: font-size: 16px;

---

## **26. ¿Qué significa em y rem en CSS?**

em es relativo al tamaño de fuente del elemento padre, mientras que rem es relativo al tamaño de fuente raíz definido en html.

---

## **27. ¿Cómo se hace un texto en negrita con CSS?**

Se utiliza la propiedad font-weight con valores como bold o numéricos como 700.

---

Ejemplo: font-weight: bold;

---

## **28. ¿Qué es la propiedad font-style en CSS?**

La propiedad font-style define si el texto es normal, cursivo (italic) o inclinado (oblique).

---

Ejemplo: font-style: italic;

---

## **29. ¿Qué es la propiedad line-height en CSS?**

line-height controla el espacio vertical entre líneas de



texto. Puede usar valores numéricos, porcentajes o unidades.

Ejemplo: `line-height: 1.5;`

---

### **30. ¿Qué es el modelo de color RGB en CSS?**

RGB representa colores mediante combinaciones de rojo, verde y azul, con valores de 0 a 255.

Ejemplo: `color: rgb(255, 0, 0);`

---

### **31. ¿Cómo se usa RGBA en CSS?**

RGBA es una extensión de RGB que incluye un canal de opacidad (alpha) con valores entre 0 y 1.

Ejemplo: `color: rgba(0, 0, 0, 0.5);`

---

### **32. ¿Qué es el modelo de color HSL en CSS?**

HSL define colores mediante matiz (hue), saturación y luminosidad.

Ejemplo: `color: hsl(120, 100%, 50%);`

---

### 33. ¿Cómo se utiliza HSLA en CSS?

HSLA añade un canal de opacidad (alpha) al modelo HSL.

Ejemplo: `color: hsla(240, 100%, 50%, 0.7);`

---

### 34. ¿Qué son los colores por nombre en CSS?

CSS soporta nombres predefinidos de colores como red, blue, green, que representan valores específicos de color.

---

### 35. ¿Cómo se definen colores usando códigos hexadecimales en CSS?

Los códigos hexadecimales representan colores mediante combinaciones de dos dígitos para rojo, verde y azul.

Ejemplo: `color: #ff5733;`

---

### 36. ¿Qué es la propiedad background-color en CSS?

background-color define el color de fondo de un elemento.

Ejemplo: background-color: lightblue;

---

### **37. ¿Qué es la propiedad color en CSS?**

La propiedad color establece el color del texto de un elemento.

Ejemplo: color: #333;

---

### **38. ¿Cómo se aplica un degradado como fondo en CSS?**

Se utiliza la propiedad background con linear-gradient o radial-gradient.

Ejemplo: background: linear-gradient(to right, red, blue);

---

### **39. ¿Cómo se puede hacer un texto transparente en CSS?**

Utilizando colores con un canal alfa, como en RGBA o HSLA.

Ejemplo: color: rgba(0, 0, 0, 0.5);

---

#### **40. ¿Qué es el contraste de colores y por qué es importante en CSS?**

El contraste de colores se refiere a la diferencia entre el texto y el fondo. Es crucial para la accesibilidad y la legibilidad.

---

#### **41. ¿Qué es la propiedad background en CSS y cómo se utiliza?**

La propiedad background es un atajo para definir el color, imagen, posición, repetición y más del fondo de un elemento.

Ejemplo: `background: #fff url('imagen.jpg') no-repeat center;`

---

#### **42. ¿Cómo se establece un color de fondo sólido en CSS?**

Usa la propiedad background-color para definir un color sólido.

Ejemplo: `background-color: lightgray;`

---

### **43. ¿Qué es background-image en CSS?**

La propiedad background-image se utiliza para establecer una imagen como fondo.

Ejemplo: background-image: url('imagen.jpg');

---

### **44. ¿Cómo se controla la posición de un fondo en CSS?**

Usa la propiedad background-position para especificar dónde se coloca el fondo.

Ejemplo: background-position: top right;

---

### **45. ¿Qué es la propiedad background-repeat en CSS?**

Controla si el fondo se repite y en qué dirección. Valores comunes: repeat, no-repeat, repeat-x, repeat-y

---

### **46. ¿Cómo se establece un fondo fijo en CSS?**

Usa la propiedad background-attachment con el valor fixed.

Ejemplo: background-attachment: fixed;

---

#### **47. ¿Qué es la propiedad background-size en CSS?**

Define el tamaño del fondo, con valores como auto, cover, contain o medidas específicas.

Ejemplo: background-size: cover;

---

#### **48. ¿Qué significa background-clip en CSS?**

Determina hasta dónde se aplica el fondo, con valores como border-box, padding-box o content-box.

---

#### **49. ¿Cómo se establece un degradado de fondo en CSS?**

Usa background con linear-gradient o radial-gradient.

Ejemplo: background: linear-gradient(to bottom, red, blue);

---

#### **50. ¿Qué es la propiedad color en CSS y para qué se utiliza?**

La propiedad color define el color del texto en un elemento.

Ejemplo: color: #333;

---

### **51. ¿Qué hace la propiedad text-align en CSS?**

Alinea el texto dentro de un contenedor con valores como left, center, right o justify.

