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

Основные глобальные объекты Node.js и их назначение:

* **console**: Объект для вывода сообщений в консоль. Он предоставляет методы, такие как **console.log**, **console.error**, и другие, для вывода информации, отладки и обработки ошибок.
* **global**: Это глобальный объект, предоставляющий доступ к другим глобальным объектам и переменным в Node.js. Он может использоваться для определения глобальных переменных, но рекомендуется избегать его использования.
* **process**: Объект, предоставляющий информацию и управление текущим процессом Node.js. Он содержит информацию о аргументах командной строки, окружении, стандартных потоках ввода/вывода и другие методы для работы с процессом.
* **Buffer**: Глобальный объект для работы с бинарными данными. Он предоставляет методы для создания, изменения и чтения буферов, которые могут использоваться для манипуляции данными, например, при работе с файлами и сетевыми запросами.
* **require**: Глобальная функция, используемая для подключения модулей в Node.js. Она позволяет импортировать функциональность из других модулей и использовать их в текущем модуле.
* **module**: Глобальный объект, который представляет текущий модуль Node.js. Он содержит информацию о текущем модуле, его экспортах и может быть использован для создания собственных модулей

1. Поясните понятие «асинхронная функция».

**Асинхронная функция** - это функция, которая выполняется в асинхронном (несинхронном) режиме. Она не блокирует выполнение программы и позволяет продолжать выполнение других задач, в то время как она сама выполняется. Асинхронные функции часто используются в Node.js для выполнения ввода/вывода (I/O) и других операций, которые могут занимать много времени, без блокировки работы приложения

1. Поясните понятие «стандартные системные потоки».

Стандартные системные потоки - это потоки ввода/вывода (I/O), которые предоставляются операционной системой для взаимодействия между программой и окружающей средой. В Node.js три основных стандартных системных потока:

stdin: Стандартный поток ввода. Он предназначен для приема данных с клавиатуры или других источников ввода.

stdout: Стандартный поток вывода. Он используется для вывода данных программы, обычно на экран (консоль).

stderr: Стандартный поток ошибок. Он используется для вывода сообщений об ошибках и другой информации об ошибках

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

process.nextTick и setImmediate - это механизмы для выполнения функций асинхронно, но с разным приоритетом:

process.nextTick: Функции, переданные в process.nextTick, выполняются в первую очередь, перед другими асинхронными операциями, такими как I/O. Этот механизм полезен, когда нужно убедиться, что определенный код будет выполнен максимально быстро.

setImmediate: Функции, переданные в setImmediate, выполняются после выполнения текущей итерации цикла событий. Этот механизм полезен, когда нужно выполнить код после завершения текущей "порции" асинхронных операций.

Разница между ними заключается в том, когда они выполняются относительно текущей очереди событий в цикле событий Node.js. process.nextTick выполняется перед другими операциями, а setImmediate - после текущей "порции"