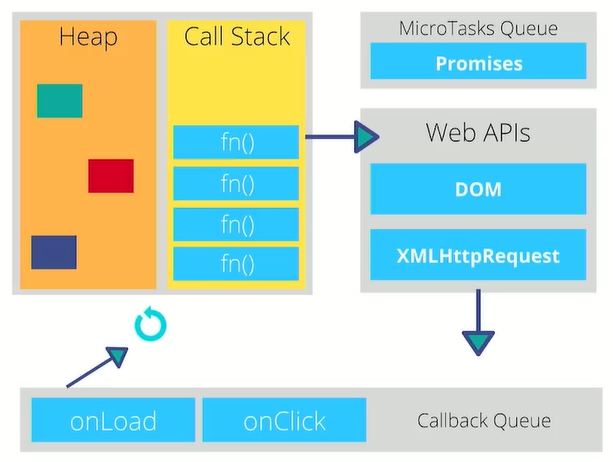
# EventLoop

El bucle de eventos es un patrón de diseño que espera y distribuye eventos o mensajes es un programa.

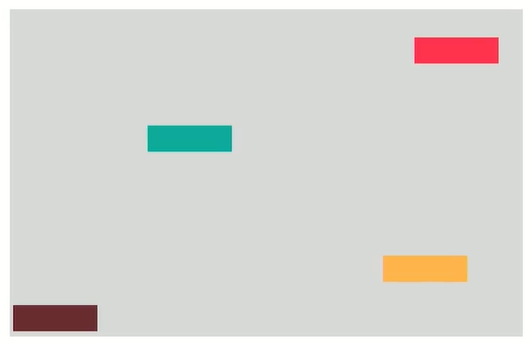
Tarea asignada para mover del Task Queue al Stack, sólo si el stack esta vacio.



[El recurso para ver el flujo](http://latentflip.com/loupe/?code=JC5vbignYnV0dG9uJywgJ2NsaWNrJywgZnVuY3Rpb24gb25DbGljaygpIHsKICAgIHNldFRpbWVvdXQoZnVuY3Rpb24gdGltZXIoKSB7CiAgICAgICAgY29uc29sZS5sb2coJ1lvdSBjbGlja2VkIHRoZSBidXR0b24hJyk7ICAgIAogICAgfSwgMjAwMCk7Cn0pOwoKY29uc29sZS5sb2coIkhpISIpOwoKc2V0VGltZW91dChmdW5jdGlvbiB0aW1lb3V0KCkgewogICAgY29uc29sZS5sb2coIkNsaWNrIHRoZSBidXR0b24hIik7Cn0sIDUwMDApOwoKY29uc29sZS5sb2coIldlbGNvbWUgdG8gbG91cGUuIik7!!!PGJ1dHRvbj5DbGljayBtZSE8L2J1dHRvbj4%3D).

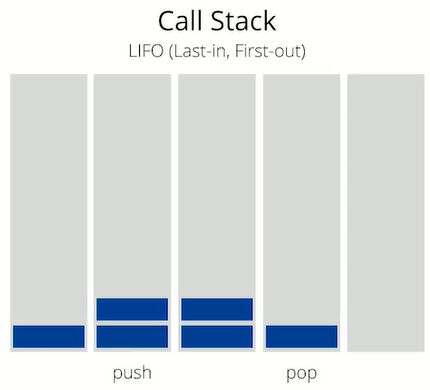
# Memory Heap

Los objetos son asignados a un montículo (espacio grande en memoria no organizado)



# Call Stack (Pila)

Apila de forma organizada las instrucciones de nuestro programa.



# Task Queue

Cola de tareas que se maneja directamente ahí la concurrencia, se agregan las tareas que ya están listas para pasar al stack (Pila).

El Stack debe estar vacío, para continuar con esta lógica.

# Micro Task Queue

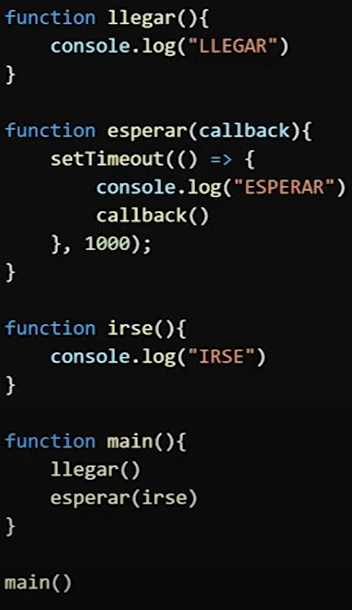
Son las promesas que tiene otra forma de ejecutarse y una prioridad superior.

# Web API

JavaScript del lado del cliente: setTimeout, XMLHttpRequest, File Reader, DOM.

# callback

Función que se pasa como argumento a otra función para que complete una serie de instrucciones.



# callbackHell

El infierno de las devoluciones de llamada. Se produce cuando se anidan muchas devoluciones de llamada.

El resultado de una, es el parámetro que necesita otra. Esta se ve como una pirámide y si una falla, todas las demás también.



# callbackHell

El infierno de las devoluciones de llamada. Se produce cuando se anidan muchas devoluciones de llamada.