

Undefined Academy

Bootcamp de JavaScript Full-stack

Semana 1, Clase 4:

Introducción a CSS



Metodología de trabajo

Formato de las sesiones (1h 30m)

- 15 min Revisión de ejercicio anterior
- 60 min Presentación teorica-práctica
- 15 min Preguntas y discusión

Herramientas del Bootcamp

- **Twitch:** <https://undf.sh/en-vivo>
- **Discord:** <https://undf.sh/discord>
- **Notion:** <https://undf.sh/base>

Agenda

- Las hojas de estilos, propiedades y unidades en CSS
- Modelo de caja en CSS
- Cómo funcionan los selectores en CSS
- Práctica de selectores en CSS
- Cómo funciona el layout en CSS
- Cascada y especificidad en CSS

Revisión de ejercicio anterior



Las hojas de
estilos,
propiedades y
unidades en CSS

¿Qué es el CSS?

Las **hojas de estilo en cascada**¹ son un *lenguaje de estilo* utilizado para describir la presentación de un documento HTML.

Incluyendo aspectos como el diseño, los colores, las fuentes, los tamaños, etc.

1. Cascading Style Sheets.

Regla de CSS

```
<style>
  body {
    background: tomato;
    font-size: 16px;
  }
</style>
```

```
<style>
  body {
    background: tomato;
    font-size: 16px;
  }
</style>
```

- Selector `body`
- Propiedades: `background`, `font-size`
- Valores: `tomato`, `16px`
- Unidad: `px`

Enlazando una hoja de estilos

```
<!-- Adentro de la etiqueta head -->  
<link rel='stylesheet' href='main.css' />
```

Modelo de caja en CSS

Elementos bloque y elementos en línea

The background of the entire image is a textured pattern of irregular, light brown and tan stones, resembling a stone wall or cobblestone pavement. The stones are of various shapes and sizes, creating a mosaic-like effect.

