

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бэк-энд разработка

Отчет

Практическая работа №1

“Основы работы с npm, node.js”

Выполнил:

Екушев Владислав

К33402

Проверил:

Добряков Д. И.

Санкт-Петербург

2022 г.

## Задача

1. Установка npm (10.2.4), node.js (18.19.1 / 20.11.1)
2. Инициализация npm-пакета
3. Установка express
4. Удаление express/попытка его обновить
5. Написание кастомной команды для npm, чтобы можно удалить express командой “npm run rme”

## Ход работы

Установим Node.JS вместе с NPM через менеджер “nvm” – инструмент для установки нескольких версий Node и быстрого переключения между ними.

Установка производится по инструкции из официального репозитория:

<https://github.com/nvm-sh/nvm>

### Installing and Updating

#### Install & Update Script

To install or update nvm, you should run the [install script](#). To do that, you may either download and run the script manually, or use the following cURL or Wget command:

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.7/install.sh | bash
```

```
wget -qO- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.7/install.sh | bash
```

Running either of the above commands downloads a script and runs it. The script clones the nvm repository to `~/.nvm`, and attempts to add the source lines from the snippet below to the correct profile file (`~/.bash_profile`, `~/.zshrc`, `~/.profile`, or `~/.bashrc`).

```
export NVM_DIR="$([ -z "${XDG_CONFIG_HOME-}" ] && printf %s "${HOME}/.nvm" || printf %s "${XDG_CONFIG_H  
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh" # This loads nvm
```

Получим список доступных версий с сервера:

```
v19.0.1
v19.1.0
v19.2.0
v19.3.0
v19.4.0
v19.5.0
v19.6.0
v19.6.1
v19.7.0
v19.8.0
v19.8.1
v19.9.0
v20.0.0
v20.1.0
v20.2.0
v20.3.0
v20.3.1
v20.4.0
v20.5.0
v20.5.1
v20.6.0
v20.6.1
v20.7.0
v20.8.0
v20.8.1
v20.9.0 (LTS: Iron)
v20.10.0 (LTS: Iron)
-> v20.11.0 (LTS: Iron)
v20.11.1 (Latest LTS: Iron)
v21.0.0
v21.1.0
v21.2.0
v21.3.0
v21.4.0
v21.5.0
v21.6.0
v21.6.1
v21.6.2
v21.7.0
```

Установим версию Node 20.11.1:

```
-> ~ nvm install 20.11.1

Downloading and installing node v20.11.1...
Downloading https://nodejs.org/dist/v20.11.1/node-v20.11.1-linux-x64.tar.xz...
##### 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v20.11.1 (npm v10.2.4)
-> ~
-> ~ node -v
v20.11.1
-> ~ npm -v
10.2.4
-> ~
```

Создадим прм-пакет в выбранной директории. Для этого достаточно выполнить команду “npm init” и согласиться на каждый промпт.

```
→ backend-test npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (backend-test)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to /home/jarvis394/Projects/JS/backend-test/package.json:

{
  "name": "backend-test",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}

Is this OK? (yes)
→ backend-test
```

Установим пакет **express** через команду “npm install express”:

```
./backend-test
→ backend-test
→ backend-test npm install express

added 64 packages, and audited 65 packages in 2s

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
→ backend-test l
total 44K
drwxr-xr-x  3 jarvis394 jarvis394 4.0K Mar  7 05:41 .
drwxr-xr-x 31 jarvis394 jarvis394 4.0K Mar  7 05:39 ..
drwxr-xr-x 66 jarvis394 jarvis394 4.0K Mar  7 05:41 node_modules
-rw-r--r--  1 jarvis394 jarvis394 25K Mar  7 05:41 package-lock.json
-rw-r--r--  1 jarvis394 jarvis394 258 Mar  7 05:41 package.json
→ backend-test cat package.json
{
  "name": "backend-test",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.18.3"
  }
}
→ backend-test ls node_modules
accepts          ee-first         has-property-descriptors  mime-types       send
array-flatten    encodeurl        has-proto                ms               serve-static
body-parser      es-define-property  has-symbols              negotiator        set-function-length
bytes            es-errors        hasown                    object-inspect    setprototypeof
call-bind        escape-html      http-errors              on-finished      side-channel
content-disposition  etag             iconv-lite               parseurl          statuses
content-type     express          inherits                 path-to-regexp    toidentifier
cookie           finalhandler     ipaddr.js                proxy-addr        type-is
cookie-signature forwarded         media-typer              qs                unpipe
debug            fresh            merge-descriptors        range-parser      utils-merge
define-data-property  function-bind    methods                  raw-body          vary
depd              get-intrinsic    mime                     safe-buffer
destroy           gopd             mime-db                  safer-buffer
```

Видим, что пакет добавился в package.json, создался файл package-lock.json, а также добавились зависимости пакета в папку node\_modules.

Удаление пакета выполняется через команду “npm uninstall <package>”:

```
→ backend-test npm uninstall express

removed 64 packages, and audited 1 package in 153ms

found 0 vulnerabilities
→ backend-test
```

Обновление пакета выполняется через команду “npm update <package>”. Аргумент “--save” сохраняет в package.json новую версию пакета. Проверить устаревшие пакеты можно командой “npm outdated”.

```

→ backend-test cat package.json | grep express
"express": "^4.11.0"
→ backend-test npm outdated
Package Current Wanted Latest Location Depended_by
express 4.11.0 4.18.3 4.18.3 node_modules/express backend-test
→ backend-test npm update express --save

added 30 packages, removed 4 packages, changed 29 packages, and audited 65 packages in 612ms

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
→ backend-test cat package.json | grep express
"express": "^4.18.3"
→ backend-test █

```

Реализуем команду “rme” для удаления пакета express. Для этого достаточно добавить строчку в поле scripts в package.json:

```

backend-test > package.json > {} scripts > rme
1  {
2    "name": "backend-test",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1",
8      "rme": "npm uninstall express"
9    },
10   "author": "",
11   "license": "ISC",
12   "dependencies": {
13     "express": "^4.18.3"
14   }
15 }
16

```

```

→ backend-test npm run rme

> backend-test@1.0.0 rme
> npm uninstall express

removed 64 packages, and audited 1 package in 164ms

found 0 vulnerabilities
→ backend-test █

```

## Вывод

В ходе работы были изучены основы работы с npm.