


# Iniciando con funciones

¡La práctica hace al maestro!. A lo largo del curso tendrás diversas oportunidades para implementar el uso de funciones, e incorporar sus mejores usos y beneficios. Cada ejercicio te permitirá adquirir las habilidades para mejorar en su implementación.

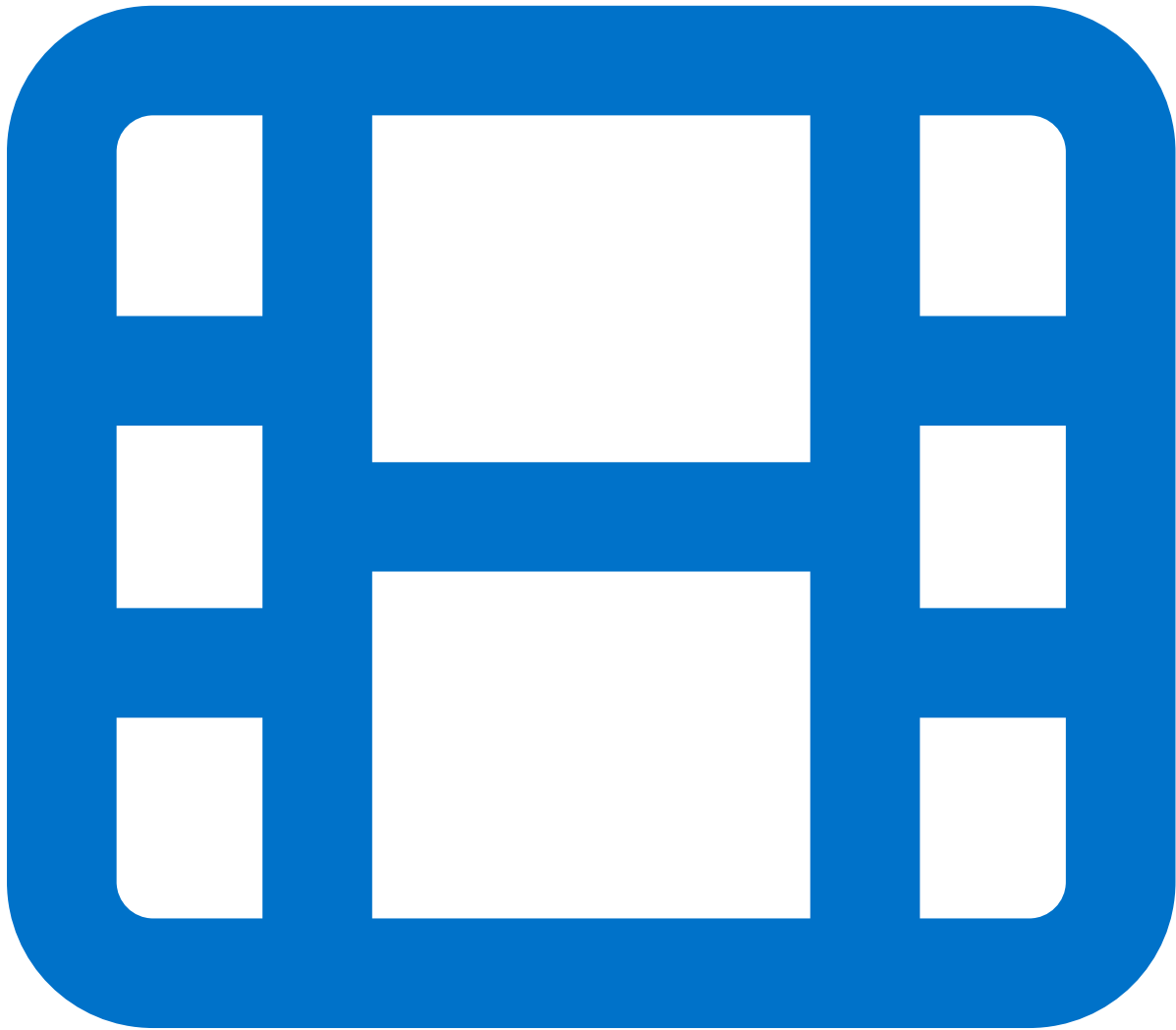
 **¡Importante!** Recuerda que no estarás sol@ en este viaje de aprendizaje. Siempre puedes recurrir al material teórico como referencia, interactuar con tus compañer@s de la mesa de trabajo y solicitar la ayuda de nuestros mentores expertos. Estamos aquí para apoyarte en cada paso del camino y asegurarnos de que tengas una experiencia de aprendizaje enriquecedora.

