IIC2026
Visualización de Información

Hernán F. Valdivieso López
(2022 - 2 / Clase 07)

Antes de empezar... Revisión de contenidos (RC)

1. ¡No se olviden de las últimas actividades publicadas! En Canvas están los *links*.
   Hoy se cierra el de Abstracción de Datos a las 20:00 👀.

2. Hoy no tendremos actividad de revisión de contenidos.

Antes de empezar... Tarea 1

¡Veamos rápidamente el enunciado!

Temas de la clase - Introducción a JavaScript y D3

1. Introducción a Javascript.
    a. Variables, strings, listas y objetos.
    b. Control de flujo y loops.
    c. Funciones y *function arrows.*
    d. HTML y JS
2. Document Object Model (DOM).
3. Introducción a D3.js.

Introducción a Javascript

Introducción a Javascript

- Lenguaje de programación de la web.
- Es un lenguaje de alto nivel, interpretado y multiparadigma.
- Puede ejecutarse en un navegador web y afectar el contenido de una página.
- Lo utilizaremos como lenguaje para construir visualizaciones a partir de datos.

Introducción a Javascript II

- Lenguaje en constante evolución.
- Su estándar se llama ECMAScript (ES).
- Consideraremos desde ES2015 o ES6 en adelante para este curso.

Introducción a Javascript - Variables

Variable (let)

```
let miVariable = 1;
miVariable = 4;
```

Variable (var) → Ya no se ocupa

```
var miVariable = 1;
miVariable = 4; // Funciona, pero ya no se ocupa
```

Constante (const)

```
const miVariableConstante = 1;
miVariableConstante = 4; // ERROR
```

Introducción a Javascript - String y Arrays

String

```
const nombre = "Hernán";
const saludo = "¡Hola " + nombre + "!";
const saludo2 = `¡Hola ${nombre}!`;
```

Arrays

```
let arreglo = ["awa", "ewe", "iwi"];
const variable = arreglo[0]; // "awa"
arreglo.push("uwu");
const largo = arreglo.length; // 4
```

Introducción a Javascript - Objetos

Objetos

```
let miObjeto = {
    title: "Spy x Family",
    year: 2019,
};
```

Obtener dato del objeto

```
const titulo = miObjeto["title"]; // "Spy x Family"
const year = miObjeto.year; // 2019
```

Editar dato del objeto

```
miObjeto["genre"] = ["Comedy", "Spy"];
miObjeto.author = "Tatsuya Endō";
```

Introducción a Javascript - Formato

Uso de llaves. No importa la indentación.

```
if (miVariable > 5) {
 miVariable = miVariable / 2;
}

for (let i = 0; i < 10; i++) {
 miVariable += i;
}

function miFuncion(arg1, arg2) {
    return arg1 + 2 * args2;
}
```

Introducción a Javascript - Control de flujo

```
if (miVariable > 5 && miVariableConstante == 1) {
 // se ejecuta si ambas cumplen
 miVariable = miVariable / 2;
} else if (miVariable > 5 || miVariableConstante == 1) {
 // se ejecuta si alguna cumple
 miVariable -= 2;
} else {
 miVariable = 0;
}
```

Introducción a Javascript - Loops (while)

```
let valor = 0;
let contador = 0;
while (contador < 10) {
 valor += 2;
 contador += 1;
}

let valor = 0;
for (let contador = 0; contador < 10; contador += 1) {
 valor += 2;
}
```

Introducción a Javascript - Loops (for)

```
let valor = 0;
for (let contador = 0; contador < 10; contador += 1) {
 valor += 2;
}

const arreglo = [1, 2, 3];
let suma = 0;
for (const element of arreglo) {
 suma += 0;
}

const objeto = { a: 1, b: 2, c: 3 };
let suma = 0;
for (const propiedad in objeto) {
 suma += objeto[propiedad]
}
```

Introducción a Javascript - Funciones

Opción 1

```
function sumar10(numero) {
 return numero + 10;
}
sumar10(6); // 16
sumar10(sumar10(6)); // 26
```

Opción 2

```
const sumar10 = function(numero) {
 return numero + 10;
}
sumar10(6); // 16
sumar10(sumar10(6)); // 26
```

Introducción a Javascript - *Functions Arrows*

```
const sumar10 = (numero) => numero + 10;
sumar10(6); // 16


const algoMasComplicado = (arg1, arg2, arg3) => {
 // ...
 return valorResultado;
};
algoMasComplicado(1, 2, 3);
```