---

### **52. ¿Qué es la propiedad text-transform en CSS?**

Controla la capitalización del texto, con valores como uppercase, lowercase o capitalize.

---

### **53. ¿Cómo se establece el espaciado entre letras en CSS?**

Usa la propiedad letter-spacing para ajustar el espacio entre caracteres.

Ejemplo: letter-spacing: 2px;

---

### **54. ¿Qué es la propiedad word-spacing en CSS?**

Define el espacio entre palabras en un texto.

Ejemplo: word-spacing: 5px;

---

### **55. ¿Cómo se hace que un texto parpadee en CSS?**

Usa la propiedad animation junto con @keyframes para crear un efecto de parpadeo.

---

### **56. ¿Qué hace la propiedad line-height en CSS?**

Controla la altura de las líneas de texto. Puede ser un número, porcentaje o unidad.

Ejemplo: line-height: 1.5;

---

### **57. ¿Qué es la propiedad white-space en CSS?**

Controla cómo se manejan los espacios en blanco y saltos de línea.

Ejemplo: white-space: nowrap;

---

### **58. ¿Cómo se aplica una sombra al texto en CSS?**

Usa la propiedad text-shadow para añadir sombras.

Ejemplo: text-shadow: 2px 2px 5px gray;



---

### **59. ¿Qué es la propiedad overflow-wrap en CSS?**

Controla cómo se ajusta el texto para evitar desbordamientos.

Ejemplo: overflow-wrap: break-word;

---

### **60. ¿Qué hace la propiedad direction en CSS?**

Define la dirección del texto, como de izquierda a derecha (ltr) o de derecha a izquierda (rtl).

---

### **61. ¿Qué es el modelo de cajas en CSS?**

Es un concepto que describe cómo se construyen y distribuyen los elementos en CSS, incluyendo márgenes, bordes, padding y contenido.

---

### **62. ¿Qué es la propiedad margin en CSS?**

margin define el espacio exterior de una caja, separándola de otros elementos.

Ejemplo: margin: 10px;

---

### **63. ¿Qué es la propiedad padding en CSS?**

padding define el espacio interior entre el contenido de una caja y sus bordes.

Ejemplo: padding: 15px;

---

### **64. ¿Qué significa box-sizing en CSS?**

Determina cómo se calcula el tamaño total de una caja: con solo el contenido (content-box) o incluyendo bordes y padding (border-box).

---

### **65. ¿Qué hace la propiedad border en CSS?**

border establece un borde alrededor de una caja, incluyendo ancho, estilo y color.

Ejemplo: border: 2px solid red;

---

### **66. ¿Cómo se define el ancho de una caja en CSS?**

Usa la propiedad width con valores absolutos (px) o relativos (%).

Ejemplo: width: 50%;

---

### **67. ¿Cómo se define la altura de una caja en CSS?**

Se utiliza la propiedad height con valores absolutos o relativos.

Ejemplo: height: 100px;

---

### **68. ¿Qué es el margin auto y cómo funciona?**

margin: auto; centra horizontalmente una caja cuando tiene un ancho definido y el contenedor permite espacio libre.

---

### **69. ¿Qué es el colapso de márgenes en CSS?**

Ocurre cuando los márgenes verticales de dos elementos adyacentes se combinan en un único valor mayor.

---

### **70. ¿Cómo se usa float en CSS?**

float posiciona elementos hacia la izquierda o derecha del contenedor, permitiendo que otros elementos fluyan a su alrededor.

Ejemplo: float: left;

---

## **71. ¿Qué es el problema del colapso de contenedores en CSS con floats?**

Sucede cuando los elementos flotantes no expanden el contenedor padre. Se puede solucionar con overflow: hidden;

---

## **72. ¿Qué es la propiedad clear en CSS?**

clear impide que un elemento quede al lado de un elemento flotante.

Ejemplo: clear: both;

---

## **73. ¿Cómo se utiliza overflow en el modelo de cajas?**

Controla qué sucede si el contenido de una caja desborda su área.

Ejemplo: overflow: scroll;

---

#### **74. ¿Qué es el posicionamiento estático en CSS?**

El valor por defecto para todos los elementos. No afecta la posición del elemento en el flujo normal.

---

#### **75. ¿Cómo funciona position: relative en CSS?**

Posiciona un elemento relativo a su posición original en el flujo del documento.

Ejemplo: top: 10px;

---

#### **76. ¿Qué hace position: absolute en CSS?**

Saca un elemento del flujo normal y lo posiciona en relación a su contenedor con posición relativa.

Ejemplo: left: 20px;

---

#### **77. ¿Cómo funciona position: fixed en CSS?**

Posiciona un elemento en relación a la ventana del navegador, permaneciendo fijo al hacer scroll.

---

#### **78. ¿Qué es el contexto de apilamiento (stacking context) en CSS?**

Es el orden en el que se dibujan elementos en la pantalla, definido por z-index y el contexto de posicionamiento.

---

### **79. ¿Qué es z-index en CSS?**

Controla la superposición de elementos; los valores más altos aparecen por encima de los más bajos.

---

### **80. ¿Cómo usar flexbox para solucionar problemas de layout relacionados con floats?**

flexbox elimina la necesidad de usar floats para crear diseños flexibles y alineados fácilmente.

