

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Практическая работа 1. “Знакомство со средой Node.JS,
пакетным менеджером npm”

Выполнил:

Тимофеев Николай Антонович

Группа
К33402

Проверил:
Добряков Д. И.

Санкт-Петербург

2022 г.

Задача

- Установка npm и node.js
- Инициализация npm-пакета
- Установка express
- Удаление express/попытка его обновить
- Написание кастомной команды для npm, чтобы можно удалить express командой “npm run rme”

Ход работы

1) Установка npm и node.js

NPM и Node (LTS) уже были установлены на мой компьютер с помощью утилиты Brew. Ниже представлен скриншот с версиями и путем до исполняемого файла.

```
timofeev41 on MacBook-Pro-Nikolas.local in ~  
$ which node  
/usr/local/bin/node  
timofeev41 on MacBook-Pro-Nikolas.local in ~  
$ node --version  
v16.14.2  
timofeev41 on MacBook-Pro-Nikolas.local in ~  
$ which npm  
/usr/local/bin/npm  
timofeev41 on MacBook-Pro-Nikolas.local in ~  
$ npm -v  
8.5.0
```

2) Инициализация npm-пакета

```
$ 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: (backend2023)
version: (1.0.0)
description: Test NPM Package
entry point: (index.js)
test command:
git repository:
keywords:
author: Timofeev Nikolas
license: (ISC)
About to write to /Users/timofeev41/Projects/Backend2023/package.json:

{
  "name": "backend2023",
  "version": "1.0.0",
  "description": "Test NPM Package ",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Timofeev Nikolas",
  "license": "ISC"
}

Is this OK? (yes) yes
```

```
timofeev41 on MacBook-Pro-Nikolas.local in ~/Projects/Backend2023
$ ls
package.json
timofeev41 on MacBook-Pro-Nikolas.local in ~/Projects/Backend2023
$ cat package.json
{
  "name": "backend2023",
  "version": "1.0.0",
  "description": "Test NPM Package ",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Timofeev Nikolas",
  "license": "ISC"
}
```

3) Установка express

```
timofeev41 on MacBook-Pro-Nikolas.local in ~/Projects/Backend2023
$ npm i express --save

added 57 packages, and audited 58 packages in 892ms

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

found 0 vulnerabilities
timofeev41 on MacBook-Pro-Nikolas.local in ~/Projects/Backend2023
$ cat package.json | grep express
  "express": "^4.18.2"
```

4) Удаление express/попытка его обновить

```
timofeev41 on MacBook-Pro-Nikolas.local in ~/Projects/Backend2023
$ npm update express

up to date, audited 58 packages in 474ms

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

found 0 vulnerabilities
-----
timofeev41 on MacBook-Pro-Nikolas.local in ~/Projects/Backend2023
$ npm remove express

removed 57 packages, and audited 1 package in 215ms

found 0 vulnerabilities
```

5) Написание кастомной команды для npm, чтобы можно удалить express командой “npm run rme”

```
{
  "name": "backend2023",
  "version": "1.0.0",
  "description": "Test NPM Package ",
  "main": "index.js",
  "scripts": {
    "rme": "npm remove express"
  },
  "author": "Timofeev Nikolas",
  "license": "ISC"
}
```

```
$ npm run rme
```

```
> backend2023@1.0.0 rme
> npm remove express
```

```
up to date, audited 1 package in 136ms
```

Вывод

В ходе данной работы я научился работать с npm (обновлять пакеты, устанавливать их и удалять), писать собственные команды для выполнения необходимых действий (например, удаление express). Оказалось, что scripts в npm это неплохой аналог Makefile для выполнения простых действий (например для запуска автотестов, запуска docker-контейнеров и тд).