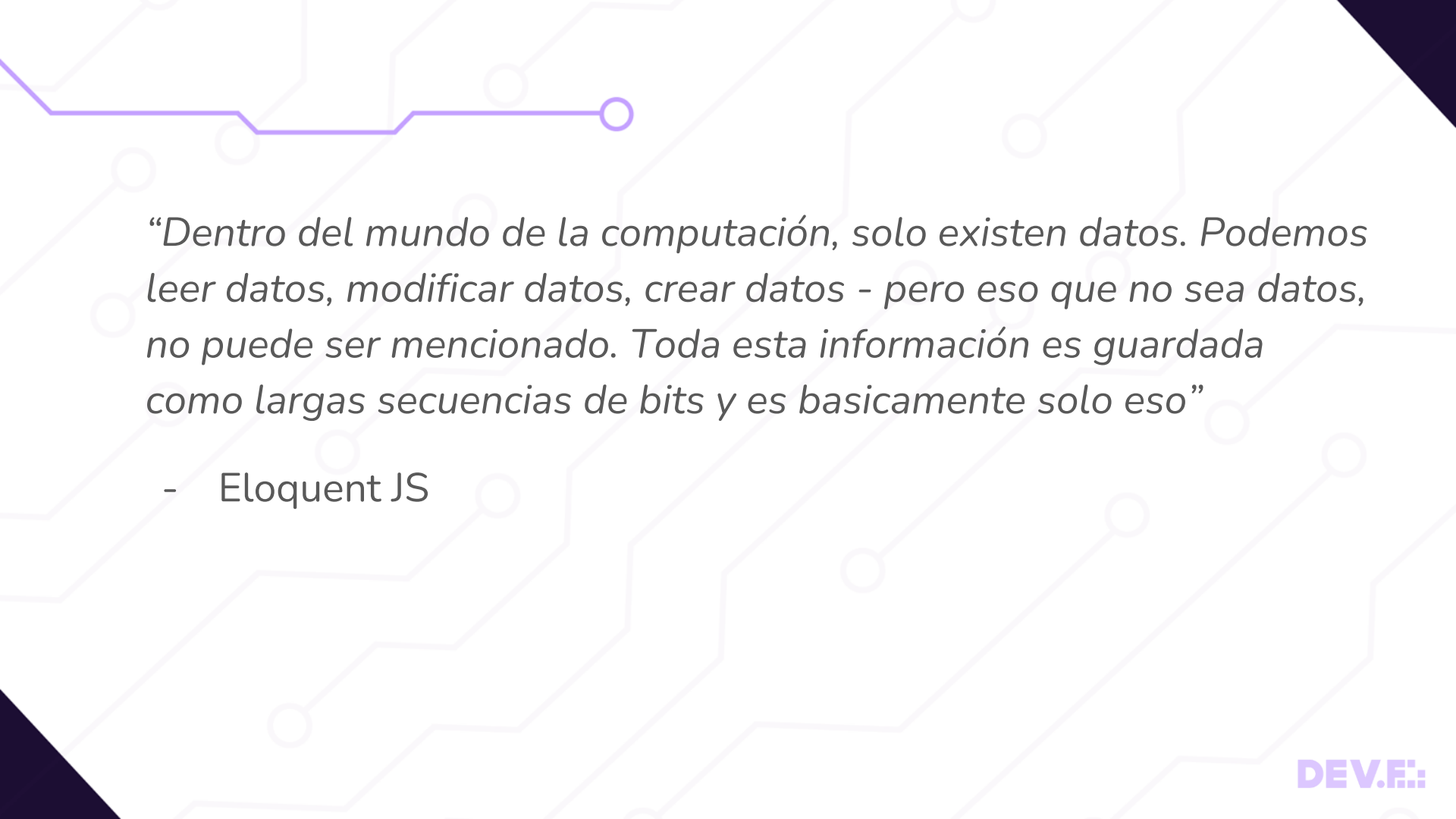


# Tipos de Dato, Variables y Operadores

**DEV.FX**  
DESARROLLAMOS(PERSONAS);

A decorative background pattern consisting of light purple lines and circles, resembling a circuit board or a network diagram. The lines are of varying thickness and form a complex, interconnected web. The circles are small and are placed at various points along the lines.

*“Dentro del mundo de la computación, solo existen datos. Podemos leer datos, modificar datos, crear datos - pero eso que no sea datos, no puede ser mencionado. Toda esta información es guardada como largas secuencias de bits y es basicamente solo eso”*

- Eloquent JS

# Variables

**DEV.F**  
DESARROLLAMOS(PERSONAS);

# Variables

Una variable es un símbolo que permite identificar a un elemento.