---

### **81. ¿Qué es la propiedad box-shadow en CSS?**

box-shadow aplica una sombra a los bordes de una caja. Se define con desplazamientos, difuminado, extensión y color.

Ejemplo: `box-shadow: 5px 5px 10px rgba(0, 0, 0, 0.5);`

---

### **82. ¿Cómo se crea una sombra interior con box-shadow?**

Usa el valor inset para crear sombras dentro de una caja.

Ejemplo: `box-shadow: inset 2px 2px 5px gray;`

---

### **83. ¿Qué significa el primer valor de desplazamiento en box-shadow?**

El primer valor representa el desplazamiento horizontal de la sombra.

Ejemplo: `box-shadow: 10px 0 5px black;` mueve la sombra 10px a la derecha.

---

### **84. ¿Qué representa el segundo valor en box-shadow?**

El segundo valor define el desplazamiento vertical de la sombra.

Ejemplo: `box-shadow: 0 10px 5px black;` mueve la sombra 10px hacia abajo.

---

### **85. ¿Qué hace el tercer valor en box-shadow?**

Controla el radio de desenfoque (blur) de la sombra. Un valor más alto crea bordes más suaves.

Ejemplo: `box-shadow: 0 0 10px black;`

---

### **86. ¿Qué hace el cuarto valor en box-shadow?**

Determina la extensión de la sombra; un valor positivo la expande y uno negativo la reduce.

Ejemplo: `box-shadow: 0 0 10px 5px black;`

---

### **87. ¿Cómo se añade una sombra a un texto en CSS?**

Usa la propiedad `text-shadow`, especificando desplazamientos, desenfoque y color.

Ejemplo: `text-shadow: 2px 2px 4px gray;`

---

### **88. ¿Qué ocurre si no se define un valor de desenfoque en text-shadow?**

Si omites el desenfoque, la sombra tendrá bordes nítidos.



Ejemplo: `text-shadow: 2px 2px black;`

---

## **89. ¿Qué es la propiedad `border-radius` en CSS?**

`border-radius` redondea las esquinas de un elemento.

Ejemplo: `border-radius: 10px;`

---

## **90. ¿Cómo se redondea completamente un elemento con `border-radius`?**

Aplica un valor del 50% para crear un círculo en elementos cuadrados.

Ejemplo: `border-radius: 50%;`

---

## **91. ¿Qué significa definir múltiples valores en `border-radius`?**

Permite redondear cada esquina de forma individual.

Ejemplo: `border-radius: 10px 20px 30px 40px;`

---

## 92. ¿Qué es la propiedad clip-path y cómo afecta las sombras?

clip-path define una región visible de un elemento; las sombras se aplican solo dentro de esa región.

---

## 93. ¿Cómo se combina box-shadow con un fondo transparente?

Se usa un color con opacidad en la sombra.

Ejemplo: `box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);`

---

## 94. ¿Qué son las sombras múltiples en box-shadow?

Se pueden definir múltiples sombras separadas por comas.

Ejemplo: `box-shadow: 5px 5px 5px gray, -5px -5px 5px white;`

---

## 95. ¿Cómo se aplica border-radius a una sola esquina?

Usa propiedades específicas como border-top-left-radius.

Ejemplo: `border-top-left-radius: 10px;`

---

## **96. ¿Qué es el filter: drop-shadow y cómo se diferencia de box-shadow?**

drop-shadow aplica una sombra a un elemento y respeta su forma, incluyendo áreas transparentes.

Ejemplo: `filter: drop-shadow(2px 2px 5px black);`

---

## **97. ¿Qué pasa si se usa un valor negativo en el desenfoque de box-shadow?**

No es permitido; el desenfoque siempre debe ser un valor positivo o cero.

---

## **98. ¿Cómo afecta el color de fondo al efecto de sombra?**

El contraste entre el color de fondo y la sombra afecta su visibilidad y percepción.

---

## **99. ¿Qué es el border-image y cómo interactúa con border-radius?**

border-image aplica imágenes como bordes, y border-radius puede redondearlas si tienen esquinas.

---

## **100. ¿Cómo se crea un efecto de botón con sombras y bordes redondeados?**

Combina box-shadow para profundidad y border-radius para suavizar esquinas.

Ejemplo:

```
button {  
  
    border-radius: 8px;  
  
    box-shadow: 2px 2px 5px rgba(0, 0, 0, 0.3);  
  
}
```

---

## **101. ¿Qué es el posicionamiento relativo (position: relative) en CSS?**

position: relative posiciona un elemento en relación a su posición normal en el flujo del documento, permitiendo ajustes con propiedades como top, right, bottom o left.

---

## **102. ¿Qué sucede si no se especifican valores de desplazamiento en position: relative?**

El elemento permanece en su posición normal, sin ningún cambio visual.

---

## **103. ¿Qué es el posicionamiento absoluto (position: absolute) en CSS?**

position: absolute posiciona un elemento en relación a su contenedor más cercano que tenga una posición relativa, absoluta o fija definida.

