### Variables y operaciones

1. Responde las siguientes preguntas en la sección de comentarios:

* ¿Qué es una variable y para qué sirve?
* Es un espacio en memoria que nos permite almacenar diversos tipos de datos.
* ¿Cuál es la diferencia entre declarar e inicializar una variable?
* Declarar una variable incluye únicamente la creación, pero inicializarla es directamente asignarle un valor a la variable
* ¿Cuál es la diferencia entre sumar números y concatenar strings?
* A pesar de que se usa el miso signo, al sumar números estamos realizando una operación matemática mientras que al concatenar string únicamente estamos uniendo palabras.
* ¿Cuál operador me permite sumar o concatenar?
* Signo +

1. 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” (@ y dominio)
* Mayor de edad = Boolean
* Dinero ahorrado = Number
* Deudas = Number

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

let name = "Lorena";

let lastname = "Murcia";

let platziUserName = "Indefinido";

let age = 22;

let mail = "karitolmurciap@gmial.com";

let adult = true;

let savedMoney = 200000;

let credit = 10000000;

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

Nombre completo (nombre y apellido)

console.log (`Mi nombre completo es ${name} ${lastname}`);

Dinero real (dinero ahorrado menos deudas)

console.log(`El dinero con que cuento actualmente es: ${realMoney}`);

### Funciones

1. Responde las siguientes preguntas en la sección de comentarios:

* ¿Qué es una función?
* Las funciones nos permiten encapsular bloques de código para reutilizarlos y ejecutarlos en el futuro.
* ¿Cuándo me sirve usar una función en mi código?
* Cuando se requiere realizar acciones repetitivas y optimizar código.
* ¿Cuál es la diferencia entre parámetros y argumentos de una función?
* Un parámetro es el valor que recibe la función cuando está siendo creada y el argumento es el valor que se ingresa en la función al ser llamada

### Condicionales

1. Responde las siguientes preguntas en la sección de comentarios:

* ¿Qué es un condicional?
* Es una expresión que nos permite validar si se cumple o no una acción determinada.
* ¿Qué tipos de condicionales existen en JavaScript y cuáles son sus diferencias?
* If (else y else if)🡪 Permite validar si una acción se cumple y conforme a eso realizar otra.
* Switch 🡪 Nos permite agregar una validación y por medio de casos indicar qué queremos que pase en cada ocasión
* ¿Puedo combinar funciones y condicionales?
* Si, es posible. Las funciones pueden encapsular cualquier bloque de código incluyendo condicionales.

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

Texto

Descripción generada automáticamente

var opc1 = "Free";

var opc2 = "Basic";

var opc3 = "Expert";

var opc4 = "ExpertPlus";

var tipoDeSuscripcion1 = function (suscripcion) {

    if (suscripcion == opc1){

        return ("Solo puedes tomar los cursos gratis");

    } else if (suscripcion == opc2){

        return ("Puedes tomar casi todos los cursos de Platzi durante un mes");

    } else if (suscripcion == opc3){

        return ("Puedes tomar casi todos los cursos de Platzi durante un año");

    } else if (suscripcion == opc4){

        return ("Tú y alguien más pueden tomar TODOS los cursos de Platzi durante un año");

    }

}

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

let opc1 = "Free";

let opc2 = "Basic";

let opc3 = "Expert";

let opc4 = "ExpertPlus";

let tipoDeSuscripcion1 = function (suscripcion) {

    if (suscripcion == opc1){

        return ("Solo puedes tomar los cursos gratis");

    } if (suscripcion == opc2){

        return ("Puedes tomar casi todos los cursos de Platzi durante un mes");

    } if (suscripcion == opc3){

        return ("Puedes tomar casi todos los cursos de Platzi durante un año");

    } if (suscripcion == opc4){

        return ("Tú y alguien más pueden tomar TODOS los cursos de Platzi durante un año");

    }

}

💡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. 😏

### Ciclos

1. Responde las siguientes preguntas en la sección de comentarios:

* ¿Qué es un ciclo?
* Un método que permite repetir una acción hasta que otra se cumpla y rompa el ciclo
* ¿Qué tipos de ciclos existen en JavaScript?
* While, for y do while.
* ¿Qué es un ciclo infinito y por qué es un problema?
* Es un bucle que no tiene fin y es un problema porque no permitirá que el resto de las acciones se cumplan.
* ¿Puedo mezclar ciclos y condicionales?
* Si es posible

1. Replica el comportamiento de los siguientes ciclos for utilizando ciclos while:

Una captura de pantalla de un celular con texto e imagen

Descripción generada automáticamente con confianza media

let i = 0;

while (i < 5) {

  i++;

  console.log(`El valor de i es: ${i}`)

}

Una captura de pantalla de un celular con texto e imagen

Descripción generada automáticamente con confianza media

let i=10;

while (i>=2){

    i--;

    console.log (`El valor de i es ${i}`);

}

1. 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.

let n1 = 2;

let n2 = 2;

let resultado = n1+n2;

function operacion (){

    var rtaUser = prompt(`¿Cuánto es ${n1} + ${n2}`);

    if (rtaUser == resultado){

        alert ("Tu respuesta es correcta 😊");

    }else {

        alert ("Tu respuesta es incorrecta 😔");

    }

}

operacion();

### Listas

1. Responde las siguientes preguntas en la sección de comentarios:

* ¿Qué es un array?
* Es un dato tipo objeto que permite realizar un conjunto o lista de elementos
* ¿Qué es un objeto?
* Es un conjunto de datos que relaciona varias variables con elementos.
* ¿Cuándo es mejor usar objetos o arrays?
* Se usan los objetos cuando existen distintos tipos de datos dentro de un conjunto muy grande de elementos.
* Arrays cuando lo que haremos en un elemento es lo mismo que en todos los demás (la aplicación se puede incumplir). Mientras que un objeto es cuando los nombres de cada elemento son importantes para nuestro algoritmo.
* ¿Puedo mezclar arrays con objetos o incluso objetos con arrays?
* Si, es posible. Los array pueden guardar objetos y los objetos pueden guardar arrays

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

var lista = function (array) {

    console.log(array[0]);

}

1. 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).

var lista = function (array) {

    for (var i=0; i< array.length; i++){

        console.log(`${i+1}`)

    }

}

1. 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).

var computador = {

    tipo: "portátil",

    color: "gris",

    marca: "hp"

};

function equipo(objeto) {

    for (var key in objeto){

        console.log (objeto[key]);

    }

}

equipo(computador);

CICLOS, LISTAS, PROMT JAVASCRIPT