



**Microsoft** Partner  
Silver Learning



Введение в Node.js



ITVVDN  
IT VIDEO DEVELOPERS NETWORK

# Node.js

## Автор курса



Владимир Виноградов



MCID: 9210561

# Node.js

После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство вашего учебного центра



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

# Node.js

Тема

## Введение в Node.js

# Node.js

## Node.js

**Node.js** — это серверная JavaScript платформа, предназначенная для создания масштабируемых распределенных сетевых приложений. В основу положена событийно-ориентированная архитектура и асинхронное взаимодействие с неблокирующим вводом/выводом. Основана на JavaScript движке V8 (транслирующем JavaScript в машинный код), от компании Google.

**Node.js** добавляет возможность JavaScript взаимодействовать с устройствами ввода-вывода через свой API (написанный на C++), подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript-кода.



# Node.js

## История Node.js



В 2009 году, в ходе исследований по созданию событийно-ориентированных серверных систем, **Райаном Далом**, была создана платформа Node.js. В ходе своих исследований он пришёл к выводу, что вместо традиционной модели параллелизма на основе потоков следует обратиться к событийно-ориентированным системам.

Асинхронная модель была выбрана из-за простоты, низких накладных расходов (по сравнению с идеологией «один поток на каждое соединение») и высокого быстродействия. Целью **Node.js** является предложить «простой способ построения масштабируемых сетевых серверов».

Разработка **Node.js** спонсируется компанией Joyent, основанной Райаном.

# V8

## Движок V8

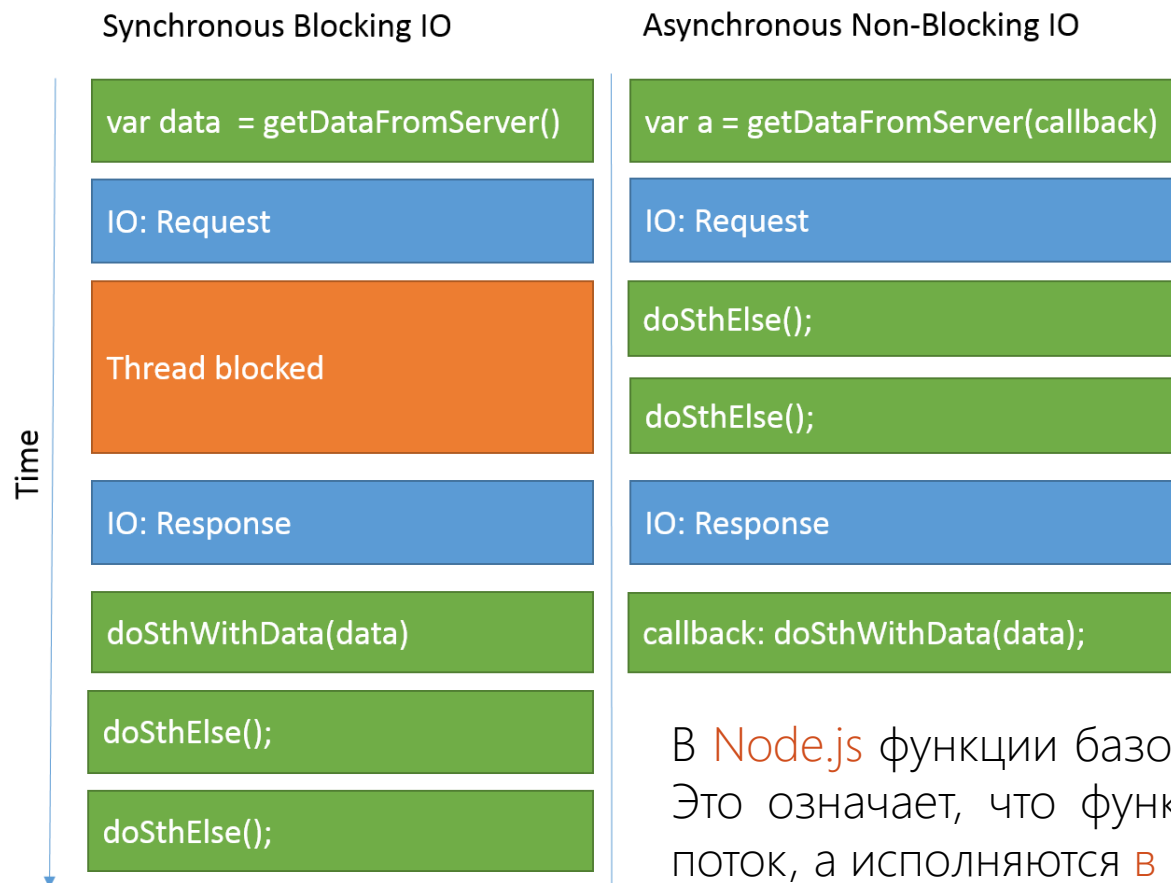
**V8** - движок JavaScript с открытым исходным кодом, разрабатываемый датским отделением компании Google. Он написан на C++ и используется в Google Chrome. Движок может работать автономно или быть установлен в любое C++ приложение.

Несмотря на динамическую природу JavaScript, разработчикам удалось применить методы, характерные для реализации классических объектно-ориентированных языков, такие как компиляция кода «на лету», внутреннее кэширование, точный процесс сборки мусора, снапшоттинг при создании контекстов.

Движок V8 отличается от других движков (JScript, SpiderMonkey, JavaScriptCore, Nitro) высокой производительностью.

# Node.js

## Преимущества Node.js



В Node.js функции базовых модулей **асинхронны**. Это означает, что функции ноды не блокируют поток, а исполняются в **фоновом режиме**.