---

## **104. ¿Qué ocurre si no hay un contenedor posicionado cuando se usa position: absolute?**

El elemento se posiciona en relación al elemento raíz (<html>), ocupando todo el espacio disponible.

---

## **105. ¿Cómo se define el desplazamiento de un elemento con position: absolute?**

Se usan las propiedades top, right, bottom o left para especificar la distancia desde el borde del contenedor posicionado.

---

### **106. ¿Qué es el posicionamiento fijo (position: fixed) en CSS?**

position: fixed posiciona un elemento en relación a la ventana del navegador, manteniéndose fijo incluso al hacer scroll.

---

### **107. ¿Qué pasa con un elemento con position: fixed cuando se cambia el tamaño de la ventana?**

El elemento permanece fijo en su posición definida en relación a la ventana, adaptándose a los cambios de tamaño.

---

### **108. ¿Cuál es la diferencia entre position: absolute y position: fixed?**

absolute se posiciona en relación a un contenedor posicionado, mientras que fixed lo hace en relación a la ventana del navegador.

---

### **109. ¿Qué es el overflow en CSS?**

overflow controla cómo manejar el contenido que excede el tamaño de un contenedor.

Ejemplo: overflow: hidden;

---

### **110. ¿Cuáles son los valores posibles para overflow?**

Los valores principales son visible (por defecto), hidden, scroll y auto.

---

### **111. ¿Qué hace overflow: hidden?**

Oculto cualquier contenido que exceda los límites del contenedor.

---

### **112. ¿Qué hace overflow: scroll?**

Añade barras de desplazamiento para permitir la navegación del contenido, incluso si no es necesario.

---

### **113. ¿Qué hace overflow: auto?**

Añade barras de desplazamiento solo si el contenido excede los límites del contenedor.

---

### **114. ¿Qué propiedad se usa para controlar el desbordamiento en solo una dirección?**

Usa overflow-x para el eje horizontal y overflow-y para el eje vertical.

---

### **115. ¿Qué es el contexto de apilamiento en el posicionamiento absoluto y fijo?**

Se refiere al orden visual de los elementos, controlado por z-index y el tipo de posicionamiento definido.

---

### **116. ¿Cómo afecta position: relative a los elementos hijos posicionados absolutamente?**

Define un nuevo contenedor de referencia para cualquier elemento hijo con position: absolute.

---

### **117. ¿Cómo centrar un elemento en la pantalla usando position: absolute?**

Define las propiedades top, left, transform, y usa un contenedor posicionado:

```
div {  
  
    position: absolute;  
  
    top: 50%;  
  
    left: 50%;  
  
    transform: translate(-50%, -50%);  
  
}
```

---



### **118. ¿Qué es el clipping de contenido en elementos con overflow: hidden?**

El contenido que excede los límites del contenedor no se muestra, creando un "recorte".

---

### **119. ¿Cómo funciona el desplazamiento con overflow: scroll y overflow: auto en dispositivos móviles?**

En dispositivos táctiles, permite deslizar el contenido con gestos táctiles, pero scroll muestra siempre barras, mientras que auto las oculta si no son necesarias.

---

### **120. ¿Qué significa establecer position: static en un elemento?**

position: static es el valor por defecto en CSS y posiciona el elemento según el flujo normal del documento, sin posibilidad de usar propiedades como top o left.

---

### **121. ¿Qué es una pseudoclase en CSS?**

Una pseudoclase define un estado especial de un elemento.

Ejemplo: :hover aplica estilos cuando el puntero está sobre un elemento.

---

## 122. ¿Qué es un pseudoelemento en CSS?

Un pseudoelemento permite estilizar partes específicas de un elemento, como `::before` o `::after`.

---

## 123. ¿Cuál es la diferencia entre pseudoclases y pseudoelementos?

Las pseudoclases se refieren a estados de un elemento, mientras que los pseudoelementos crean subpartes estilizadas de un elemento.

---

## 124. ¿Qué hace la pseudoclase `:hover`?

Aplica estilos a un elemento cuando el usuario pasa el puntero sobre él.

---

## 125. ¿Cómo funciona la pseudoclase `:focus`?

Aplica estilos a un elemento cuando obtiene el foco, como un campo de entrada al seleccionarlo.

---

## 126. ¿Qué es `::before` y cómo se utiliza?

`::before` inserta contenido antes del contenido real de un elemento.

Ejemplo:

```
p::before {  
  
  content: "Nota: ";  
  
}
```

---

## 127. ¿Qué hace ::after en CSS?

::after inserta contenido después del contenido de un elemento.

```
div::after {  
  
  content: "*";  
  
}
```

---

## 128. ¿Qué es la pseudoclase :nth-child(n)?

Selecciona el enésimo hijo de un elemento padre, donde n puede ser un número o una fórmula.

---

### **129. ¿Qué hace la pseudoclase :not(selector)?**

Selecciona todos los elementos que no coincidan con el selector especificado.

Ejemplo: `div:not(.active)`

---

### **130. ¿Qué es una transición en CSS?**

Las transiciones permiten animar cambios en las propiedades de estilo de un elemento durante un periodo de tiempo definido.

