

¡Es hora de poner a prueba cuánto sabes sobre JavaScript!

Instrucciones para tomar esta prueba

- Evalúa muy críticamente tu conocimiento.
- Si logras resolver la prueba, no importa cuánto te cueste, puedo asegurarte que tienes todo para continuar a las siguientes clases y tomar el resto del curso.
- Si no lo logras, no te preocupes, absolutamente nadie puede juzgarte, solo tú. Vuelve al [Curso Básico de JavaScript](#), anota los temas clave donde puedes mejorar, ubica las clases donde puedes aprenderlos y estudia vigorosamente.
- Es completamente válido hacer búsquedas en Google, cursos y tutoriales de Platzi, incluso usar tu cuaderno de notas sin importar si es físico o virtual.

Recuerda que el éxito no se mide por cuánto tiempo te toma aprender, esa métrica es relativamente inútil. Mejor concéntrate en completar los cursos de tu ruta de aprendizaje profesional y desarrollar los proyectos que realmente demuestran que dominas cada tecnología.

¡Mucha suerte!

Variables y operaciones

1 Responde las siguientes preguntas en la sección de comentarios:

1. ¿Qué es una variable y para qué sirve?
 - ☒ Una variable es un espacio de memoria que permite almacenar datos de distinto tipo.
2. ¿Cuál es la diferencia entre declarar e inicializar una variable?
 - ☒ Inicializar una variable es crear (declarar) una variable y asignarle un valor para que se guarde. Declarar una variable, en cambio, es crear dicha variable sin asignarle ningún tipo de dato al bloque de memoria.
3. ¿Cuál es la diferencia entre sumar números y concatenar strings?
 - ☒ Sumar números es una operación matemática, concatenar se trata de unir sin mezclar. Es decir, sin que cambien los valores originales que se están empleando.
4. ¿Cuál operador me permite sumar o concatenar?

- ☒ En JS el símbolo + permite realizar ambas operaciones, depende del contexto y la sintaxis si aplica una u otra operación.

2 Determina el nombre y tipo de dato para almacenar en variables la siguiente información:

- Nombre
 - ☒ string
- Apellido
 - ☒ string
- Nombre de usuario en Platzi
 - ☒ string
- Edad
 - ☒ number
- Correo electrónico
 - ☒ string
- Mayor de edad
 - ☒ boolean
- Dinero ahorrado
 - ☒ number
- Deudas
 - ☒ number

3 Traduce a código JavaScript las variables del ejemplo anterior y deja tu código en los comentarios.

```
var nombre = prompt ("Escriba su nombre");
var apellido = prompt ("Escriba su apellido");
var user = prompt ("Escriba su nombre de usuario");
var edad = prompt ("Escriba su edad");
var correo = prompt ("Escriba su correo electrónico");
//Si es mayor de edad:
var edad = prompt("Escriba su edad");
console.log (edad > 18);
//
var ahorro;
var deuda;
```

4 Calcula e imprime las siguientes variables a partir de las variables del ejemplo anterior:

- Nombre completo (nombre y apellido)

```
console.log(nombreCompleto = nombre + " " + apellido)
```

- Dinero real (dinero ahorrado menos deudas)

```
console.log(dineroReal = ahorro-deuda)
```

Funciones

1 Responde las siguientes preguntas en la sección de comentarios:

- ¿Qué es una función?
 - Una función es un bloque de código que realiza tareas asignadas y tiene la característica de ser reutilizable. Este acepta entradas (argumentos/parámetros) y retornar un dato en concreto resultado de la operación de las tareas dentro de la misma
- ¿Cuándo me sirve usar una función en mi código?
 - Siempre que sea posible, ya que las funciones solo pueden aportar ventajas
- ¿Cuál es la diferencia entre parámetros y argumentos de una función?
 - Los parámetros son las variables declaradas en la definición de la función que actúan como plantilla cuando se invoque dicha función, para que esta trabaje con ellos.
 - Los argumentos, son los valores que se le asignan a esas parámetros de la función cuando se llama dicha función.

2 Convierte el siguiente código en una función, pero, cambiando cuando sea necesario las variables constantes por parámetros y argumentos en una función:

```
const name = "Juan David";
const lastname = "Castro Gallego";
const completeName = name + lastname;
const nickname = "juandc";

console.log("Mi nombre es " + completeName + ", pero prefiero que me digas " + nickname + ".");
```

Solución:

```
function info (name, lastname, nickname){
    const completeName = name + " " + lastname;
    console.log("Mi nombre es " + completeName + ", pero prefiero que me digas " + nickname + ".");
}

info(prompt("Escriba su nombre"), prompt("Escriba su apellido"), prompt("Escriba su nickname"));
```

Condicionales

1 Responde las siguientes preguntas en la sección de comentarios:

- ¿Qué es un condicional?
 - Es un bloque de código que ejecuta unas instrucciones u otras basándose en si una condición específica es verdadera o falsa.
- ¿Qué tipos de condicionales existen en JavaScript y cuáles son sus diferencias?
 - If
 - Se utiliza cuando tienes una condición booleana para evaluar (verdadero/falso).
 - Puedes tener múltiples bloques else if para evaluar condiciones adicionales.
 - Switch
 - Se utiliza cuando quieres comparar una expresión con varios valores posibles.
 - Cada case representa un valor específico.
 - La ejecución se detiene cuando se encuentra un case que coincide (se debe usar break)

