



Instituto Nacional de Formación  
Técnico Profesional  
[Infotep.gob.do](http://Infotep.gob.do)

Tecnico Ciberseguridad  
Profesor: Kevin Feliz Henriquez  
Nombre: Joice Pérez Paulino



Dapiato Prosalud  
A la vanguardia de la tecnologia  
[secomdr.com](http://secomdr.com)

# Actividad 1: Elementos básicos de la programación

## Ciberseguridad

### Introducción a la programación

#### Actividad 1

#### Variables.

Son contenedores para almacenar datos. Pienso en ellas como cajas etiquetadas donde puedes guardar diferentes tipos de información, como números o textos. Su valor puede cambiar durante la ejecución del programa.

Ejemplo:

```
let edad = 25;
console.log(edad); // salida: 25
edad = 26; // se puede cambiar el valor
console.log(edad); // salida: 26
```

#### Constante.

Son similares a las variables, pero su valor no puede ser reasignado una vez que se le ha asignado un valor inicial. Se utilizan para datos que no deben cambiar. Se declaran con la palabra clave **const**.

Ejemplo:

```
const nombre = "Juan";
console.log(nombre); // salida: "Juan"
// nombre = "Pedro"; // Esto daría error, ya que no se puede reasignar una constante
```

#### Estructuras de Control.

Dirigen el flujo de ejecución de un programa. Permiten que tu código tome decisiones basadas en ciertas condiciones. La más común es la estructura **if...else**.

Ejemplo:

```
let temperatura = 28;
if (temperatura > 25) {
  console.log("¡Hace calor!");
} else {
  console.log("El clima es agradable.");
} // salida: ¡Hace calor!
```

#### Bucles (Loops).

Se usan para repetir un bloque de código varias veces. Son útiles para iterar sobre una lista de elementos o para ejecutar una acción un número determinado de veces. Un tipo de bucle común es el bucle **for**.

Ejemplo:

```
for (let i = 0; i < 3; i++) {
  console.log("El número es: " + i);
} // salida:
// El número es: 0
// El número es: 1
// El número es: 2
```

#### Funciones.

Son bloques de código reutilizables que realizan una tarea específica. Ayudan a organizar el código y a evitar la repetición. Pueden aceptar entradas (llamadas parámetros) y devuelven un resultado.

Ejemplo:

```
función sumar(a, b) {
  return a + b;
}

let resultado = sumar(5, 3);
console.log(resultado); // salida: 8
```

# Codigos de la actividad1 Html.

Actividad 1 - Codigos A Provar

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-
width, initial-scale=1.0">
  <title>Document</title>
  <script type="module" src="app.js"></script>
</head>
<body>
  <h1>Presiona la tecla F12, Luego haz click en
la herramienta de Consola para ver la ejecución del
Codigo Javascript. </h1>
</body>
</html>
```

# Codigos de la actividad1 app.js.

```
app.js
let nombre = "Juan Pérez";
let edad = 30;
let tienePermiso = false;
const PI = 3.14159;
console.log("--- 1. variables y tipos de datos ---");
console.log('nombre: ${nombre}');
console.log('Edad: ${edad}');
console.log('¿tiene permiso? ${tienePermiso}');
console.log('El valor de PI es: ${PI}');
console.log('tipo de dato de edad: ${typeof edad}');
console.log("\n--- 2.1 Condicionales ---");
const Calificación = 85;
```

```
if (Calificación >= 70) {
  console.log("Aprobado!");
} else {
  console.log("Reprobado.");
}

console.log("\n--- 2.2 Bucles ---");
console.log("Bucle for:");
const frutas = ["manzana", "banana", "cereza"];
for (let i = 0; i < frutas.length; i++) {
  console.log('me gusta la ${frutas[i]}');
}
```

```
console.log("Bucle while:");
let contador = 0;
while (contador < 3) {
  console.log('El contador es: ${contador}');
  contador++;
}