---

### **131. ¿Cuáles son las propiedades clave de una transición en CSS?**

transition-property (propiedad a animar), transition-duration (duración), transition-timing-function (aceleración) y transition-delay (retraso).

---

### **132. ¿Qué hace la propiedad transition-timing-function?**

Define la velocidad de cambio durante la transición, con valores como ease, linear, ease-in o ease-out.

---

### 133. ¿Cómo se aplican transiciones a múltiples propiedades en CSS?

Se enumeran las propiedades separadas por comas.

Ejemplo: `transition: color 0.3s, background-color 0.5s;`

---

### 134. ¿Qué es una animación en CSS?

Las animaciones permiten crear movimientos complejos mediante la propiedad `@keyframes` y `animation`.

---

### 135. ¿Qué es `@keyframes` en CSS?

`@keyframes` define el comportamiento de una animación, especificando estilos en diferentes momentos.

Ejemplo:

```
@keyframes fade {  
  
  from { opacity: 0; }  
  
  to { opacity: 1; }  
  
}
```

---

### **136. ¿Cuáles son las propiedades principales de animation en CSS?**

Incluyen animation-name, animation-duration, animation-timing-function, animation-delay, animation-iteration-count y animation-direction.

---

### **137. ¿Qué hace animation-iteration-count?**

Define cuántas veces se repite una animación. Puede ser un número o infinite.

---

### **138. ¿Cómo se hace que una animación vaya y vuelva?**

Se usa animation-direction: alternate; para alternar el sentido de la animación.

---

### **139. ¿Qué diferencia hay entre animation y transition en CSS?**

animation permite movimientos complejos definidos por @keyframes, mientras que transition se limita a animar cambios en propiedades específicas.

---

### **140. ¿Qué es la pseudoclase :checked y para qué se usa?**

Aplica estilos a elementos seleccionados, como un checkbox o un radio button.

---

### **141. ¿Qué es el diseño web adaptable o Responsive Web Design (RWD)?**

Es una técnica de diseño que permite que un sitio web se adapte a diferentes tamaños de pantalla y dispositivos, proporcionando una experiencia óptima al usuario.

---

### **142. ¿Qué es una unidad relativa en CSS y para qué se utiliza?**

Las unidades relativas, como %, em, o rem, se ajustan en función de elementos padres o raíces, lo que facilita un diseño adaptable.

---

### **143. ¿Cuál es la diferencia entre rem y em en CSS?**

rem se basa en el tamaño de la fuente raíz del documento, mientras que em depende del tamaño de la fuente del elemento padre.

---

### **144. ¿Cómo se usan porcentajes en CSS para un diseño adaptable?**

Los porcentajes se utilizan para definir dimensiones relativas al tamaño del contenedor, como en `width: 50%`;

---

### **145. ¿Qué son las media queries en CSS?**

Son reglas que aplican estilos según las características del dispositivo, como ancho de pantalla o resolución.

---

### **146. ¿Cómo se escribe una media query básica para pantallas pequeñas?**

```
@media (max-width: 768px) {  
  
  body {  
  
    font-size: 14px;  
  
  }  
  
}
```

---

### **147. ¿Qué hace el atributo viewport en HTML?**

El meta viewport controla cómo se escala y muestra una página en dispositivos móviles.

Ejemplo:



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

---

### **148. ¿Qué significa initial-scale=1.0 en el meta viewport?**

Establece el nivel de zoom inicial de la página para que se ajuste al ancho del dispositivo.

---

### **149. ¿Qué es un diseño fluido en CSS?**

Es un diseño basado en porcentajes o unidades relativas, en lugar de píxeles fijos, para adaptarse a diferentes tamaños de pantalla.

---

### **150. ¿Cómo se utiliza min-width en una media query?**

min-width aplica estilos cuando el ancho del dispositivo es mayor o igual al valor especificado.

Ejemplo: @media (min-width: 1024px) {}

---

### **151. ¿Qué es el diseño mobile-first?**

Es un enfoque de diseño donde se comienza optimizando para dispositivos móviles, añadiendo estilos para pantallas más grandes con media queries.

---

### **152. ¿Cómo combinar múltiples condiciones en una media query?**

Se usan operadores como and, or o not.

Ejemplo:

```
@media (min-width: 768px) and (max-width: 1024px) {}
```

---

### **153. ¿Qué es el grid layout y cómo mejora el diseño Responsive?**

CSS Grid Layout es un sistema de diseño bidimensional que organiza elementos en filas y columnas, facilitando el diseño adaptable.

---

### **154. ¿Cómo se adapta un diseño basado en flexbox a pantallas pequeñas?**

Usando propiedades como flex-wrap y media queries para ajustar el comportamiento de los elementos en filas o columnas.

---

### **155. ¿Qué son los breakpoints en diseño web adaptable?**

Son puntos de corte en las media queries donde el

diseño cambia para ajustarse a diferentes tamaños de pantalla.

---

### **156. ¿Qué significa max-width: 100% y cuándo se utiliza?**

Limita el ancho máximo de un elemento al 100% del contenedor, evitando desbordamientos en diseños adaptables.

---

### **157. ¿Qué es un contenedor fluido en diseño Responsive?**

Un contenedor que utiliza unidades relativas, como %, para adaptarse al ancho del dispositivo.

