

---

---

# IIC2026

## Ayudantía 1 - HTML, CSS y JS

— Sofía Hosiasson —

---

---

# HTML & CSS

---

# HTML - Estructura completa

```
<!DOCTYPE html>
<html>
  <!-- Comentario -->
  <head>
    <meta charset="utf-8">
    <title>Título de documento completo</title>
  </head>
  <body>
    <h1> Un título </h1>
    <p> Un párrafo </p>
    <h2> Un subtítulo </h2>
    <p> Otro párrafo </p>
    <ul>
      <li>Primer elemento de lista.</li>
      <li>Segundo elemento de lista.</li>
      <li>Tercer elemento de lista.</li>
    </ul>
  </body>
</html>
```

▼ Título de documento completo × +

← → ↻ ⓘ Archivo

## Un título

Un párrafo

## Un subtítulo

Otro párrafo

- Primer elemento de lista.
- Segundo elemento de lista.
- Tercer elemento de lista.

# CSS

## Cascading Style Sheets

External CSS

```
<link href="estilo.css" rel="stylesheet">
```

```
1
2  ✓ body{background-color: ■rosybrown;
3      font-family: monospace;
4      font-size: 20px;
5  }
6
7  ✓ #identificador{
8      color: ■red;
9  }
10
11 ✓ .clase1{
12     font-size: 30px;
13 }
14
```

# SVG

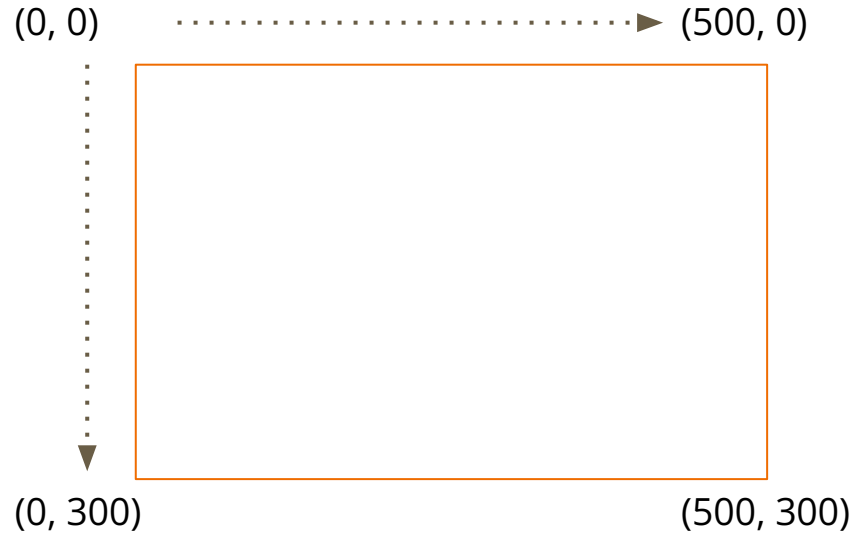
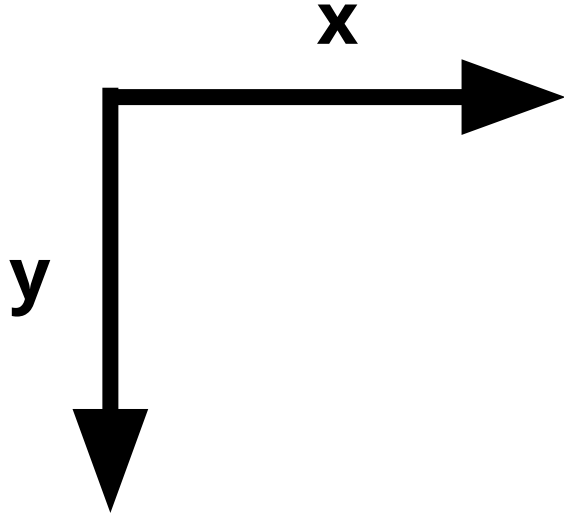
---

