

Hoy vemos **promesas**, y arrancamos con modelos **síncronos** y **asíncronos** (Sincronizado y no sincronizado).

**set Timeout()** es un modelo **síncrono** (lo que el profe hace para levantar código JS, es pararse en "salida" y ejecutar.) Lo que hace, es marcar un "lápso" de tiempo, a realizar la tarea solicitada.

**callStack()** es una lista que "imprime las cosas que se le pide"

Todas las funciones que creamos, son **síncronas**.

A diferencia de **set Timeout()**, **set Interval()** itera indefinidamente.

**clear Interval()** se usa para cortar al **set Interval()**.

---

-Intervalo de clase-

---

Las **promesas** en javascript, son eventos **asíncronos** a ejecutarse a futuro. Estas, lo que hacen es, poner al pendiente una solicitud.  
**.then** va a recibir un **array**, que va a devolver datos.

Y digamos que **.then** y **.catch** reaccionan en posde ser verdadero, o falso respectivamente.  
**.finally** se ejecutà indefinidamente de si es verdadero, o falso.

**.catch** lo que hace es recibir errores, esto cuando también devuelva falso.

La próxima clase, vamos a hacer ejercicio práctico de todo esto.

Después el profe mandó un ejercicio para resolver en clase, con 15' de hacer una función con promesas.

---

-Clase validada-

*Edmoodt*