---

### **158. ¿Qué ventaja tiene usar clamp() en propiedades CSS adaptables?**

Permite definir un valor mínimo, preferido y máximo para propiedades como font-size, ajustándose automáticamente.

---

### **159. ¿Qué problema soluciona el uso de imágenes responsivas en CSS?**

Asegura que las imágenes se adapten correctamente al

ancho de su contenedor, mejorando la experiencia en dispositivos móviles.

Ejemplo: `img { max-width: 100%; height: auto; }`

---

## **160. ¿Qué es el diseño basado en columnas y cómo se aplica en CSS?**

Es un enfoque que organiza contenido en una estructura de columnas, a menudo usando frameworks o CSS Grid para facilitar el diseño adaptable.

---

## **161. ¿Qué es Flexbox en CSS y para qué se utiliza?**

Flexbox es un modelo de diseño unidimensional que organiza elementos en filas o columnas, facilitando el alineamiento y distribución del espacio.

---

## **162. ¿Qué propiedad se utiliza para activar Flexbox en un contenedor?**

La propiedad `display: flex;` activa Flexbox en un elemento contenedor.

---

### **163. ¿Cuál es la diferencia entre flex-start y flex-end en CSS Flexbox?**

flex-start alinea los elementos al inicio del contenedor, mientras que flex-end los alinea al final.

---

### **164. ¿Qué hace la propiedad justify-content en Flexbox?**

Controla la alineación horizontal de los elementos dentro del contenedor. Valores comunes: flex-start, center, space-between.

---

### **165. ¿Qué función tiene la propiedad align-items en Flexbox?**

Alinea los elementos hijos a lo largo del eje transversal (por defecto, vertical). Valores: stretch, center, flex-start, flex-end.

---

### **166. ¿Qué hace la propiedad align-content en Flexbox?**

Alinea varias líneas de contenido cuando hay espacio adicional en el eje transversal.

---

### **167. ¿Qué es el eje principal y el eje transversal en Flexbox?**

El eje principal define la dirección de los elementos (fila o columna), mientras que el eje transversal es perpendicular al principal.

---

### **168. ¿Cómo se cambia la dirección del eje principal en Flexbox?**

Con la propiedad flex-direction. Valores: row, row-reverse, column, column-reverse.

---

### **169. ¿Qué hace la propiedad flex-wrap en Flexbox?**

Permite que los elementos se ajusten en múltiples líneas si no caben en una sola. Valores: nowrap, wrap, wrap-reverse.

---

### **170. ¿Cómo se usa la propiedad order en Flexbox?**

Cambia el orden visual de los elementos, independientemente de su posición en el HTML.

---

### **171. ¿Qué hace la propiedad flex en un elemento hijo de Flexbox?**

Define el crecimiento, contracción y tamaño base del elemento. Ejemplo: flex: 1;

---

### **172. ¿Cuál es la diferencia entre flex-grow, flex-shrink y flex-basis?**

- flex-grow: Controla cómo crece un elemento si hay espacio adicional.
  - flex-shrink: Controla cómo se reduce un elemento si falta espacio.
  - flex-basis: Establece el tamaño base inicial de un elemento.
- 

### **173. ¿Qué hace la propiedad gap en Flexbox?**

Define el espacio entre elementos dentro del contenedor flex.

---

### **174. ¿Qué significa justify-content: space-between en Flexbox?**

Distribuye los elementos con espacio igual entre ellos, dejando los extremos alineados al borde del contenedor.

---

### **175. ¿Qué valor de align-items estira los elementos en el eje transversal?**

El valor stretch estira los elementos para que ocupen todo el espacio disponible en el eje transversal.

---

### **176. ¿Qué es justify-content: space-around y cómo afecta a los elementos?**

Distribuye los elementos con espacios iguales a ambos lados, dejando más espacio en los bordes exteriores.

---

### **177. ¿Qué sucede si flex-wrap está en nowrap y los elementos no caben en una línea?**

Los elementos se mantendrán en una sola línea, causando desbordamiento del contenedor.

---

### **178. ¿Cómo se usa align-self en Flexbox y qué hace?**

Permite alinear un elemento hijo de manera individual, sobrescribiendo el valor de align-items del contenedor.

---

### **179. ¿Qué pasa si usas flex: auto; en un elemento hijo?**

El elemento crece o se encoge según el espacio



disponible, combinando flex-grow: 1, flex-shrink: 1 y flex-basis: auto.

---

## **180. ¿Cómo se centra un elemento horizontal y verticalmente con Flexbox?**

Usando:

display: flex;

justify-content: center;

align-items: center;

---

## **181. ¿Qué es CSS Grid Layout?**

CSS Grid Layout es un sistema de diseño bidimensional que permite organizar los elementos en filas y columnas, facilitando la creación de layouts complejos.

---

## **182. ¿Cómo se define un contenedor Grid en CSS?**

Se define con display: grid; en el contenedor.

---

## **183. ¿Qué son las "grid areas" en CSS Grid?**

Las "grid areas" permiten definir áreas específicas

dentro de una cuadrícula, asignando nombres a ciertas zonas del grid.

