

# Инструменты

Frontend - разработчик

# Содержание

- node.js
- npm, package.json
- yarn
- http-server

# Node.js

Понимание того, что такое Node.js поможет вам лучше разобраться с npm. В двух словах — Node.js это интерпретатор языка JavaScript. Сам по себе Node.js является C++ приложением, которое получает на входе JavaScript-код и выполняет его.


## Задание:

<http://learn.javascript.ru/screencast/nodejs>

### Часть 1: Изучаем Node.JS

Выпуски были записаны для Node 0.10.

Каждую запись можно просмотреть или скачать в низком и хорошем качестве.

1. Введение в Node.JS, об этом скринкасте	
2. Что такое Node.JS? Почему Node.JS?	
3. Установка и запуск	
4. Исходники и документация	
5. Модули для Node.JS	
6. Приёмы работы с модулями	
7. Введение в NPM - менеджер пакетов для Node.JS	
8. Структура пакета NPM	
9. Глобальные модули	
10. Модуль util и наследование	

# npm, package.json

## Пакеты в Node.js

Вкратце, пакетом в Node.js называется один или несколько JavaScript-файлов, представляющих собой какую-то библиотеку или инструмент.

npm (аббр. node package manager) — это стандартный менеджер пакетов, автоматически устанавливающийся вместе с Node.js. Он используется для скачивания пакетов из облачного сервера npm, либо для загрузки пакетов на эти сервера.

## Файл package.json

Файл package.json содержит в себе информацию о вашем приложении: название, версия, зависимости и тому подобное. Любая директория, в которой есть этот файл, интерпретируется как Node.js-пакет, даже если вы не собираетесь публиковать его.

Способ использования файла package.json зависит от того, собираетесь ли вы скачивать пакет или публиковать его.

```
{
  "name": "lnx-riskit",
  "version": "0.0.0",
  "license": "MIT",
  "angular-cli": {},
  "scripts": {
    "ng": "ng",
    "start": "http-server dist/",
    "test": "ng test",
    "pre2e": "webdriver-manager update --standalone false --gecko false",
    "e2e": "protractor",
    "postinstall": "ng build --aot --prod"
  },
  "private": true,
  "dependencies": {
    "@angular/common": "^2.3.1",
    "@angular/compiler": "^2.3.1",
    "@angular/compiler-cli": "^2.3.1",
    "@angular/core": "^2.3.1",
    "@angular/forms": "^2.3.1",
    "@angular/http": "^2.3.1",
    "@angular/platform-browser": "^2.3.1",
    "@angular/platform-browser-dynamic": "^2.3.1",
    "@angular/router": "^3.3.1",
    "angular-cli": "^1.0.0-beta.28.3",
    "bootstrap": "^4.0.0-alpha.6",
    "core-js": "^2.4.1",
    "csshake": "^1.5.2",
    "font-awesome": "^4.7.0",
    "http-server": "^0.10.0",
    "rxjs": "^5.0.1",
    "ts-helpers": "^1.1.1",
    "zone.js": "^0.7.2"
  },
  "devDependencies": {
    "@types/jasmine": "2.5.38",
    "@types/node": "^6.0.42",
    "codeylzer": "^2.0.0-beta.1",
    "jasmine-core": "2.5.2",
    "jasmine-spec-reporter": "2.5.0",
    "karma": "1.2.0",
    "karma-chrome-launcher": "^2.0.0",
    "karma-cli": "^1.0.1",
    "karma-jasmine": "^1.0.2",
    "karma-remap-istanbul": "^0.2.1",
    "protractor": "^4.0.13",
    "ts-node": "1.2.1",
    "tslint": "^4.3.0",
    "typescript": "^2.0.3"
  }
}
```

## Npm

**Задание:** <https://docs.npmjs.com/getting-started/what-is-npm>

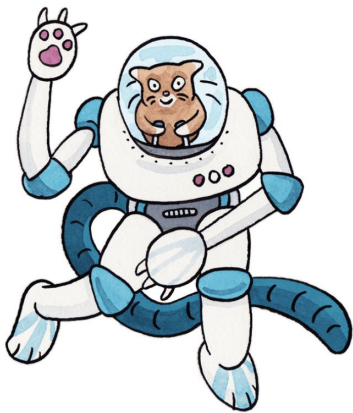
```
$ npm install canvas-chart
```

```
{  
  "devDependencies": {  
    "canvas-chart": "~1.3.0"  
  }  
}
```

# Yarn

## Ultra Fast.

Yarn caches every package it downloads so it never needs to download it again. It also parallelizes operations to maximize resource utilization so install times are faster than ever.



## Mega Secure.

Yarn uses checksums to verify the integrity of every installed package before its code is executed.





## Задание: <https://yarnpkg.com/en/docs/getting-started>

### Starting a new project

```
yarn init
```

### Adding a dependency

```
yarn add [package]
yarn add [package]@[version]
yarn add [package]@[tag]
```

### Adding a dependency to different categories of dependencies

Add to `devDependencies`, `peerDependencies`, and `optionalDependencies` respectively:

```
yarn add [package] --dev
yarn add [package] --peer
yarn add [package] --optional
```

### Upgrading a dependency

```
yarn upgrade [package]
yarn upgrade [package]@[version]
yarn upgrade [package]@[tag]
```

### Removing a dependency

```
yarn remove [package]
```

### Installing all the dependencies of project

```
yarn
```

or

```
yarn install
```

# http-server: a command-line http server

**Задание:** <https://github.com/indexzero/http-server>

## Usage:

---

```
http-server [path] [options]
```

Checkout this repository locally, then:

```
$ npm i  
$ node bin/http-server
```

Now you can visit <http://localhost:8080> to view your server

# Вопросы

---



**Dmitry Anikin**

CTO of Roonyx

**E-mail:** [dima@roonyx.tech](mailto:dima@roonyx.tech)

**GitHub:** <https://github.com/d-anikin>