Bloque

Bloque

Bloque



Bloque

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

En Línea

Bloque

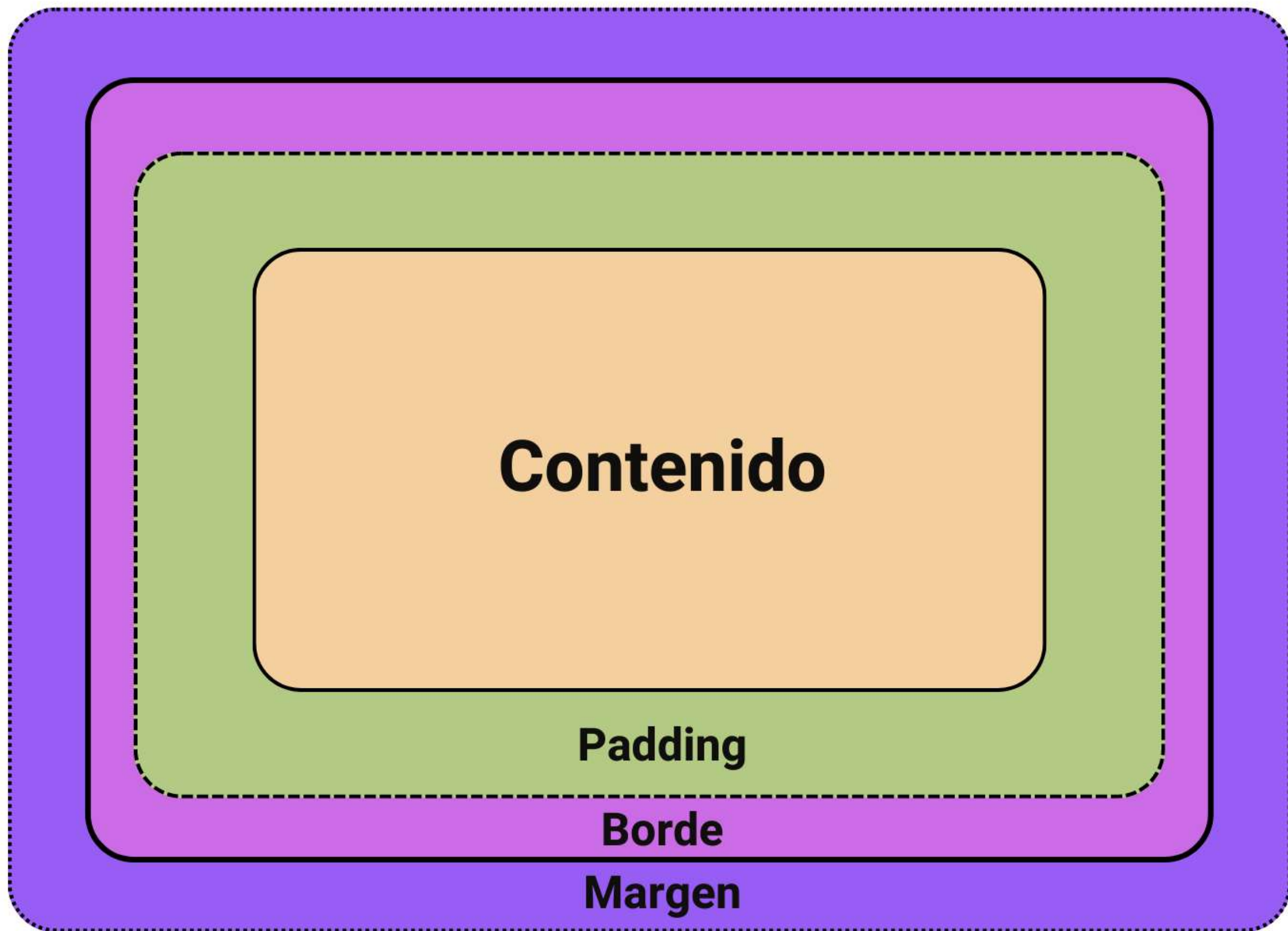
Los encabezados son tipo bloque `<h1>`

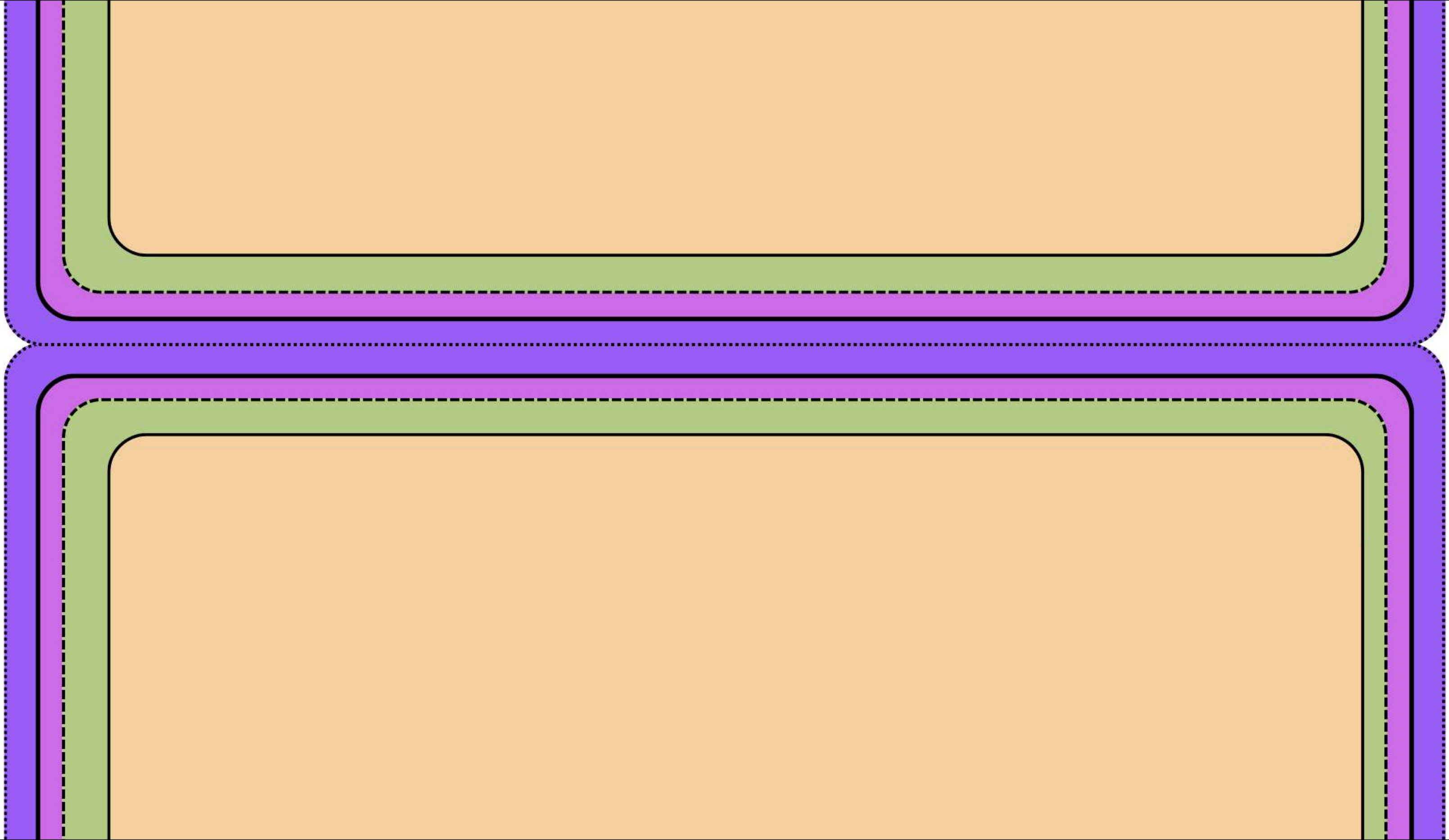
Los párrafos `<p>` también son tipo bloque, sin embargo los elementos de *énfasis en cursiva* `` y **énfasis en negrilla** `` son elemento en línea.

- Los **elementos de bloque** siempre aparecen debajo del elemento de bloque anterior.
- El ancho de los **elementos de bloque** se establece automáticamente en función del ancho de su contenedor principal.
- La altura predeterminada de los **elementos de bloque** se basa en el contenido que contiene.
- Los **elementos en línea** no afectan el espacio vertical.
- El ancho de los **elementos en línea** se basa en el contenido que contienen, no en el ancho del elemento principal.

Contenido, Padding, Borde y Margen

- **Contenido:** es el texto, las imágenes o cualquier otro contenido multimedia.
- **Padding:** es el espacio entre el contenido de la caja y su borde.
- **Borde:** es la línea entre el relleno y el margen del cuadro.
- **Margen:** el espacio entre la caja y las cajas alrededor de esta.





```
section {  
    background: MediumPurple;  
}  
  
div {  
    background: PaleGoldenRod;  
    padding: 16px 32px;  
    margin: 20px;  
    border: 8px solid Orchid;  
  
}  
  
p {  
    background: PapayaWhip;  
    color: LightGray;  
}
```

Formatos cortos

```
div {  
  padding-top: 8px;  
  padding-right: 16px;  
  padding-bottom: 8px;  
  padding-left: 16px;  
}
```

```
div {  
  padding: 8px; /* Se aplica a todos los lados */  
}
```

```
div {  
  padding: 8px 16px; /* Vertical, Horizontal */  
}
```

```
div {  
  padding: 8px 16px 0; /* Arriba, Horizontal, Abajo */  
}
```

¿Cómo se diferencia esto de las Logical Properties?¹

1. <https://web.dev/learn/css/logical-properties/>

Otras consideraciones del modelo de caja

- Margen en elementos en línea¹
- Colapso vertical del margen
- Dimensiones explícitas
- Tamaño de la caja `box-sizing`

