

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет
Практическая работа №3

Выполнил:

Екушев Владислав

К33402

Проверил:

Добряков Д. И.

Санкт-Петербург

2024 г.

Задача

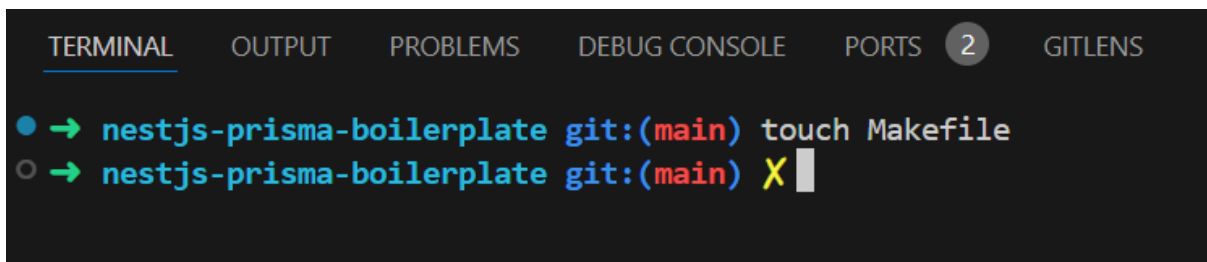
Составьте Makefile, который будет автоматизировать ваши рутинные действия, такие как:

1. запуск приложения;
2. установка зависимостей и сборка приложения

Ход работы

Будем код из лабораторной работы №1.

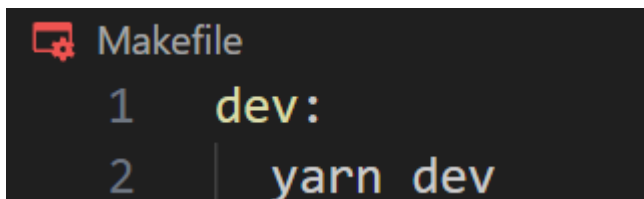
Добавляем Makefile:



The screenshot shows a terminal window with a dark background. At the top, there are tabs: TERMINAL, OUTPUT, PROBLEMS, DEBUG CONSOLE, PORTS, 2, and GITLENS. The 'TERMINAL' tab is active. The terminal shows two commands being executed in a shell named 'nestjs-prisma-boilerplate' on the 'git:(main)' branch. The first command is 'touch Makefile', which is successful. The second command is 'X', which is a placeholder for a failed command.

```
• → nestjs-prisma-boilerplate git:(main) touch Makefile
○ → nestjs-prisma-boilerplate git:(main) X
```

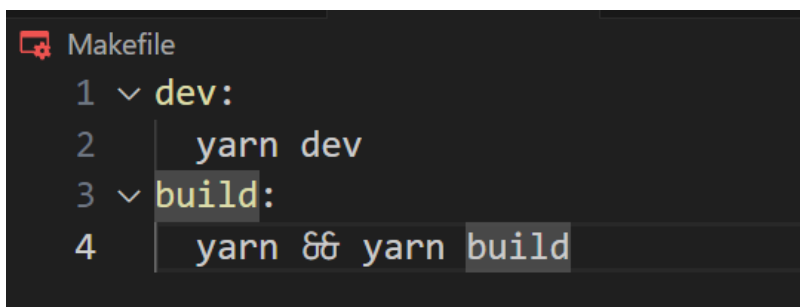
Добавляем команду dev для запуска приложения в dev режиме:



The screenshot shows a Makefile editor with a dark background. The title bar says 'Makefile'. The file content is as follows:

```
1 dev:
2   yarn dev
```

Добавляем команду build для установки зависимостей и сборки приложения:



The screenshot shows a Makefile editor with a dark background. The title bar says 'Makefile'. The file content is as follows:

```
1  ✓ dev:
2    yarn dev
3  ✓ build:
4    yarn && yarn build
```

В начале Makefile добавляем команды phony “dev” и “build”, чтобы система не пробовала найти файлы с данными названиями для сборки.

```
Makefile
1  .PHONY: dev build
2
3  dev:
4      yarn dev
5  build:
6      yarn && yarn build
```

Тестирование работы:

```
➔ nestjs-prisma-boilerplate git:(main) X make dev
yarn dev
yarn run v1.22.22
$ yarn nx run-many -t serve -p frontend backend
$ /home/jarvis394/Projects/JS/nestjs-prisma-boilerplate/node_modules/.bin/nx run-many -t serve -p frontend backend

NX Running target serve for project backend:

- backend

> nx run backend:serve:development

chunk (runtime: main) main.js (main) 56.2 KiB [entry] [rendered]
webpack compiled successfully (f2915718c88f4dff)
Debugger listening on ws://localhost:9229/4fef7a8-e802-4007-aa8d-12f573660237
For help, see: https://nodejs.org/en/docs/inspector

[Nest] 19137 - 06/17/2024, 4:23:47 AM LOG [NestFactory] Starting Nest application...
[Nest] 19137 - 06/17/2024, 4:23:47 AM LOG [InstanceLoader] PassportModule dependencies initialized +9ms
[Nest] 19137 - 06/17/2024, 4:23:47 AM LOG [InstanceLoader] PrismaModule dependencies initialized +0ms
[Nest] 19137 - 06/17/2024, 4:23:47 AM LOG [InstanceLoader] AppModule dependencies initialized +0ms
[Nest] 19137 - 06/17/2024, 4:23:47 AM LOG [InstanceLoader] ConfigHostModule dependencies initialized +0ms
```

```

➤ → nestjs-prisma-boilerplate git:(main) X make build
yarn && yarn build
yarn install v1.22.22
[1/4] Resolving packages...
success Already up-to-date.
$ husky && yarn prisma migrate dev
yarn run v1.22.22
$ /home/jarvis394/Projects/JS/nestjs-prisma-boilerplate/node_modules/.bin/prisma migrate dev
Environment variables loaded from .env
Prisma schema loaded from prisma/schema.prisma
Datasource "db": PostgreSQL database "nestjs-prisma-boilerplate", schema "public" at "127.0.0.1:5432"

Already in sync, no schema change or pending migration was found.

✓ Generated Prisma Client (v5.12.0) to ./node_modules/@prisma/client in 41ms

Done in 1.60s.
Done in 1.91s.
yarn run v1.22.22
$ yarn nx run-many -t build
$ /home/jarvis394/Projects/JS/nestjs-prisma-boilerplate/node_modules/.bin/nx run-many -t build

  ✓ nx run shared:build (1s)
  ✓ nx run backend:build:production (2s)

NX Successfully ran target build for 2 projects (3s)

Done in 3.56s.
➤ → nestjs-prisma-boilerplate git:(main) X

```

Вывод

В ходе работы были получены навыки работы с Makefile.