What is node js ?

Node js is a platform that runs javascript code outside the browser. It runs in v8 which is an engine written in c++. Node js has a single threaded event loop so for this issue it uses a non blocking i/o to prevent the need for multi threading since the server can get queued many requests at the same time which makes it concurrent. Node js uses an event-driven programming that makes it efficient,this one follows publisher subscriber pattern.

What is event loop in Node?

┌───────────────────────────┐

┌─>│ timers │

│ └─────────────┬─────────────┘

│ ┌─────────────┴─────────────┐

│ │ pending callbacks │

│ └─────────────┬─────────────┘

│ ┌─────────────┴─────────────┐

│ │ idle, prepare │

│ └─────────────┬─────────────┘ ┌───────────────┐

│ ┌─────────────┴─────────────┐ │ incoming: │

│ │ poll │<─────┤ connections, │

│ └─────────────┬─────────────┘ │ data, etc. │

│ ┌─────────────┴─────────────┐ └───────────────┘

│ │ check │

│ └─────────────┬─────────────┘

│ ┌─────────────┴─────────────┐

└──┤ close callbacks │

└───────────────────────────┘

Timers- this section runs callbacks scheduled by the setTimeOut() or setinterval().

Pending callbacks: this section runs callbacks for some system operations like tcp errors.EX: IF Tcp socket receives an error while is attempting to connect some system have to wait to send this error.So those errors will be queued in this section.

Poll: the main funcs of pull are :

1->calculate the time of blocking and polling for input and output

2-> to process the events in queue

We have 2 cases :

If poll is not empty: the event loop will iterate through its queue of callbacks and will execute them synch until the time limit is reached.

If pull is empty: when pull queue is empty the event loop is gonna check whose time have been reached .If one or more are ready the event loop will go back to the timer section to execute callbacks.

Check: This section execute callbacks after the pull section is done .if the pull has been queued with setImmediate the event loop will continue to check rather than waiting.

Close callbacks:if a socket or handle is closed event will b e emmited in this section otherwise it goes to nextticket()🡪 which is not part of event loop .

Based on what we learned so far what are the differences between AngularJS and Angular?

**Based on Architecture:**

Angular uses components including structural directives .

Angular.js uses model view controller .

**Dependecy injection:**

Angular use a bunch of dependencies to increase the performace.

Angular js doesn’t have dependencies its uses directives.

**Routing:**

AngularJs uses router to init the routing info.

Angular uses a URL to imitate a directive to reach the client-view.

**Testing:**

AngularJs uses third party to test the erros

Angular: has command line interface for project building which reduces the time.