

EVIDENCIAS 8

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Curso JavaScript - Desarrollo Personal</title>
  <style>
    body {
      font-family: 'Roboto', sans-serif; /* Fuente moderna y limpia */
      margin: 20px;
      background-color: #f0f2f5; /* Fondo gris claro y suave */
      color: #333; /* Texto principal oscuro */
    }
    h1 {
      color: #2c3e50; /* Azul oscuro para el título principal */
      text-align: center;
      font-weight: 700; /* Negrita */
      margin-bottom: 30px;
    }
    h2 {
      color: #34495e; /* Azul grisáceo para los subtítulos */
      border-bottom: 2px solid #3498db; /* Línea azul para destacar */
      padding-bottom: 8px;
      margin-top: 35px;
      font-weight: 600;
    }
    h3 {
      color: #3498db; /* Azul medio para los encabezados de sección */
      margin-top: 25px;
      font-weight: 500;
    }
    section {
      background-color: #ffffff; /* Fondo blanco para las secciones */
      margin-bottom: 25px;
      padding: 20px 25px;
      border-radius: 10px; /* Bordes más suaves */
      box-shadow: 0 4px 10px #rgba(0,0,0,0.08); /* Sombra más pronunciada pero sutil */
    }
  </style>
</head>
<body>
  <h1>Curso JavaScript - Desarrollo Personal</h1>
  <h2>Fundamentos de la Sintaxis de Clases</h2>
  <h3>9.1 Fundamentos de la Sintaxis de Clases</h3>
  <div>
    <p>Exploración de la declaración básica de una clase en JavaScript y cómo se instancian objetos a partir de ella, sentando las bases de la programación orientada a objetos.</p>
  </div>
  <div>
    <p>Hola, mi nombre es daniel y tengo 19 años.</p>
    <p>Hola, mi nombre es diaz y tengo 25 años.</p>
  </div>
  <h2>9.2 Herencia entre Clases</h2>
  <div>
    <p>Demostración práctica de cómo una clase puede extenderse de otra, heredando sus propiedades y métodos. Se ilustra la relación "es un" fundamental en el diseño orientado a objetos.</p>
  </div>
  <div>
    <p>Hola, soy Carlos, tengo 20 años y estudio Ingeniería de Sistemas.</p>
    <p>Carlos está estudiando Ingeniería de Sistemas.</p>
  </div>
</body>
</html>
```

```
const suma = Matematica.sumar(5, 3);
const resta = Matematica.restar(10, 4);
const piValor = Matematica.PI;
document.getElementById('output-9-3').innerHTML += `<p>Suma: ${suma}</p><p>Resta: ${resta}</p><p>PI: ${piValor}</p>`;
console.log("9.3: Suma =", suma, "Resta =", resta, "PI =", piValor);

// TEMA: 9.4 Propiedades y métodos privados y protegidos
class CuentaBancaria {
  #saldo;
  _titular;
  constructor(titular, saldoInicial) {
    this.titular = titular;
    this.#saldo = saldoInicial;
  }
  #generarNumeroCuenta() { return Math.floor(Math.random() * 1000000); }
  depositar(cantidad) {
    if (cantidad > 0) { this.#saldo += cantidad; console.log(`Depósito: ${cantidad}. Saldo: ${this.#saldo}`); }
  }
  retirar(cantidad) {
    if (cantidad > 0 && cantidad <= this.#saldo) { this.#saldo -= cantidad; console.log(`Retiro: ${cantidad}. Saldo: ${this.#saldo}`); }
    else { console.log("Fondos insuficientes."); }
  }
  getSaldo() { return this.#saldo; }
  getNumeroCuentaVisible() { return `Número de cuenta: ${this.#generarNumeroCuenta()}`; }
}

const miCuenta = new CuentaBancaria("Juan Perez", 1000);
miCuenta.depositar(200);
miCuenta.retirar(500);
document.getElementById('output-9-4').innerHTML += `<p>Titular: ${miCuenta.titular}</p><p>Saldo: ${miCuenta.getSaldo()}</p>`;
console.log("9.4: Titular =", miCuenta.titular, "Saldo =", miCuenta.getSaldo());

