1. Перечислите основные глобальные объекты Node.js и поясните их назначение.

global: Глобальный объект, предоставляющий доступ к глобальным переменным и функциям в приложении Node.js. Все переменные и функции, объявленные без ключевого слова var, let, или const, автоматически становятся членами объекта global.

process: Этот объект предоставляет информацию и управление текущим процессом Node.js. Он содержит множество полезных свойств и методов, таких как process.env (для доступа к переменным окружения), process.argv (для аргументов командной строки), и process.exit() (для завершения процесса).

console: Глобальный объект, предоставляющий функции для вывода сообщений в консоль. Это основной способ отладки и записи информации в Node.js приложениях.

Buffer: Глобальный объект для работы с бинарными данными. Он используется для создания, чтения и записи бинарных данных, таких как байтовые массивы.

require(): Эта функция используется для подключения модулей в Node.js. Она позволяет импортировать функциональность других файлов и библиотек в ваше приложение.

1. Поясните понятие «асинхронная функция».

Асинхронная функция – это функция, которая выполняется асинхронно, то есть не блокирует выполнение остального кода и может завершиться в будущем. Она инициирует выполнение некоторой задачи и возвращает управление сразу, вместо ожидания завершения этой задачи. Вместо блокировки исполнения, асинхронные функции используют колбэки, промисы или асинхронные/ожидаемые ключевые слова (async/await) для управления последовательностью выполнения.

1. Поясните понятие «стандартные системные потоки».

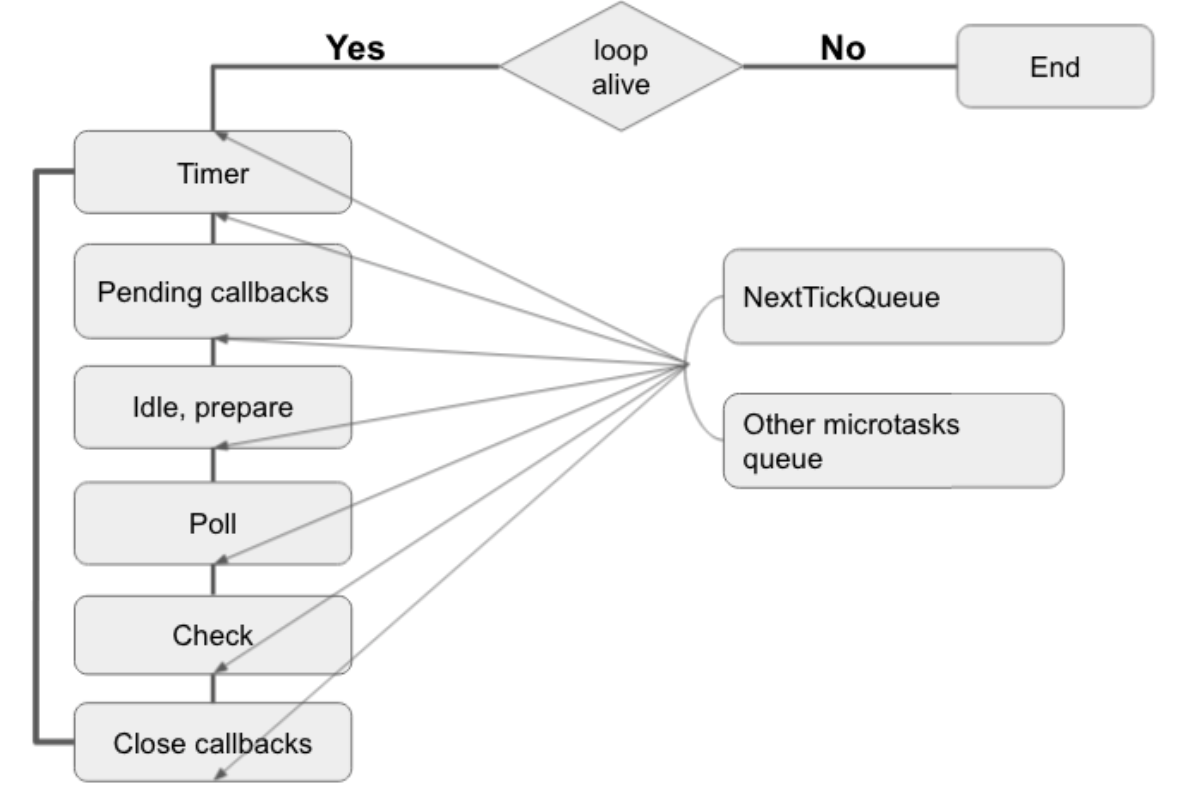
Стандартные системные потоки - это основные потока ввода-вывода, предоставляемые операционной системой для взаимодействия с программами:

stdin (стандартный поток ввода): Этот поток предназначен для приема ввода от пользователя или других программ. В Node.js, он доступен через объект process.stdin.

stdout (стандартный поток вывода): Этот поток используется для вывода данных и сообщений программы. В Node.js, он доступен через объект process.stdout.

stderr (стандартный поток ошибок): Этот поток используется для вывода ошибок и диагностических сообщений. В Node.js, он доступен через объект process.stderr.

1. Поясните назначение функций **process.nextTick**, **setImmediate**. Поясните в чем их разница.



Функции process.nextTick и setImmediate используются в Node.js для выполнения асинхронного кода. Они имеют следующие различия:

process.nextTick: Эта функция позволяет выполнить функцию или колбэк как можно скорее после завершения текущей операции Event Loop, но до выполнения других асинхронных операций, таких как таймеры и промисы. Функции, переданные process.nextTick, имеют приоритет над setImmediate.

setImmediate: Эта функция планирует выполнение функции или колбэка после завершения текущей операции Event Loop, но перед началом следующей итерации Event Loop. Она предпочтительна, когда вам нужно выполнить асинхронный код, который не должен блокировать Event Loop, и который не требует немедленного выполнения.

Макрозадачи — выполняются по одной за один проход цикла, микрозадачи — на каждом проходе цикла выполняет все накопившиеся