# Node.js

## Преимущества Node.js

- Одни и те же программисты могут работать над обеими сторонами приложения;
- Код проще переносить с сервера на клиент и обратно;
- Общий для клиента и сервера формат данных (JSON);
- Общий программный инструментарий;
- Общие для клиента и сервера средства тестирования и контроля качества;
- На обеих сторонах можно использовать общие шаблоны представлений;
- Общий язык общения между группами, работающими над клиентской и серверной частью.

# Module Node.js

## Модули Node.js

**Модули** – это базовые строительные блоки для создания приложений на платформе Node.js. Любой JavaScript файл, используемый в Node является модулем.

Преимущества модульной системы:

1. Структурированность программных частей;
2. Модули похожи на пространства имен;
3. Взаимозаменяемость;
4. Упрощает взаимодействие пользователя с разными частями;
5. Вместе с **Node.js** поставляется несколько встроенных модулей.

# Module Node.js

## Модули Node.js

Подключение модуля происходит с помощью вызова функции `require`, которой нужно передать путь к файлу. При повторном подключении Node.js проверяет, подключался ли модуль ранее, если это так, то возвращается ссылка на уже существующий модуль.

Подключение модуля:

```
var module = require('имя_файла')
```



**Встроенные модули имеют приоритет над всеми остальными.**

# Module Node.js

## Модули Node.js

Для создания нового модуля необходимо создать файл с расширением \*.js и реализовать в нем логику. Чтобы функционал Вашего модуля был доступен извне, необходимые члены модуля требуется поместить в объект **exports**.

Содержимое запускаемого файла

```
var obj = require('./mod');  
  
console.log(obj.number);  
obj.func();
```

Содержимое модуля

```
var x = 10;  
  
function test(){  
    console.log('Test function, number = ', x);  
};  
exports.number = x;  
exports.func = test;
```

# Module Node.js

## Модули Node.js

Свойства объекта `module`:

1. `id` – путь к исполняемому файлу
2. `exports` – объект который возвращается функцией `require`
3. `parent` – ссылка на родительский модуль
4. `filename` – абсолютный путь к файлу
5. `loaded` – статус обработки файла
6. `children` – массив дочерних модулей
7. `paths` – массив путей, по которым происходит поиск подключаемых модулей

# Module Node.js

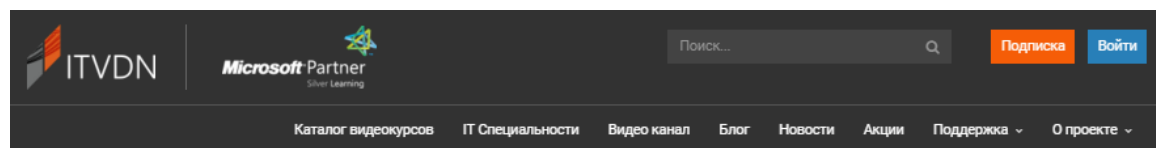
## Модули Node.js

Алгоритм поиска подключаемых модулей:

1. Если модуль является системным, происходит его загрузка
2. Если 'имя\_модуля' начинается со знака '/' - поиск модуля будет производиться в корне файловой системы
3. Если 'имя\_модуля' начинается со знака '/', './', '../' - поиск осуществляется несколькими способами:
  - 3.1 Происходит поиск файла 'имя\_модуля' по указанному пути
  - 3.2 Происходит поиск директории имя которой 'имя\_модуля' по указанному пути
4. Поиск папки *node\_modules*
  - 4.1 Поиск модуля происходит в директории *node\_modules*
  - 4.2 Если в текущем каталоге нет каталога *node\_modules*, происходит переход в каталог уровнем выше, и производится поиск в нее
5. Ошибка, модуль не найден

# Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.

### 1 ITVDN 2015. Наши награды

### ITVDN 2015. Итоги года

В марте 2015 года ITVDN стал победителем конкурса IT Education Awards, который проходил в рамках IT Jam 2015 и награжден как лучший образовательный ресурс в номинации Online Education. Экспертное жюри, в состав которого вошли представители ведущих IT компаний, отметили такие преимущества ITVDN, как системный подход в обучении, позволяющий удаленно получить качественное образование по наиболее популярным специальностям, высокий профессионализм авторов видео курсов и использование современных методик оценки знаний.

Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics

## Новые видео

Исключения	0
Итераторы и генераторы	0

## Популярные видео курсы

Видео курс C# Стартовый (для начинающих)	9 уроков (16 ч. 3 мин.)
Видео курс по шаблонам проектирования	29 уроков (16 ч. 7 мин.)

## Теги

.NET Developer
Frontend Developer



# Проверка знаний

## TestProvider.com

TestProvider | Мы помогаем людям оценить себя

Регистрация | Войти

Главная | Каталог | Сертификация Microsoft | Поддержка | О нас

Тестирование

Языки программирования и информационные технологии

Microsoft

C# ASP.NET MVC JavaScript Patterns Of Design SQL Architecture Guide WCF HTML&CSS XML SEO WPF HTML5&CSS3 JQuery XNA SharePoint GUI for Android Windows Azure Platform Microsoft Patterns&Practices TFS SCRUM ReSharper TDD WWF LINQ Entity Framework Windows Forms Refactoring Microsoft Expression Blend 4 Windows Phone 8 Windows 8 AppStore Visual Studio Tips&Tricks MSF MEF SilverLight AJAX MEF Service Oriented Architecture

Пройти тест

Наши партнеры

Microsoft Partner CyberBionic ITVDN PROMETRIC TEST CENTER PEARSON VUE Authorized Test Center Windows Azure Cloud Partner EBA

Дополнительные ресурсы:

Очное обучение On-line обучение Видео обучение

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.





## Q&A

# Информационный видеосервис для разработчиков программного обеспечения