# Variables

Normalmente se utilizan para guardar/almacenar información para su uso posterior.



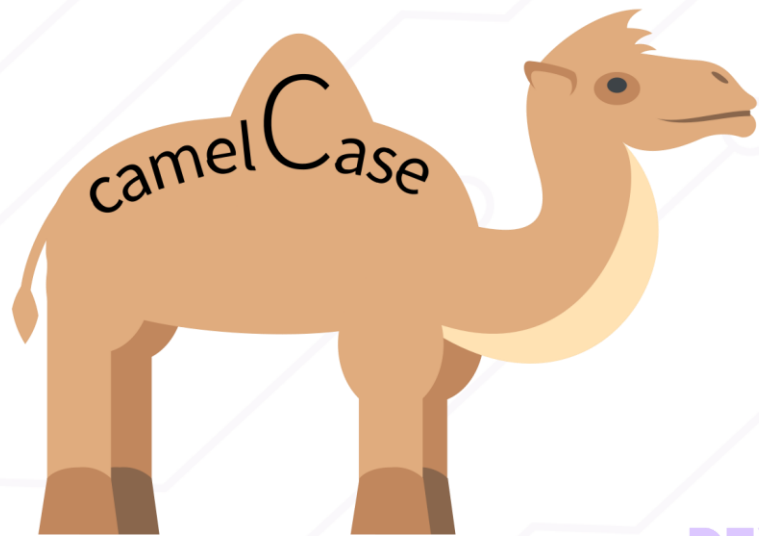
```
/* Guardo el string Pedro en la variable nombre */  
var nombre = 'Pedro';
```

## Tips: Nombrando Variables

Es importante dar un nombre descriptivo a nuestra variable. Por ejemplo, es mejor: **var temperatura;** que **var t;**

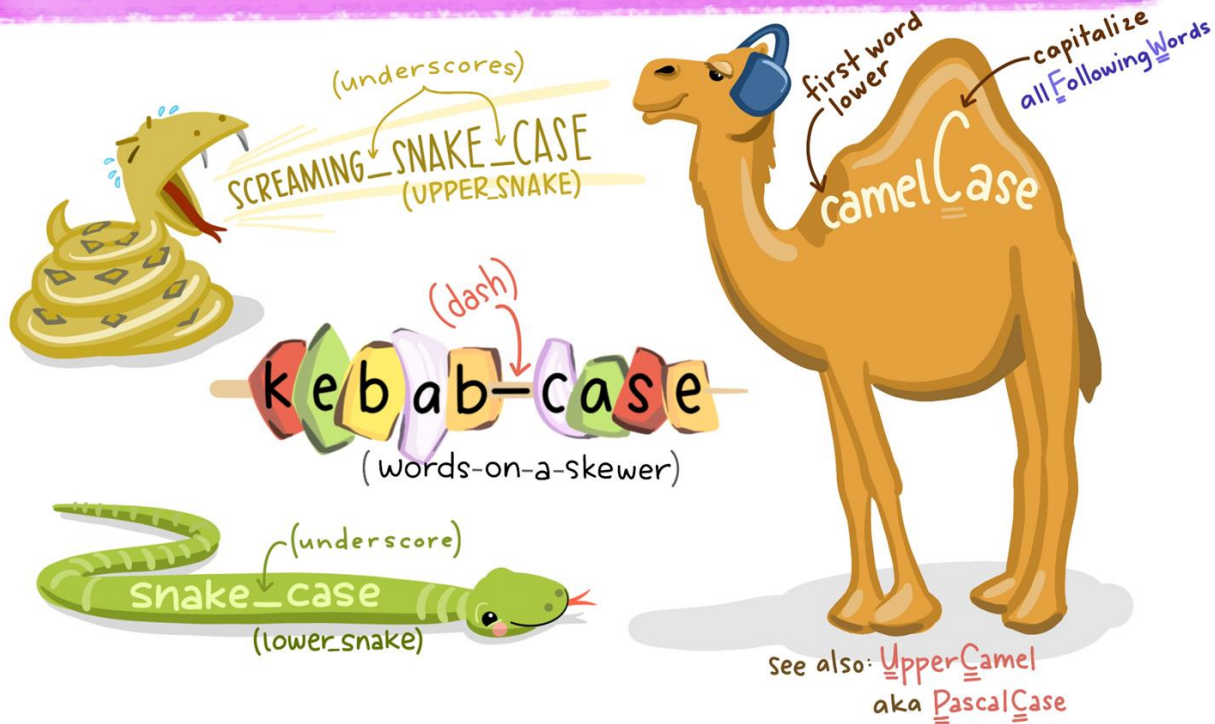
En JavaScript escribimos los nombres de variables usando **camelCase**.