Introducción a Javascript - 👀🧐

```
const sumar10 = (numero) => numero + 10;

sumar10("a")        // 'a10'
sumar10("1")        // '110'
sumar10([])         // '10'
sumar10([2])        // '210'
sumar10([2, 2, 7]) // '2,2,710'
sumar10(+"2")       // 12
sumar10(-"1")       // 9
sumar10(-"a")       // NaN
```

Aunque Javascript no diga nada, tener cuidado con no mezclar diferentes tipos de datos

Introducción a Javascript - HTML y JS

```
// programa_1.js
let arreglo = [];
for (let numero = 0; numero <= 20; numero += 2) {
  arreglo.push(numero);
}
console.log(arreglo);
for (numero of arreglo) {
  console.log(numero);
}
```

Introducción a Javascript - HTML y JS

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ejemplo con JS 1</title>
  </head>
  <body>
    <script src='programa_1.js' charset='utf-8'></script>
  </body>
</html>
```

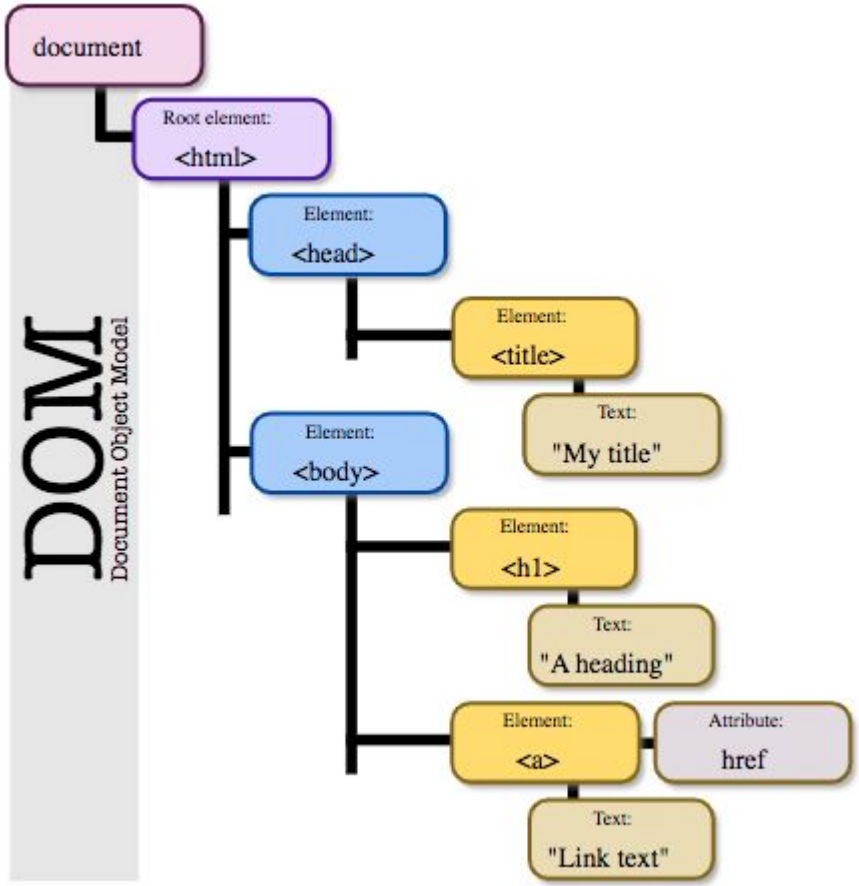
Document Object Model (DOM)

Document Object Model (DOM)

- Forma de representar el contenido de un documento HTML como objeto de programación.

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Title</title>
  </head>

  <body>
    <h1>A heading</h1>
    <a href="...">Link text</a>
  </body>
</html>
```



Fuente: Document Object Model

Document Object Model (DOM)

Con la variable `document` podremos acceder a todo tipo información del HTML

```
// Imprimimos el link de donde está el documento
console.log(document.URL);


// Obtenemos la lista de hijos del documento
document.children; // [<html>]

// Obtenemos elemento HTML con id=”uwu”
const elemento = document.getElementById("uwu");

// Creamos un elemento cuyo tag es <p>
const parrafo = document.createElement("p");
```

Document Object Model (DOM)

```
// Creamos un elemento cuyo tag es <p>
const parrafo = document.createElement("p");

// Creamos un elemento tipo texto
const texto = document.createTextNode("¡Soy un texto >.<!");

// Agregamos el texto al párrafo
parrafo.appendChild(texto);
```

Document Object Model (DOM) - Ejemplo

```
// programa_2.js
const parrafo = document.createElement("p");
const texto = document.createTextNode("¡Soy un texto >.<!");
parrafo.appendChild(texto);

const elementoRaiz = document.getElementById("raiz");
elementoRaiz.appendChild(parrafo);
```

