|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**Отчет по практической работе №3**

по дисциплине

«**Технология разработки программных приложений**»

**Тема: «Системы сборки»**

|  |  |
| --- | --- |
| **Выполнил:**  Студент группыИКБО-36-22 | Утенков Юрий Юрьевич |
| **Проверил:** | ассистент Петрова А.А. |

МОСКВА 2024 г.

**О Г Л А В Л Е Н И Е**

[Практическая работа № 3 3](#_Toc160911945)

[Цель работы 3](#_Toc160911946)

[Выполнение работы 3](#_Toc160911947)

[Контрольные вопросы. 13](#_Toc160911948)

[Выводы 14](#_Toc160911949)

# Практическая работа № 3

Цель работы

Знакомство с системой сборки Gradle. Возможности gradle. Управление зависимостями.

Выполнение работы

**Задание:**

Для выполнения необходимо клонировать (или форкнуть) git-репозиторий

согласно варианту, и выполнить следующие задания:

1. Найти отсутствующую зависимость и указать ее в соответствующем блоке в build.gradle, чтобы проект снова начал собираться
2. В некоторых классах поправить имя пакета
3. Собрать документацию проекта, найти в ней запросы состояния и сущности по идентификатору
4. Собрать jar со всеми зависимостями (так называемый UberJar), после чего запустить приложение. По умолчанию, сервер стартует на порту 8080.
5. Запросить состояние запущенного сервера (GET запрос по адресу http://localhost:8080)
6. Запросить сущность по идентификатору (GET запрос по адресу: <http://localhost:8080/сущность/идентификатор>). Идентификатором будут 3 последних цифры в серийном номере вашего студенческого билета.
7. В задаче shadowJar добавить к jar-файлу вашу фамилию
8. Выполнить задачу checkstyleMain. Посмотреть сгенерированный отчет. Устранить ошибки оформления кода.

