**Solution**

1. The order of execution between setTimeout(callback,0) and setImmediate(callback) will vary depending on the context in which they are called.
   1. When used in script without I/O circle, the order of execution of these two timers is non-deterministic, as it is bound by the performance of the process.
   2. When used within an I/O cycle, then setImmediate(callback) will always execute first.
2. - The difference between process.nextTick() and setImmediate:
   1. Process.nextTick() is not part of the event loop, while setImmediate() is part of the event loop.
   2. The callback of process.nextTick() is registered in a priority queue ”nextTick” queue. Node processes all the callbacks in the nextTick queue after current operation completes and before the event loop continues, which means it runs before any additional I/O events or timers fire in subsequent tick of the event loop.
   3. setImmediate will schedule a callback to run at check phase of the event loop after I/O events callcack.

* process.nextTick will always run before setImmediate, but when the event loop is in check pahse the setImmediate timer will run first

1. core modules that Node provides by default includes:
2. https
3. url
4. http
5. net
6. V8
7. Path
8. Util
9. Fs
10. Timers
11. stream