---

## **184. ¿Cómo se crea un grid básico con CSS?**

Usando `grid-template-columns` y `grid-template-rows` para definir el número y tamaño de las filas y columnas.

Ejemplo:

```
.container {  
  
    display: grid;  
  
    grid-template-columns: 1fr 1fr;  
  
    grid-template-rows: auto;  
  
}
```

---

## **185. ¿Qué hace la propiedad `grid-template-columns`?**

Define el número y tamaño de las columnas en un grid.

Ejemplo: `grid-template-columns: 1fr 2fr;`

---

## **186. ¿Qué hace la propiedad `grid-template-rows`?**

Define el número y tamaño de las filas en un grid.

Ejemplo: `grid-template-rows: 100px 200px;`

---

### 187. ¿Qué son las unidades fr en CSS Grid?

La unidad fr (fracción) representa una parte del espacio disponible en el contenedor. Ejemplo: `grid-template-columns: 1fr 2fr;` distribuye el espacio disponible en una proporción de 1:2.

---

### 188. ¿Cómo se definen las "grid areas" en CSS?

Usando la propiedad `grid-template-areas`, se asignan nombres a las zonas del grid. Ejemplo:

```
.container {  
  
    display: grid;  
  
    grid-template-areas: "header header"  
                        "main aside"  
                        "footer footer";  
  
}
```

---

### 189. ¿Cómo se asigna un elemento a una "grid area"?

Usando la propiedad `grid-area` en el elemento, indicando el nombre de la área definida previamente.

Ejemplo:

```
.header { grid-area: header; }
```

---

### **190. ¿Qué significa grid-template-areas: "header header" "main aside" "footer footer";?**

Define una cuadrícula con tres filas y dos columnas. La primera fila tiene dos columnas "header", la segunda tiene "main" y "aside", y la tercera tiene dos columnas "footer".

---

### **191. ¿Qué hace la propiedad grid-gap en CSS Grid?**

Define el espacio entre las filas y columnas del grid. También puede ser separada en grid-row-gap y grid-column-gap.

---

### **192. ¿Cómo se usan las líneas del grid con grid-column y grid-row?**

grid-column y grid-row se usan para definir en qué línea de la cuadrícula comienza y termina un elemento.

Ejemplo: grid-column: 1 / 3;

---

### **193. ¿Cómo se utiliza grid-column-start y grid-column-end?**

grid-column-start define la línea inicial y grid-column-end la línea final del área de un elemento. Ejemplo: grid-column-start: 1; grid-column-end: 3;

---

### **194. ¿Cómo se utiliza grid-row-start y grid-row-end?**

De manera similar a grid-column, grid-row-start y grid-row-end definen las líneas de inicio y fin en las filas. Ejemplo: grid-row-start: 1; grid-row-end: 2;

---

### **195. ¿Qué es un "item" en CSS Grid?**

Un "item" es un elemento hijo dentro de un contenedor con display: grid; aplicado, que se distribuye en el grid.

---

### **196. ¿Cómo se alinean los elementos dentro de un grid?**

Usando propiedades como justify-items, align-items, justify-self y align-self para controlar la alineación de los elementos dentro de sus celdas.

---

### **197. ¿Qué hace la propiedad justify-items en CSS Grid?**

Controla la alineación horizontal de los elementos dentro de sus celdas en el grid.

---

### **198. ¿Cómo se alinean todos los elementos en el eje transversal de un grid?**

Usando la propiedad align-items para definir la alineación vertical de todos los elementos dentro del contenedor grid.

---

### **199. ¿Qué diferencia hay entre justify-items y justify-content en CSS Grid?**

justify-items alinea los elementos dentro de sus celdas, mientras que justify-content alinea las filas o columnas dentro del contenedor grid.

---

### **200. ¿Qué es grid-auto-rows y cómo se utiliza?**

Es una propiedad que define el tamaño de las filas que no están explícitamente definidas en el grid, ajustando su tamaño automáticamente.

---

### **201. ¿Qué son las unidades de medida absolutas en CSS?**

Son unidades que no dependen de las características del entorno de visualización, como px, cm, mm, in, etc.

---

## **202. ¿Qué son las unidades de medida relativas en CSS?**

Son unidades que se ajustan en función del entorno, como em, rem, %, vw, vh, entre otras.

---

## **203. ¿Qué significa la unidad rem en CSS?**

rem (root em) es una unidad relativa que se refiere al tamaño de fuente raíz del documento, generalmente html.

---

## **204. ¿Qué es la unidad em en CSS?**

em es una unidad relativa al tamaño de la fuente del elemento actual. Si el tamaño de la fuente del elemento es 16px, 1em será 16px.

---

## **205. ¿Cuál es la diferencia entre rem y em?**

rem se basa en el tamaño de la fuente raíz, mientras que em se basa en el tamaño de la fuente del elemento actual.

---

### **206. ¿Qué es la unidad vh en CSS?**

vh (viewport height) es una unidad relativa al 1% de la altura del área de visualización del navegador.

---

### **207. ¿Qué es la unidad vw en CSS?**

vw (viewport width) es una unidad relativa al 1% de la anchura del área de visualización del navegador.

