JavaScript Event Loop and calls Stack
Asynchronous Operations:-

JavaScript is a Single threaded meaning it execu
tes code one line at a time in a
Single Sequence. However, it can handle
asynchronous operations efficiently using
it's call stack and event loop.

1. what Is the call Stack and How
Does It Operate ?

-The call stack is a data Structure where
JavaScript keeps track of function calls
it operates on a last. in, first out
(LIFO) principle

- when a function is called - it gets
pushed onto the Stack
- when a function complete it is
removed from the stack
- The stack executes Synchronous code
one step at a time.

2. what is the Event Loop and its Role in Asynchronous Processing ?

The event loop ensures JavaScript handles asynchronous code without blocking the main thread

3. How Do setTimer and Promises Fit into the Event Loop

SetTimeOut
The timer starts in the Web APIs environment
After the Specified time, its callback is moved to the task queue
The event loop waits until the call stack is empty before moving the callback from the task queue to the call Stack.

Promises
Promises use the micro task queue which has higher priority than the task queue

Once a promise resolves or reject its then or catch call backs is queued in the micro task queue
The event loop processes all micro tasks before tasks in the task queue

4 How SetTimeout and Promises Re. Enter the call Stack

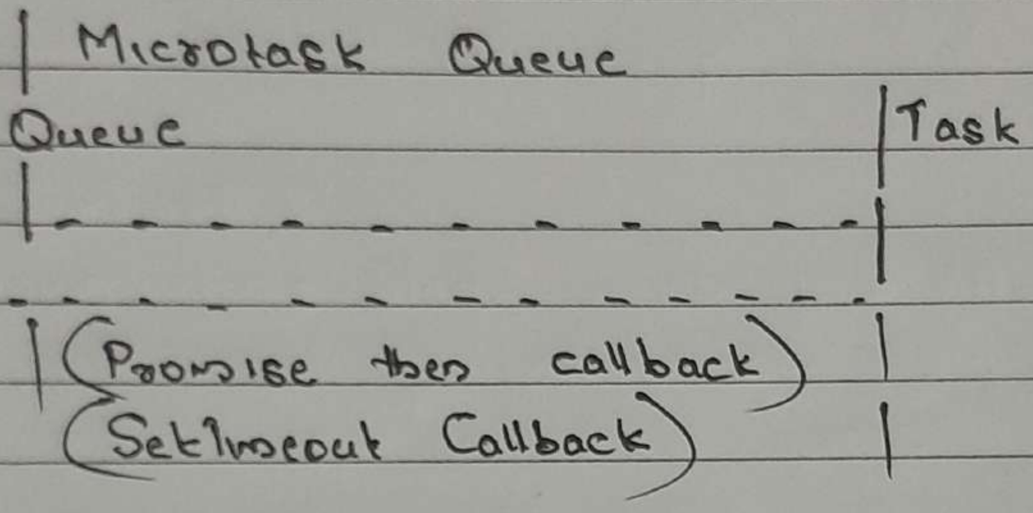
when JavaScript encounters setTimeout or a promise
1. SetTimeout
The timer begins in the Web APIs environment
Once the timer expires its callback moves to the task queue
The event loop waits for the call stack to the empty then moves the call back to the Stack for execution

Promises
when a promise resolve or reject the then or catch callback is placed in the microstack queue

The event loop processess all micro tasks immediately after the current operation before moving to tasks in the task queue.

call stack : [Empty]

queue Management:



1. Stack processess synchronous tasks
2. Microtask Queue executes resolved promises
3. Task Queue executes 'setTimeout' callbacks.