// TEMA: 9.5 Ampliación de clases integradas
class MiArrayExtendido extends Array {
  sumarTodos() { return this.reduce((acc, current) => acc + current, 0); }
  obtenerPares() { return this.filter(num => num % 2 === 0); }
}
```

```
// TEMA: 9.1 Sintaxis básica de clase
class Persona {
  constructor(nombre, edad) {
    this.nombre = nombre;
    this.edad = edad;
  }
  saludar() {
    return `Hola, mi nombre es ${this.nombre} y tengo ${this.edad} años.`;
  }
}

const persona1 = new Persona("daniel", 19);
const persona2 = new Persona("diaz", 25);
document.getElementById('output-9-1').innerHTML += `<p>${persona1.saludar()}</p><p>${persona2.saludar()}</p>`;
console.log("9.1:", persona1.saludar());

// TEMA: 9.2 Herencia de clase
class Estudiante extends Persona {
  constructor(nombre, edad, carrera) {
    super(nombre, edad);
    this.carrera = carrera;
  }
  estudiar() {
    return `${this.nombre} está estudiando ${this.carrera}.`;
  }
  saludar() {
    return `Hola, soy ${this.nombre}, tengo ${this.edad} años y estudio ${this.carrera}.`;
  }
}

const estudiante1 = new Estudiante("Carlos", 20, "Ingeniería de Sistemas");
document.getElementById('output-9-2').innerHTML += `<p>${estudiante1.saludar()}</p><p>${estudiante1.estudiar()}</p>`;
console.log("9.2:", estudiante1.saludar());

// TEMA: 9.3 Propiedades y métodos estáticos
class Matematica {
  static PI = 3.14159;
  static sumar(a, b) { return a + b; }
  static restar(a, b) { return a - b; }
}
```

```
// TEMA: 9.5 Ampliación de clases integradas
class MiArrayExtendido extends Array {
  sumarTodos() { return this.reduce((acc, current) => acc + current, 0); }
  obtenerPares() { return this.filter(num => num % 2 === 0); }
}

const numeros = new MiArrayExtendido(1, 2, 3, 4, 5, 6);
const sumaTotal = numeros.sumarTodos();
const pares = numeros.obtenerPares();
document.getElementById('output-9-5').innerHTML += `<p>Array: [${numeros.join(', ')}]</p><p>Suma: ${sumaTotal}</p><p>Pares: [${pares.join(', ')}]</p>`;
console.log("9.5: Suma =", sumaTotal, "Pares =", pares);

// TEMA: 9.6 Comprobación de clase: "instanceof"
class Animal {}
class Perro extends Animal {}
const unPerro = new Perro();
const unaPersona = new Persona("Laura", 40);
document.getElementById('output-9-6').innerHTML += `<p>unPerro instanceof Perro: ${unPerro instanceof Perro}</p>`;
console.log("9.6: instanceof. Ver HTML.");

// TEMA: 9.7 Mixins
let SaludarMixin = {
  decirHola() { console.log(`Hola, soy ${this.nombre}.`); },
  decirAdios() { console.log(`Adiós de parte de ${this.nombre}.`); }
};

class Usuario {
  constructor(nombre) { this.nombre = nombre; }
}

Object.assign(Usuario.prototype, SaludarMixin);
const usuario1 = new Usuario("Maria");
usuario1.decirHola();
usuario1.decirAdios();
```

JavaScript - Estructuras de Clase

9.1 Fundamentos de la Sintaxis de Clases

Exploración de la declaración básica de una clase en JavaScript y cómo se instancian objetos a partir de ella, sentando las bases de la programación orientada a objetos.

Hola, mi nombre es daniel y tengo 19 años.

Hola, mi nombre es diaz y tengo 25 años.

9.2 Herencia entre Clases

Demostración práctica de cómo una clase puede extenderse de otra, heredando sus propiedades y métodos. Se ilustra la relación "es un" fundamental en el diseño orientado a objetos.

Hola, soy Carlos, tengo 20 años y estudio Ingeniería de Sistemas.

Carlos está estudiando Ingeniería de Sistemas.