---

### **208. ¿Cómo se usa la unidad % en CSS?**

El porcentaje se usa en relación con el elemento contenedor. Por ejemplo, width: 50% significa que el ancho será el 50% del contenedor.

---

### **209. ¿Qué son las variables en CSS?**

Las variables en CSS permiten almacenar valores reutilizables, como colores o tamaños, que pueden ser referenciados en múltiples lugares del CSS.

---

### **210. ¿Cómo se declara una variable en CSS?**

Se declara con la sintaxis --nombre-variable: valor;.

Ejemplo:



```
:root {  
  
  --color-principal: #3498db;  
  
}
```

---

### **211. ¿Cómo se utiliza una variable en CSS?**

Se usa con la función `var()`, de la siguiente manera: `color: var(--color-principal);`

---

### **212. ¿Qué es la propiedad `box-sizing` en CSS?**

`box-sizing` controla cómo se calcula el ancho y la altura de los elementos. Con `box-sizing: border-box;`, el padding y el borde se incluyen en el cálculo del tamaño.

---

### **213. ¿Cuál es el valor predeterminado de `box-sizing` en CSS?**

El valor predeterminado es `content-box`, donde el padding y el borde se añaden al tamaño total del elemento.

---

### **214. ¿Qué significa `box-sizing: border-box;`?**

Significa que el tamaño de un elemento incluye el borde

y el padding, por lo que no aumenta el tamaño total del elemento.

---

### **215. ¿Cómo afecta box-sizing: border-box al diseño de una página?**

Hace que el cálculo del ancho y alto de los elementos sea más predecible y fácil de manejar, ya que no se tiene que considerar el borde y el padding por separado.

---

### **216. ¿Qué es clip-path en CSS?**

clip-path permite recortar un elemento en una forma específica, como un círculo o un polígono.

---

### **217. ¿Cómo se utiliza clip-path para crear una forma circular?**

Con clip-path: circle(50%); se recorta un elemento en forma de círculo.

---

### **218. ¿Qué es clip-path: polygon() en CSS?**

polygon() permite definir una forma personalizada mediante coordenadas, especificando puntos para crear una figura compleja.

---

### **219. ¿Cómo recortar un elemento en forma de triángulo usando clip-path?**

Con clip-path: polygon(50% 0%, 0% 100%, 100% 100%);, creando un triángulo con coordenadas.

---

### **220. ¿Qué es el método BEM en CSS?**

BEM (Block Element Modifier) es un enfoque de nomenclatura para estructurar clases CSS de manera modular, clara y reutilizable.

---

### **221. ¿Qué significa "Bloque" en el contexto de BEM?**

Un "Bloque" es una entidad independiente que representa una sección de la interfaz, como un botón o un formulario.

---

### **222. ¿Qué es un "Elemento" en el contexto de BEM?**

Un "Elemento" es una parte de un bloque, que depende de él para tener sentido.

Ejemplo: el texto dentro de un botón.

---

### **223. ¿Qué es un "Modificador" en BEM?**

Un "Modificador" define variaciones de un bloque o elemento, como su estado o apariencia, utilizando un sufijo en el nombre de la clase.

Ejemplo: `.button--active`

---

### **224. ¿Cómo se define un bloque en BEM?**

Se define usando un nombre único para la clase, por ejemplo, `.header`, `.button`, etc.

---

### **225. ¿Cómo se define un elemento dentro de un bloque en BEM?**

Se define usando un doble guion en el nombre de la clase, como `.button__icon`, donde `button` es el bloque y `icon` es el elemento.

---

### **226. ¿Cómo se define un modificador en BEM?**

Se define agregando un doble guion al nombre del bloque o elemento, como `.button--primary`, donde `primary` es una variante del botón.

---

### **227. ¿Por qué es importante usar BEM en CSS?**

BEM ayuda a mantener el código limpio, modular y fácil de mantener, evitando conflictos de nombres y favoreciendo la reutilización de estilos.

---

### **228. ¿Cómo organizarías un archivo CSS siguiendo el método BEM?**

Organizando las clases por bloques, elementos y modificadores, manteniendo una estructura coherente y consistente en los nombres de clases.

---

### **229. ¿Cuál es una ventaja principal de usar clip-path en lugar de otras técnicas como border-radius?**

clip-path ofrece una mayor flexibilidad para crear formas complejas y personalizadas, mientras que border-radius solo es útil para bordes redondeados.

Y esto es todo.

Si quieres aprender CSS de forma sencilla y divertida, tengo 3 cursos de 3 niveles diferentes que puedes hacer.

Nivel 1, desde cero a intermedio:

**<https://victorroblesweb.es/master-css>**

Nivel 2, de intermedio a avanzado:

**<https://victorroblesweb.es/master-css-avanzado>**

Nivel 3, de avanzado a extremo:

**<https://victorroblesweb.es/master-css-extremo>**

Y si quieres aprender desarrollo web desde cero hasta estar preparado para trabajar, al menos como junior, te recomiendo que sigas esta ruta de aprendizaje y programes mucho:

**<https://victorroblesweb.es/ruta>**

Gracias,

Nos vemos y nos leemos

Atentamente, Víctor Robles WEB ;)