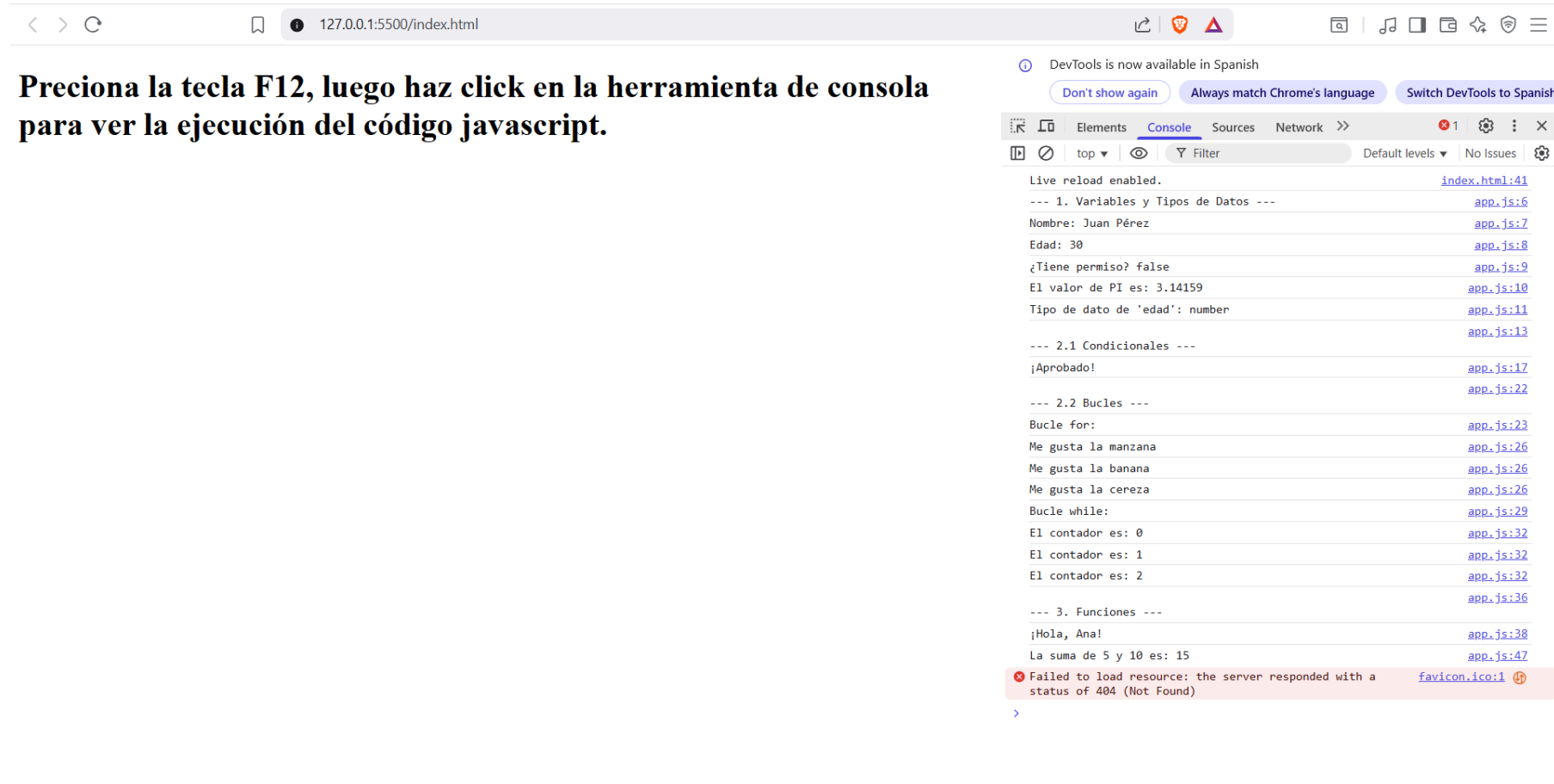
console.log("\n--- 3. Funciones ---");
function saludar(nombre) {
  console.log('¡Hola, ${nombre}!');
}

function sumar(a, b) {
  return a + b;
}

saludar("Ana");
const resultadoSuma = sumar(5, 10);
console.log('La suma de 5 y 10 es: ${resultadoSuma}');
```

# Ejecucion de códigos actividad1

**Preciona la tecla F12, luego haz click en la herramienta de consola para ver la ejecución del código javascript.**





# Audio del manual escrito