Document Object Model (DOM) - Ejemplo

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ejemplo con JS 2</title>
  </head>

  <body id="raiz">
    <script src='programa_2.js' charset='utf-8'></script>
  </body>
</html>
```

Document Object Model (DOM) - Clases y ID

```
// Creamos un elemento con tag <p>
const elemento = document.createElement("p");

// Definimos la clase del elemento
elemento.className = "importante";

// Agregamos una clase al elemento
elemento.classList.add("otra clase");

// Removemos una clase al elemento
elemento.classList.remove("otra clase");

// Definimos una único id a elemento
elemento.id = "principal";
```

Document Object Model (DOM) - Eventos

```
// Creamos un elemento con tag <h1>
const elemento = document.createElement("h1");

// Conectamos la ocurrencia de un evento en el elemento a una función
elemento.addEventListener(evento, funcion);
```

🤔 ¿Qué evento puede ser?

- Click sobre el elemento (`click`).
- Pasar el mouse sobre el elemento (`mouseover`).
- Copiar el elemento (`copy`).
- Hacer scroll en el elemento (`scroll`).
- Muchos más: https://www.w3schools.com/jsref/dom_obj_event.asp

Document Object Model (DOM) - Ejemplo de evento

```
// programa_3.js
const raiz = document.getElementById("raiz");
const principal = document.getElementById("principal");

let contador = 0;
principal.addEventListener("click", () => {
  contador += 1;
  const parrafo = document.createElement("p");
  const texto = document.createTextNode(`Cantidad de clics o.o: ${contador}`);
  parrafo.appendChild(texto);

  raiz.appendChild(parrafo);
});
```

Document Object Model (DOM) - Ejemplo de evento

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ejemplo con JS 3</title>
  </head>

  <body id="raiz">
    <h1 id="principal">¡Presioname!</h1>
    <script src='programa_3.js' charset='utf-8'></script>
  </body>
</html>
```

Introducción a D3.js

Introducción a D3.js

- Su intención es utilizar HTML, CSS y SVG para crear visualizaciones.
- Apareció en un momento en que esto no era común para herramientas de visualización.
- Escrita por Mike Bostock.

Enlaces de interés:

- Artículo (paper) donde se propone D3.
- Página oficial D3.

Introducción a D3.js

- **NO** es una librería de visualización de alto nivel

```
const grafico_de_barra = crear_grafico_barra(datos); // ❌
grafico_de_barra.graficar(); // ❌
```

- AnyChart es un ejemplo de librería de alto nivel.
- Es una herramienta para **crear visualizaciones desde 0**.
  - Definir cada elemento del SVG.
  - Personalizar los elementos del SVG según los datos.
  - Definir las interacciones posibles.
  - Agregar leyendas a mano.
  - entre otros...

Introducción a D3.js

Utilizaremos las versiones 7 o 6 en el curso.

👀 👀 con recursos y ejemplos escritos en la versión 5 o menor que encuentren.

- 🤦‍♂️ No se aceptará trabajos con versiones diferente a la 6 o 7.
- 👨‍💻 Es su deber adaptar dichos recursos a la versión 6 o 7.

Introducción a D3.js

Cargar D3 en un archivo HTML - Desde URL

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ejemplo con D3</title>
    <script src="https://d3js.org/d3.v7.min.js"></script>
  </head>
  <body>
    <script src='programa_4.js' charset='utf-8'></script>
  </body>
</html>
```

Introducción a D3.js

Cargar D3 en un archivo HTML - De forma local

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ejemplo con D3</title>
    <script src="./ruta/en/computador/d3.v7.min.js"></script>
  </head>
  <body>
    <script src='programa_4.js' charset='utf-8'></script>
  </body>
</html>
```

Introducción a D3.js

```
// programa_4.js
const datos = [150, 256, 130, 0, 11, 420, 235];
// Encontrar el body en el DOM y agregar un elemento SVG. Luego definir ancho y largo
const svg = d3.select("body").append("svg");
svg.attr("width", 50 + datos.length * 100).attr("height", 500);
// Agregar “rect” y personalizar según los datos que tenemos
svg
  .selectAll("rect")
  .data(datos)
  .enter()
  .append("rect")
  .attr("width", 50)
  .attr("fill", "orange")
  .attr("height", (d) => d)
  .attr("x", (_, index) => 50 + index * 100);
```

Próximos eventos

Próxima clase

- *Framework* Tamara Munzner: Abstracción de tareas.
- Control de alternativas para Revisión de contenidos.

Ayudantía de mañana

- Repaso y resolución dudas para la Tarea 1.

IIC2026
Visualización de Información

Hernán F. Valdivieso López
(2022 - 2 / Clase 07)