# Министерство науки и высшего образования Российской Федерации ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «Национальный исследовательский университет ИТМО» (Университет ИТМО)

Факультет Инфокоммуникационных технологий (ИКТ)

Образовательная программа Мобильные и сетевые технологии

### ОТЧЕТ

по дисциплине: Бек-енд разработка по Домашней работе 1

Тема задания: «Знакомство со средой Node.JS, пакетным менеджером прт»

Обучающийся: Баландин И.О., К33402

Преподаватель: Добряков Д.И.

Оценка: \_\_\_\_\_ 2023 г.

# ЗАДАЧИ

- установка прт и Node.js,
- инициализация прт-пакета,
- ycтaновка express,
- попытка обновить express и удалить,
- написание кастомной команды npm для удаления express командой "npm run rme".

# ХОД РАБОТЫ

1) Процесс установки прт и Node.js (рис. 1).

```
C:\Users\imexi>npm -v
9.5.0
C:\Users\imexi>node -v
v18.15.0
```

Рисунок 1 – Установка прт и node.js

2) Инициализация прт-пакета, для этого я использовала команду *прт init* (рис. 2).

```
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: (hw1)
version: (1.0.0)
description: ""
entry point: (index.js)
test command: "echo \"Error: no test specified\" && exit 1"
git repository: ""
keywords:
author: ""
license: (ISC)
About to write to C:\Users\victo\OneDrive\Pa6очий стол\Backend\hw1\package.json:
  "name": "hw1",
  "version": "1.0.0",
  "description": "\"\"",
  "main": "index.js",
  "scripts": {
    "test": "\"echo \\\"Error: no test specified\\\" && exit 1\""
  },
"repository": {
    "type": "git",
"url": "\"\""
  "author": "\"\"",
"license": "ISC"
Is this OK? (yes)
```

Рисунок 2 – Инициализация прт-пакета

По итогу инициализации npm-пакета у нас появился файл package.json (рис. 3).

```
EXPLORER
                        {} package.json X

✓ OPEN EDITORS

                         {} package.json > ...
  X {} package.json
                                  "name": "hw1",
∨ HW1
                                  "version": "1.0.0",
{} package.json
                                  "description": "",
                                  "main": "index.js",
                                  ▶ Debug
                                  "scripts": {
                                    "test": "echo \"Error: no test specified\" && exit 1"
                                  "author": "\"\"",
                                  "license": "ISC"
                          12
```

## Рисунок 3 – Результат инициализации прт-пакета

3) Установка express, сделаем мы это с помощью команды *npm install express -S* (рис. 4).

```
C:\Users\imexi>npm install express -S

added 58 packages, and audited 59 packages in 2s

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

found 0 vulnerabilities
```

Рисунок 4 — Установка express

После установки express мы можем заметить появление node-modules и package-lock.json (рисунок 5).

```
EXPLORER
                                        {} package-lock.json ×
                      {} package.json
✓ OPEN EDITORS
                       {} package-lock.json > ...
    {} package.json
                              "name": "hw1",
 X {} package-lock.json
                              "version": "1.0.0",
        回の指記
                              "lockfileVersion": 3,
 > node_modules
                               "requires": true,
package-lock.json
                               "packages": {
{} package.json
                                 "name": "hw1",
                                  "version": "1.0.0",
                                 "license": "ISC",
                                 "dependencies": {
                                  "express": "^4.18.2"
                                  "node modules/accepts": {
                                   "version": "1.3.8",
```

Рисунок 5 – Результат установки express

4) Попытка обновить и удалить express командами *npm update express* и *npm uninstall express* (рис. 6).

```
C:\Users\imexi>npm update express
up to date, audited 59 packages in 706ms
8 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
C:\Users\imexi>npm uninstall express
removed 58 packages, and audited 1 package in 488ms
found 0 vulnerabilities
```

# Рисунок 6 – Обновление и удаление express

После удаления папка node-modules и файл package-lock.json очистились (рис. 7).

```
EXPLORER
                                                   {} package-lock.json X
                             {} package.json

∨ OPEN EDITORS

                             {} package-lock.json > ...
      {} package.json
                                        "name": "hw1",
   X {} package-lock.json
                                        "version": "1.0.0",
"lockfileVersion": 3,
           回の指却

✓ node modules

                                        "requires": true,

✓ .bin

                                        "packages": {
  .package-lock.json
                                            "name": "hw1",
 {} package-lock.json
                                            "version": "1.0.0",
"license": "ISC"
 {} package.json
                               14
```

Рисунок 7 – Результат удаления express

5) Написание кастомной команды npm для удаления express. Для этого сначала снова установим express, затем зайдем в файл package.json и пропишем там команду *npm uninstall express* (рис. 8).

```
EXPLORER
                        {} package.json X
                                           {} package-lock.json
✓ OPEN EDITORS
                         {} package.json > ...
  X {} package.json
                                  "name": "hw1",
     {} package-lock.json
                                  "version": "1.0.0",
 HW<sub>1</sub>
                                  "description": "",
 ∨ node_modules
                                  "main": "index.js",

✓ .bin

                                  Debug
 {} .package-lock.json
                                  "scripts": {
                                    "test": "echo \"Error: no test specified\" && exit 1",
{} package-lock.json
                                    "rme": "npm uninstall express"
{} package.json
                                  "author": "\"\"",
                                  "license": "ISC"
                          13
                         > hw1@1.0.0 rme
                         > npm uninstall express
                         removed 58 packages, and audited 1 package in 526ms
                         found 0 vulnerabilities
```

Рисунок 8 – Удаление express кастомной командой прт

Проверим, что node-modules и package-lock.json очистились (рис. 9).

```
EXPLORER
                         {} package.json
                                            {} package-lock.json ×
✓ OPEN EDITORS
                          {} package-lock.json > ...
     {} package.json
                                   "name": "hw1",
  × {} package-lock.json
                                   "version": "1.0.0",
 нw1 [] □ □ О 🗗
                                   "lockfileVersion": 3,

✓ node_modules

                                   "requires": true,

✓ .bin

                                   "packages": {
 {} .package-lock.json
                                       "name": "hw1",
{} package-lock.json
                                       "version": "1.0.0",
{} package.json
                                       "license": "ISC"
```

Рисунок 9 – Результат удаления express кастомной командой прт

# ЗАКЛЮЧЕНИЕ

С помощью данной домашней работы мы научились работать с npm и express, узнали, как создавать кастомные команды.