# SVG - Formato

```
<svg width="500" height="300">
```

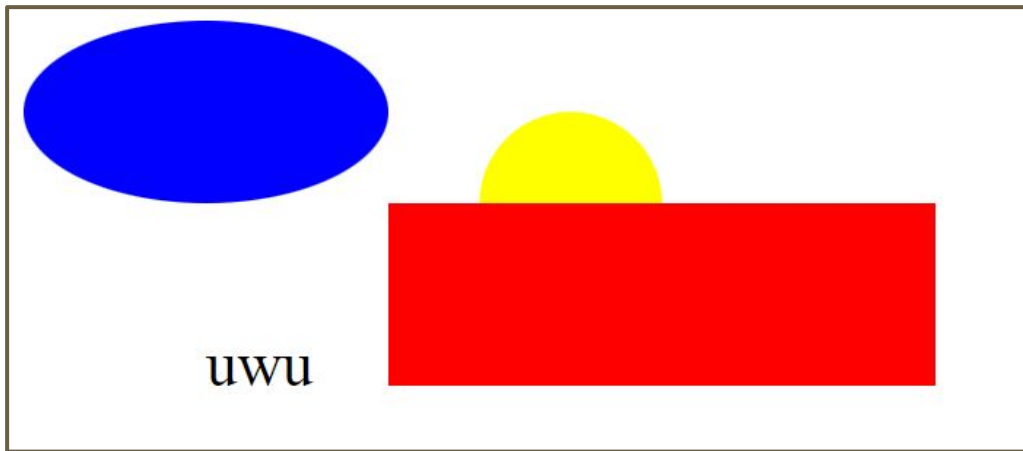
```
...
```

```
</svg>
```



# Inclur SVG en el HTML

```
<svg width="500" height="300">  
  <ellipse cx="100" cy="50" rx="100" ry="50" fill="blue" />  
  <circle cx="150" cy="100" r="50" fill="yellow" />  
  <rect x="200" y="100" width="300" height="100" fill="red" />  
  <text x="100" y="200" font-size="34" size="1">uwu </text>  
</svg>
```



# Intro a JavaScript

---



## En el HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ejemplo con JS</title>
  </head>

  <body id="raiz">
    <h1 id="principal">¡Presioname!</h1>
    <script src='programa.js' charset='utf-8'></script>
  </body>
</html>
```

**Programa  
a ejecutar**

# Declaraciones de variables y funciones en Javascript

**const** : Nos permite declarar variables inmutables, osea no se puede cambiar.

**let** : Nos permiten crear variables mutables que si se puede cambiar.

```
function NombreFuncion(a) {  
    let atributo = a;  
    ...  
    return atributo  
}
```

# CONSOLA Y ELEMENTOS DE UNA PÁGINA

## Inspeccionar (ver los elementos de una página):

- Ctrl+Shift+I para Windows
- Cmd+Shift+I para Mac


## Ver consola (lo que imprime console.log)

- Ctrl+Shift+C para Windows
- Cmd+Shift+C para Mac

# PYTHON VS JAVASCRIPT

---

## IF, ELSE IF & ELSE

 testeo.py > ...

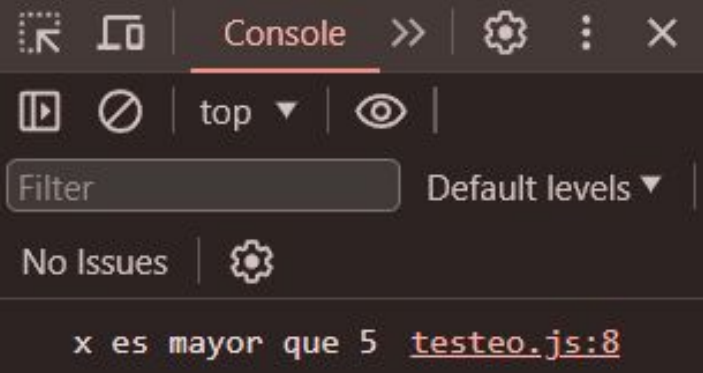
```
1  x = 10
2
3  ✓ if x < 5:
4      print("x es menor que 5")
5  ✓ elif x == 5:
6      print("x es igual a 5")
7  ✓ else:
8      print("x es mayor que 5")
```

**JS** testeo.js > ...

```
1  let x = 10;
2
3  if (x < 5) {
4      console.log("x es menor que 5");
5  } else if (x === 5) {
6      console.log("x es igual a 5");
7  } else {
8      console.log("x es mayor que 5");
9  }
```

**TERMINAL**

x es mayor que 5



The screenshot shows a web browser's developer console. At the top, there are icons for opening the console, toggling it, and a search icon. Below these are icons for a play button, a circle with a slash, and a dropdown menu showing 'top'. To the right of these are icons for a gear (settings), a vertical ellipsis (more options), and a close button (X). Below the icons is a 'Filter' input field and a 'Default levels' dropdown menu. Underneath, it says 'No Issues' followed by a gear icon. At the bottom, there is a log entry: 'x es mayor que 5' followed by a link to 'testeo.js:8'.

Console >> ⚙️ ⋮ ✕

▶ ⛔ top 👁

Filter Default levels ▾

No Issues ⚙️

x es mayor que 5 [testeo.js:8](#)

## Python

### FOR

```
12  for i in range(5):  
13      print(i)
```

## Javascript

### FOR

```
11  for (let i = 0; i < 5; i++) {  
12      console.log(i);  
13  }
```

### WHILE