Es decir, primera letra minúscula, y siguientes palabras comienzan con Mayúscula, sin usar espacios.



# Dato Cultural: ¿Qué otras formas existen?

in that case...



# Tipos de Dato

**DEV.F**  
DESARROLLAMOS(PERSONAS);



# “Strings”

El tipo **String** de JavaScript se utiliza para representar datos textuales.

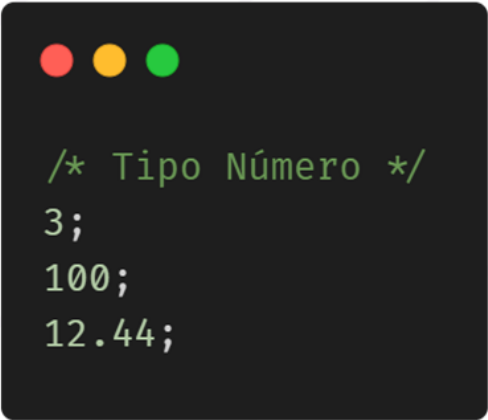
Normalmente se representan encerrando el valor textual entre comillas simples o dobles.



```
"Soy un String";  
'También soy un String';
```

# Numbers

Son valores numéricos (entre  $-(2^{53} - 1)$  y  $2^{53} - 1$ ). También comprende números de punto flotante (decimales).



```
/* Tipo Número */  
3;  
100;  
12.44;
```

También pueden contener valores **NaN** ("Not a Number" o No es un número).

# Booleans

En términos de informática, un **boolean** es un dato lógico que solo puede tener los valores **true (verdadero)** o **false (falso)**.



```
/* Tipo Boolean */  
true; // Verdadero  
false; // Falso
```

Juegan un papel fundamental en la lógica de programación, ya que podemos establecer acciones dependiendo si se cumple o no cierta condición.

# Null

El tipo Null tiene el valor: **nulo**.

```
var miVariable = null  
  
console.log(miVariable) // null
```

Un valor **nulo** significa que es un valor desconocido, indefinido o no inicializado

# Undefined

Una variable a la cual no se le haya asignado valor tiene el valor **indefinido**.

```
var saludo;  
console.log(typeof saludo); // "undefined"
```

# Symbol

Symbol es un tipo de datos cuyos valores son **únicos** e **inmutables**. Dichos valores pueden ser utilizados como identificadores (claves) de las propiedades de los objetos. Cada valor del tipo **Symbol** tiene asociado un valor del tipo String o Undefined que sirve únicamente como descripción del símbolo.

# Object

Un Objeto es un valor en memoria al que podemos acceder por un identificador.

```
var persona = {  
  nombre: "Marcel",  
  edad: "38",  
  ciudad: "Alaior",  
};
```

En JavaScript los objetos pueden ser vistos como una colección de propiedades.

# Operadores

**DEV.F**  
DESARROLLAMOS(PERSONAS);



# Asignación

=

a = 2

# Operadores Matemáticos

Suma (+)

Resta (-)

Multiplicación (\*)

División (/)

Residuo de una división (%)

Exponencial ( \*\* )

# Incremento y decremento

Incremento:

++

+=

Decremento:

--

-=

# Typeof operator

```
console.log(typeof 4.5)
```

```
// -> number
```

```
console.log(typeof "x")
```

```
// -> string
```

# Boolean

## Comparación

>

<

>=

<=

## Igualdad

==

===

!=

!==

## Lógicos

&&

||

# BREAK



DEV.F

dev.f

# Estructura de un programa

**DEV.F**  
DESARROLLAMOS(PERSONAS);

# Expresiones, declaraciones y variables

```
var precio = 123.50
```

```
var cantidad = 2
```

```
var total = precio * cantidad
```

```
console.log(total) // 247
```



# ACTIVIDAD 1

Crea un repositorio nuevo en github llamado: **intro-js**.

1. En la consola del navegador, crea un pequeño programa que le pida al usuario que ingrese un número, multiplica ese número por 2 y muestra en salida el resultado de esa operación. Guarda el código en un archivo js.
2. En la consola del navegador escribe un programa que reciba 2 números e imprime en pantalla la resta de ambos. Guarda el código en un archivo js.
3. Escribe un programa que calcule el área de un círculo con un radio  $r = 5.3$ . Imprime el área en pantalla. (Fórmula:  $\pi \cdot r^2$ )
4. Escribe un programa que calcule el área de un rombo, sabiendo que  $p = 5\text{cm}$  y  $D = 8\text{cm}$ ,  $d = 6\text{cm}$  (Fórmula:  $a = D \cdot d / 2$ )