

Базовые модули Node.js

№ урока: 3 Курс: Node.js

Средства обучения: Персональный компьютер с установленной Visual Studio Code, и Node.js.

Обзор, цель и назначение урока

В этом уроке мы узнаем, какие глобальные переменные существуют в Node.js, рассмотрим работу модуля Path, который позволяет легко и просто работать с путями файловой системы. Научимся работать с NPM, выгружать свои модули и загружать сторонние, также рассмотрим такие модули как: util, console, buffer, file system, url, timers

Изучив материал данного занятия, учащийся сможет:

- Понимать суть глобальных переменных
- Использовать модуль util
- Использовать модуль console
- Использовать модуль buffer
- Использовать модуль fs
- Использовать модуль url
- Использовать модуль timers
- Открывать файлы и считывать из них данные
- Создавать и работать с таймерами
- Загружать свои модули в npm
- Скачивать модули сторонних разработчиков из npm
- Работать в интерактивном режиме в Node.js

Содержание урока

1. Рассмотрение глобальных переменных в Node.js
2. Рассмотрение основных модулей: util, console, buffer, fs (file system), url, timers, path
3. Знакомство с NPM
4. Создание своего модуля и загрузка в систему npm
5. Основные команды для работы с npm
6. Работа с интерактивной оболочкой REPL

Резюме

- **__dirname** – глобальная переменная в каждом модуле. Хранит путь в виде строки к директории в которой находится файл.
- **__filename** – хранит строковое представление полного пути к модулю, в котором происходит обращение к этой переменной.
- Модуль **console** представляет собой простую консоль, для компиляции. Содержит методы **console.log()**, **console.error()**, **console.warn()**. Могут выводить сообщения в разные потоки.
- Модуль **util** – в первую очередь, модуль предназначен для поддержки собственных внутренних API-интерфейсов в Node.js
- Модуль **buffer** – используется для представления последовательности кодированных символов, которые можно конвертировать между буферами и обычными строками объектами JavaScript при помощи определенного метода кодирования.
- Модуль **fs** – позволяет производить операции с файлами в файловой системе. Все методы этого модуля имеют как асинхронную, так и синхронную формы.
- Модуль **url** – предоставляет утилиты для работы с url маршрутами. Также поддерживает методы для «парсинга» путей и сборки путей из отдельных значений.

- **Timers** – модуль предоставляет API-интерфейс для функций планирования, которые будут вызываться в какой-то период времени.
- Модуль timers можно не подключать, поскольку таймеры являются глобальными для всех модулей.
- **Path** – модуль, предоставляющий утилиты для работы с путями к файлам и директориям в файловой системе.
- **NPM** – менеджер пакетов, входящий в состав Node.js.
- Установка пакета производится при помощи команды: `npm install <package_name>`.
- Поиск нужного пакета производится командой: `npm search <package_name>`.
- Real eval print loop (**REPL**) – представляет собой реализацию цикла Чтение-Вычисление-Печать-Цикл. Позволяет пользователям вводить строки с инструкциями и сразу же выдавать результат выполнения инструкций.
- Основные команды REPL:
 - `.break` – прерывание последующего ввода многострочной команды;
 - `.clear` – сбрасывает весь доступный контекст;
 - `.exit` – завершение работы в интерактивном режиме;
 - `.help` – список специальных команд;
 - `.save` – сохраняет текущую сессию ;
 - `.load` – загружает файл в текущую сессию;
 - `.editor` - ходит в режим редактирования.

Закрепление материала

- Что такое глобальная переменная?
- Для чего предназначен модуль util?
- Для чего предназначен модуль console?
- Для чего предназначен модуль buffer?
- Для чего предназначен модуль fs?
- Для чего предназначен модуль url?
- Для чего предназначен модуль timers?
- Для чего предназначен модуль path?
- Для чего использовать NPM?
- Что такое REPL?

Дополнительное задание

Задание

Подключите модуль path, изучите методы этого модуля. Поработайте с маршрутами

Самостоятельная деятельность учащегося

Задание №1

Выучите основные понятия, рассмотренные на уроке.

Задание №2

Создайте приложение, реализуйте возможность чтения и записи файлов.

Задание №3

Создайте два приложения. В первом создайте свой модуль, который может считывать файл. Загрузите его в NPM. Во втором модуле найдите свой загруженный модуль, скачайте его и подключите. Используя данный модуль считайте содержимое какого-либо файла и содержимое выведите в консоль.

Рекомендуемые ресурсы

Модуль util:

<https://nodejs.org/dist/latest-v6.x/docs/api/util.html>

Модуль console:

<https://nodejs.org/dist/latest-v6.x/docs/api/console.html>

Модуль buffer:

<https://nodejs.org/dist/latest-v6.x/docs/api/buffer.html>

Модуль fs:

<https://nodejs.org/dist/latest-v6.x/docs/api/fs.html>

Модуль url:

<https://nodejs.org/dist/latest-v6.x/docs/api/url.html>

Модуль timers:

<https://nodejs.org/dist/latest-v6.x/docs/api/timers.html>

Модуль path:

<https://nodejs.org/dist/latest-v6.x/docs/api/path.html>

NPM:

<https://www.npmjs.com/>

REPL:

<https://nodejs.org/dist/latest-v6.x/docs/api/repl.html>