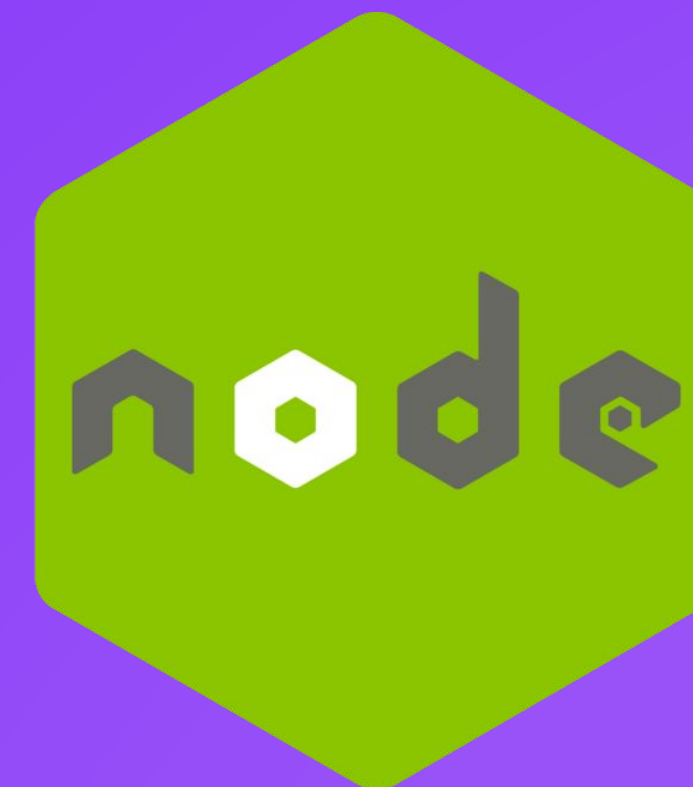




УРОК №1

Введение в NodeJS. Создание локальных приложений с помощью NodeJS



Что такое node.js

Node.js

Программная платформа, основанная на движке V8 (транслирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения.

Что сделано на node.js



PayPal



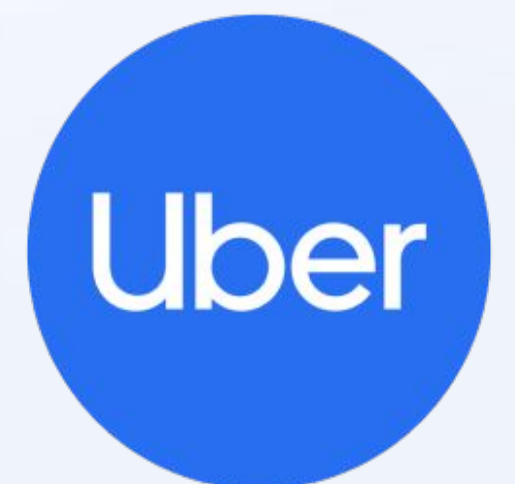
LinkedIn



Netflix



eBay



Uber

Цитата

Как асинхронное событийное JavaScript - окружение, Node.js спроектирован для построения масштабируемых сетевых приложений. Для каждого соединения вызывается функция обратного вызова, однако, когда соединений нет Node.js засыпает.

Установка node.js

<https://nodejs.org/ru/>

Ссылка на установку.

Вместе со средой node.js становится доступным пакетный менеджер npm

Интерактивная оболочка Node.js

Интерактивная оболочка (или REPL) позволяет выполнять код в терминале. Для запуска REPL необходимо в консоли вписать команду `node`.

Для того, чтобы выйти из интерактивной оболочки, необходимо дважды нажать `Ctrl + C`.

Начало проекта

Создайте пустую папку, внутри неё создайте файл `index.js`.

Откройте папку с помощью редактора Visual Studio Code.

(можно и через любой другой редактор, но это удобнее)

Первое приложение на node.js

```
//код файла index.js  
console.log("Hello  
world")
```

```
//Терминал  
node index.js
```

Структура node.js

Node.js работает на модулях. Для установки модулей используется функция `require`

```
var http = require('http');
```

```
var server = http.createServer(function(req, res) {  
  res.writeHead(200);  
  res.end('Hello Http');  
});
```

```
server.listen(8080);
```

Предустановленные модули node.js

`dns` - работа с доменными именами

`fs` - работа с файловой системой

`http`, `https` - работа с протоколами `http` или `https`

`os` - работа с операционной системой

и другие.

Для установки других модулей используется пакетный менеджер `npm`.

Создание модулей

Модули можно создавать автоматически. При подключении нужно только указать имя вашего файла.

Создайте папку “app”, а внутри файл “test.js”

Поместите в файл test.js следующий код.

```
console.log("test");
```

Подключите файл test.js с помощью функции require

```
let test = require("./app/test");
```

Код выполнится при подключении файла.

Создание модулей

Объект 'exports' доступен в любом модуле и возвращается при каждом вызове функции require при подключении модуля.

```
exports.say = function(msg) {  
  console.log(msg + "!");  
}
```

Событийная система

node.js работает на системе событий - как браузерный JS.

Для создания событий используется специальный класс EventEmitter.

Создание события

```
let Emitter = require('events')
let myEmitter = new Emitter();
myEmitter.on("loaded", function(){
  console.log("Event!")
})
myEmitter.emit("loaded");
```

Задания

1. Модуль для вычисления площадей разных фигур.
2. Модуль для поиска ключевых слов в тексте.
3. Модуль для вывода в консоли кода HTML на основе полученной информации.
4. Модуль для построения ASCII-фигур из перечня.
5. Модуль для разбора чисел (простое или нет, четное, вывод делителей и т.д.)

Конец

Давайте подведем итоги урока!
Чему мы научились?
Что мы использовали?
К чему мы пришли?