1. Su alineación vertical no se ve afectada.

Cómo funcionan los selectores en CSS

Cómo funcionan los selectores en CSS

```
<p class="callout">Este parrafo tiene una  
clase, un posible estilo compartido.</p>
```

```
<p id="main">Este parrafo tiene un  
identificador, qué debería ser único en  
todo el documento</p>
```

Cómo funcionan los selectores en CSS

- `tipo`
- `.selector-de-clase`
- `#selector-de-id`
- `[atributo]`
- `*`

Combinaciones de selectores

- Adyacente `.primero + .segundo`
- Hijo `.prade > .hijo`
- Descendiente `.abuelo .bisnieto`
- Hermano `.hermano ~ .hermana`

Práctica de selectores en CSS

1. Pintar a Paco y Luis con **Tomato**
2. Pintar al Nieto con **RebeccaPurple**
3. Darle de comer a los Nietos y Bisnietos **Bold**
4. Pintar al Sobrino **Teal**

```
<div class="abuelo">
  <div class="padre">
    <p>Hugo</p>
    <p>Paco</p>
    <p>Luis</p>
    <div>
      <span>Bisnieto</span>
    </div>
  <p>Primo</p>
  <div>Sobrino</div>
</div>
<div class="hermano">
  <span>
    Nieto
  </span>
</div>
</div>
```


Cómo funciona el layout en CSS

Cómo funciona el layout en CSS

```
<style>
main {
  width: 600px;
}
div {
  padding: 20px;
  border: 1px solid tomato;
}
</style>
<main>
  <div class="header">Encabezado</div>
  <div class="column">Uno</div>
  <div class="column">Dos</div>
  <div class="column">Tres</div>
  <div class="footer">Pie de Página</div>
</main>
```

🍏 ¿Cómo se podría solucionar el ejercicio anterior usando `float`?

Cascada y especificidad en CSS

Cascada y especificidad en CSS

1. **Posición y orden:** el orden en que se declaran las reglas CSS.
2. **Especificidad:** un algoritmo que determina qué selector de CSS tiene más peso.
3. **Origen:** el orden en que aparece CSS y de dónde proviene, ya sea un estilo de navegador, el CSS de una extensión o nuestro CSS.
4. **Importancia:** algunas reglas CSS tienen más peso que otras, especialmente con el tipo de regla `!important`

Posición y orden

```
button {  
  color: blue;  
}
```

```
button {  
  color: yellow;  
}
```

Especificidad¹

```
header h1#title > .logo.rounded
```

1 id, 2 clases, 2 elementos

1. <https://polypane.app/css-specificity-calculator>

Origen

1. **User agent base styles:** Estos son los estilos que su navegador aplica a los elementos HTML por defecto.
2. **Local user styles:** Estos pueden provenir del nivel del sistema operativo, como un tamaño de fuente base o una preferencia de movimiento reducido. También pueden provenir de extensiones de navegador.
3. **Authored CSS:** El CSS que creas.
4. **Authored !important:** Cualquier `!important` que agregue a sus declaraciones de autor.
5. **Local user styles !important:** Cualquier `!important` que provenga del nivel del sistema operativo o del nivel de extensión del navegador CSS.
6. **User agent !important:** Cualquier `!important` que esté definido en el CSS predeterminado, proporcionado por el navegador.

Importancia

1. Reglas normales, como `font-size`,
`background` o `color`
2. Reglas de animación
3. Reglas !important (siguiendo el mismo orden que el origen)
4. Reglas de transición

Recursos

- <https://cssreference.io/>
- <https://web.dev/learn/css/>
- <https://laboratoria.github.io/css-diner/>

Ejercicio

1. Revisar el tutorial de "Diseño Web en 4 minutos"¹ de Jeremy Thomas.
2. Aplicar lo aprendido a tu `index.html`.
3. No tiene que ser literal, solo asegúrate de que tu archivo `index.html` se visualmente agradable.
4. Hacer **commit** y **push**.

1. <https://jgthms.com/web-design-in-4-minutes>

¿Preguntas?