---



## Material Audiovisual

Mira el siguiente video para reforzar el contenido teórico y ayudar a resolver las actividades más rápido:



## Funciones | Egg

---



### Actividad: Función de cálculo de velocidad

En esta actividad, definirás una función que calcule la velocidad de un automóvil teniendo en cuenta la distancia recorrida y el tiempo que tarda en recorrerla. Sigue los siguientes pasos:

1. Desde Visual Studio Code, abre la carpeta **egg\_js**
2. Crea una carpeta llamada **speed**

### 3. Dentro de esta carpeta:

1. Crea un archivo index.html con su estructura básica.
2. Crea un archivo index.js y conéctalo con el html.
4. En index.js, escribe la sintaxis básica de una función común con las siguientes instrucciones:

1. Define una variable para la distancia
2. Define una variable para el tiempo
3. Define una variable para el cálculo de la velocidad (distancia \* velocidad)
4. Imprime en la consola el resultado.

```
function calcularVelocidad() {  
  const distancia = 10  
  const tiempo = 2  
  const velocidad = distancia / tiempo;  
  console.log("La velocidad del móvil es " + velocidad);  
}
```

1. Ejecuta la función tres veces. A modo de ejemplo:

```
calcularVelocidad();  
calcularVelocidad();  
calcularVelocidad();
```

*! Recuerda que para ejecutar una función hay que usar los paréntesis*

1. Abre con live server el archivo index.html y observa la consola del navegador (debería verse tres veces el mensaje programado).
- 



### **Actividad: Función de velocidad dinámica**

En esta actividad, mejorarás la función anterior para que reciba la distancia y el tiempo como parámetros, lo que la hará más reutilizable. Sigue los siguientes pasos:

1. Desde Visual Studio Code, abre speed
2. En el archivo "index.js", modifica la función para que tome como argumentos la distancia y el tiempo, y use estos parámetros en lugar de las variables programadas. Por ejemplo:

```
function calcularVelocidad(distancia, tiempo) {  
  // const distancia = 10  
  // const tiempo = 2  
  const velocidad = distancia / tiempo;  
  console.log("La velocidad del móvil es " + velocidad);  
}
```

1. Ejecuta la función tres veces. A modo de ejemplo:

```
calcularVelocidad(10,2);  
calcularVelocidad(100,2);  
calcularVelocidad(75,3);
```

1. Abre con live server el archivo index.html y observa la consola del navegador (debería verse tres veces el mensaje programado).
- 



## **Actividad: Retorno de velocidad**

En esta actividad, mejorarás la función anterior para que devuelva la velocidad y así poder reutilizar este valor. Sigue estos pasos:

1. Desde Visual Studio Code, abre **speed**
2. En el archivo "index.js", modifica la sintaxis de la función para que sea una función flecha y utilice la palabra reservada return para devolver la velocidad. Por ejemplo:

```
const calcularVelocidad = (distancia, tiempo) => {  
  const velocidad = distancia / tiempo;  
  console.log("La velocidad del móvil es " + velocidad);  
  return velocidad  
}
```

1. Guarda los valores calculados en las variables velocidad1, velocidad2 y velocidad3.
2. Define tres mensajes con estos valores y luego imprímelos en la consola. Por ejemplo:

```
const velocidad1 = calcularVelocidad(10,2);  
console.log("La velocidad de Juana es " + velocidad1);
```

1. Abre con live server el archivo index.html y observa la consola del navegador (deberían verse los tres mensajes programados).

