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

- `process`: объект, предоставляющий информацию о текущем процессе Node.js и управляющий им, такие как аргументы командной строки, окружение, директория выполнения и т.д.

- `console`: объект, предоставляющий методы для вывода сообщений в консоль.

- `global`: объект, представляющий глобальное пространство имен, в котором могут быть определены глобальные переменные и функции.

- `setTimeout` и `setInterval`: функции, используемые для планирования вызовов функций через определенное время или периодически.

- `require`: функция, используемая для подключения модулей в Node.js.

1. Поясните понятие «асинхронная функция».

Асинхронная функция в Node.js - это функция, которая выполняется в фоновом режиме, без блокирования основного потока выполнения. Вместо того, чтобы ждать завершения операции, она запускается, а затем продолжает выполнение кода без ожидания результата. По завершении асинхронной операции, обычно вызывается `callback` функция или используется другой способ обработки результата.

1. Поясните понятие «стандартные системные потоки».

Стандартные системные потоки - это предопределенные потоки ввода-вывода, которые автоматически создаются при запуске программы. В Node.js они представлены следующими объектами:

- `process.stdin`: поток ввода, используемый для чтения данных в программу.

- `process.stdout`: поток вывода, используемый для записи данных из программы.

- `process.stderr`: поток вывода для записи сообщений об ошибках.

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

Функция `process.nextTick` используется для планирования выполнения коллбэк функций в самом начале следующей итерации цикла событий. Она помещает коллбэк в конец очереди задач на выполнение, что позволяет ей быть выполненной до любых других функций или операций.

Функция `setImmediate` также используется для планирования выполнения коллбэк функций, но она помещает коллбэк в другую очередь задач на выполнение, называемую очередью проверки приоритетов. Коллбэк, запланированный с помощью `setImmediate`, будет выполнен после всех коллбэк функций, запланированных с помощью `process.nextTick`, но до других функций и операций в цикле событий. **И `setImmediate` можно отменить, а process.nextTick НЕТ.**

**1)таймеры**

**2)idle processes (коллбэки системных вызовов)**

**3)стадия padding callbacks**

**4)poll (ассинхронный чтение-вывод)**

**5)check**

**6)exit**

**7)**