```
15  x = 0  
16  while x < 5:  
17      print(x)  
18      x += 1
```

```
15  let a = 0;  
16  while (a < 5) {  
17      console.log(a);  
18      a++;  
19  }
```

# LISTAS

```
testeo.py > ...  
1 lista = []  
2 lista.append(1)  
3 lista.append(2)  
4 lista.append(3)  
5 print(lista)  
6 print(lista[2])  
7 print(lista[0:2])
```

```
[1, 2, 3]  
3  
[1, 2]
```

```
JS testeo.js ×  
JS testeo.js > ...  
1 let lista = [];  
2 lista.push(1);  
3 lista.push(2);  
4 lista.push(3);  
5 console.log(lista);  
6 console.log(lista[2]);  
7 console.log(lista.slice(0, 2));
```

```
▶ (3) [1, 2, 3] testeo.js:5  
3 testeo.js:6  
▶ (2) [1, 2] testeo.js:7
```

## Python

```
a = 10
b = 5
max_value = max(a, b)
print(max_value)
```

```
lista = [1, 5, 3, 9, 2]
max_value = max(lista)
print(max_value)
```

## Javascript

```
let a = 10;
let b = 5;
let max_value = Math.max(a, b);
console.log(max_value);
```

```
let lista = [1, 5, 3, 9, 2];
let max_value = Math.max(...lista);
console.log(max_value);
```

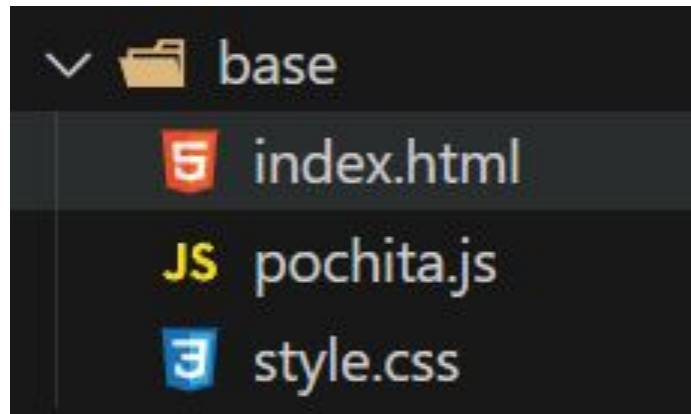
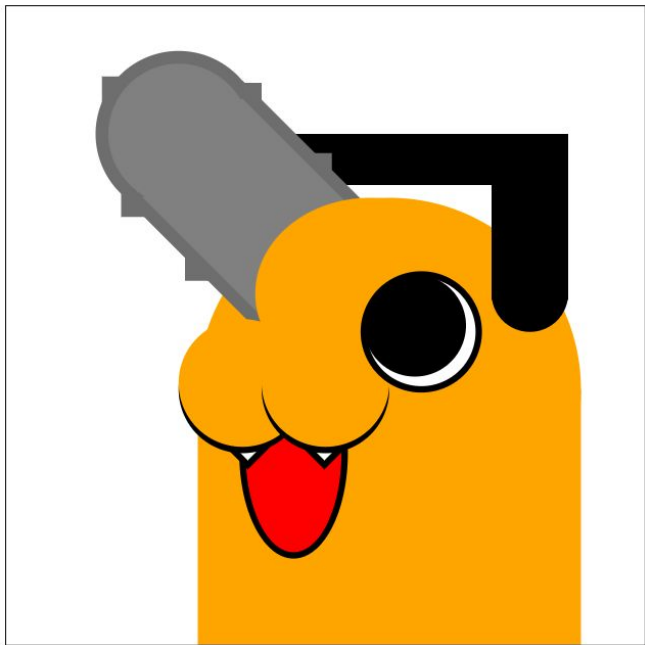


# parseInt

```
let a = "10";  
let parsedInt = parseInt(a); // Por defecto esta con base 10  
console.log(parsedInt); // Imprimirá 10  
  
let b = "1010";  
let parsedIntBinary = parseInt(b, 2); // base 2 (Binario)  
console.log(parsedIntBinary); // Imprimirá 10  
  
let decimalString = "10.78";  
let parsedInt = parseInt(decimalString);  
console.log(parsedInt); // Esto También imprimirá 10
```

# Desafío de esta ayudantía: Dibuja tu Pochita!

Usando Javascript y CSS (No tocaremos el HTML directamente)



## BONUS: Premio stickers durante clase

- El código JS y CSS más ordenado
- Quién le agregue algo a pochita creativo :)
- Las 1ras 5 personas que entreguen la imagen completa

---

---

# IIC2026

## Ayudantía 1 - HTML, CSS y JS

— Sofía Hosiasson —

---

---