- ¿Puedo combinar funciones y condicionales?
 - Sí

2 Replica el comportamiento del siguiente código que usa la sentencia switch utilizando if, else y else if:

```
const tipoDeSuscripcion = "Basic";

switch (tipoDeSuscripcion) {
  case "Free":
    console.log("Solo puedes tomar los cursos gratis");
    break;
  case "Basic":
    console.log("Puedes tomar casi todos los cursos de Platzi durante un mes");
    break;
  case "Expert":
    console.log("Puedes tomar casi todos los cursos de Platzi durante un año");
    break;
  case "ExpertPlus":
    console.log("Tú y alguien más pueden tomar TODOS los cursos de Platzi durante un año");
    break;
}
```

Solución

```
const tipoDeSuscripción = "Basic"

if (tipoDeSuscripción === "Basic"){
  console.log("Puedes tomar casi todos los cursos de Platzi durante un mes");
} else if (tipoDeSuscripción === "Free") {
  console.log("Solo puedes tomar los cursos gratis");
} else if (tipoDeSuscripción === "Expert") {
  console.log("Puedes tomar casi todos los cursos de Platzi durante un año");
} else if (tipoDeSuscripción === "ExpertPlus") {
  console.log("Tú y alguien más pueden tomar TODOS los cursos de Platzi durante un año");
}
```

3 Replica el comportamiento de tu condicional anterior con if, else y else if, pero ahora solo con if (sin else ni else if).

💡 Bonus: si ya eres una experta o experto en el lenguaje, te desafío a comentar cómo replicar este comportamiento con arrays u objetos y un solo condicional. 😊

Solución con solo if:

```
const tipoDeSuscripcion = "Basic";

if (tipoDeSuscripcion === "Free") {
  console.log("Solo puedes tomar los cursos gratis");
}

if (tipoDeSuscripcion === "Basic") {
  console.log("Puedes tomar casi todos los cursos de Platzi durante un mes");
}

if (tipoDeSuscripcion === "Expert") {
  console.log("Puedes tomar casi todos los cursos de Platzi durante un año");
}

if (tipoDeSuscripcion === "ExpertPlus") {
  console.log("Tú y alguien más pueden tomar TODOS los cursos de Platzi durante un año");
}
```

Solución 2:

```
const tipoDeSuscripcion = "ExpertPlus";

let suscripciones = {
  Free: 'Solo puedes tomar los cursos gratis',
  Basic: 'Puedes tomar casi todos los cursos de Platzi durante un mes',
  Expert: 'Puedes tomar casi todos los cursos de Platzi durante un año',
  ExpertPlus: 'Tú y alguien más pueden tomar TODOS los cursos de Platzi durante un año'
};

const mensaje = suscripciones[tipoDeSuscripcion]; // Los objetos pueden ser accesados similar a
// Traducido a lenguaje natural está diciendo: Dentro de suscripciones, busque tipoDeSuscripcion
if (mensaje){
  console.log(`Según su suscripción, ${mensaje}`)
} else {
  console.log("No está suscrito")
}
```

Ciclos

1 Responde las siguientes preguntas en la sección de comentarios:

- ¿Qué es un ciclo?
 - Un ciclo o bucle es una instrucción de código que se ejecuta hasta el final y se repite hasta que se cumpla la condición
- ¿Qué tipos de ciclos existen en JavaScript?
 - Existen cuatro tipos de bucles for: for, for in, for of y for await of.
- ¿Qué es un ciclo infinito y por qué es un problema?
 - Un ciclo infinito es aquel en el que su condición no se cumplirá jamás, por lo tanto iterará indefinidamente. Es peligroso porque consumirá recursos progresivamente hasta generar un fallo a nivel de software o hardware.
- ¿Puedo mezclar ciclos y condicionales?
 - Sí.

2 Replica el comportamiento de los siguientes ciclos for utilizando ciclos while:

```
for (let i = 0; i < 5; i++) {  
  console.log("El valor de i es: " + i);  
}
```

```
for (let i = 10; i >= 2; i--) {  
  console.log("El valor de i es: " + i);  
}
```

Solución ejercicio 1:

```
let i = 0;  
  
while (i<5) {  
  console.log("El valor de i es: " + i);  
  i++;  
}
```

Solución ejercicio 2:

```
let i = 10;  
  
while (i>=2) {  
  console.log(`El valor de i es: ${i}`);  
  i--;  
}
```

3 Escribe un código en JavaScript que le pregunte a los usuarios cuánto es $2 + 2$. Si responden bien, mostramos un mensaje de felicitaciones, pero si responden mal, volvemos a empezar.

💡 Pista: puedes usar la función prompt de JavaScript.

Solución:

```
let pregunta = prompt("Cuánto es 2+2")

while (pregunta !== "4"){
  console.log(`Intente de nuevo`)
  pregunta = prompt("Cuánto es 2+2")
}
console.log(`Correcto!`)
```

Listas

1 Responde las siguientes preguntas en la sección de comentarios:

- ¿Qué es un array?
- ¿Qué es un objeto?
- ¿Cuándo es mejor usar objetos o arrays?
- ¿Puedo mezclar arrays con objetos o incluso objetos con arrays?

2 Crea una función que pueda recibir cualquier array como parámetro e imprima su primer elemento.

3 Crea una función que pueda recibir cualquier array como parámetro e imprima todos sus elementos uno por uno (no se vale imprimir el array completo).

4 Crea una función que pueda recibir cualquier objeto como parámetro e imprima todos sus elementos uno por uno (no se vale imprimir el objeto completo).

¿Cómo te fue? 🏆

¡Felicidades por completar la prueba de JavaScript! Confío en que hayas completado cada paso y hayas pausado para repasar los temas de los ejercicios que se te complicaron.

Ahora sí, continúa a la siguiente clase, pero recuerda que ya no puedes abandonar el curso, debes completarlo hasta el final. No importa cuánto tiempo te tome. Yo sé que tú puedes. Y tú deberías de saberlo también.

¡Te espero en la siguiente clase para comenzar!