**Ход выполнения работы:**

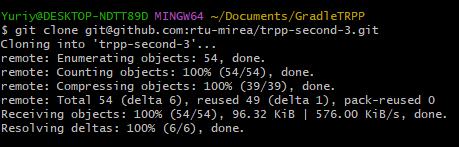


Рисунок 1 ⎯ Клонирование репозитория

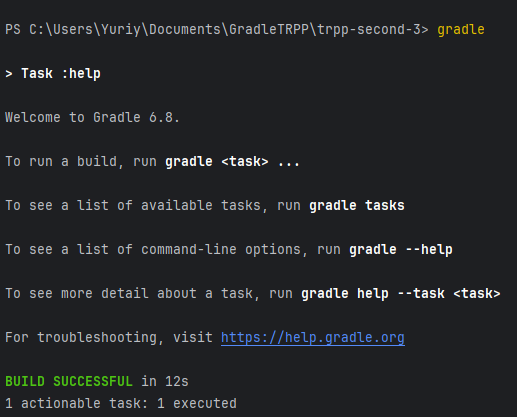


Рисунок 2 ⎯ Подготовка для работы с gradle

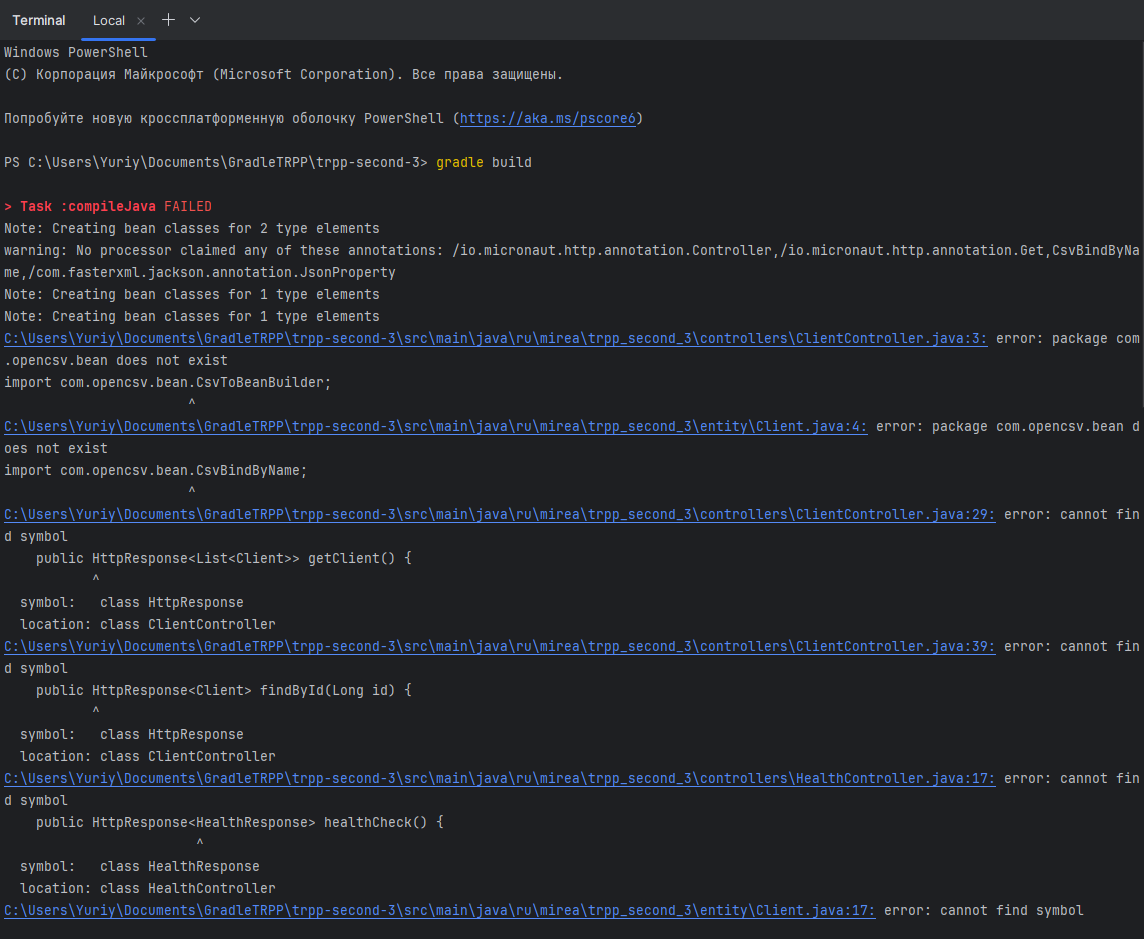


Рисунок 3 ⎯ Попытка сборки данного проекта, часть 1

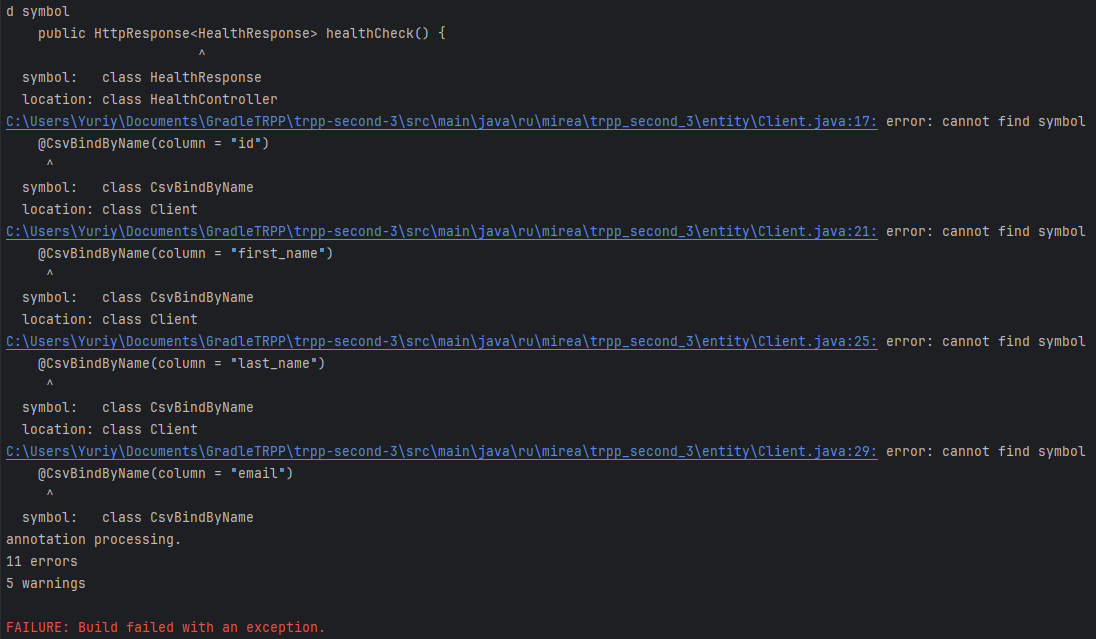


Рисунок 4 ⎯ Попытка сборки данного проекта, часть 2

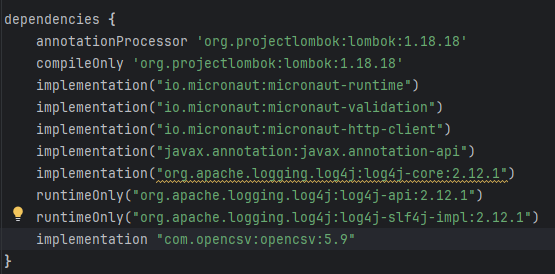


Рисунок 5 ⎯ Добавление зависимости в файл build.gradle

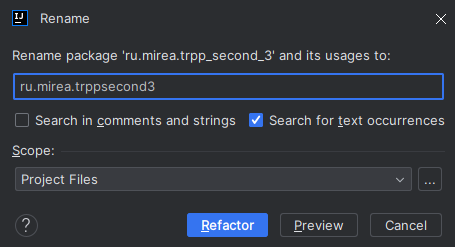


Рисунок 6 ⎯ Переименование пакета ru.mirea.trpp\_second\_3

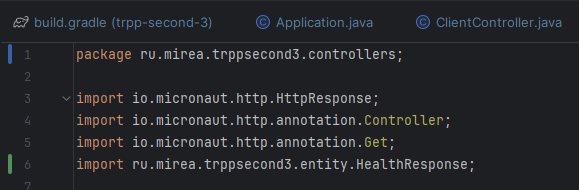


Рисунок 7 ⎯ Ручное добавление импортов в соответствии с правилами оформления кода, часть 1

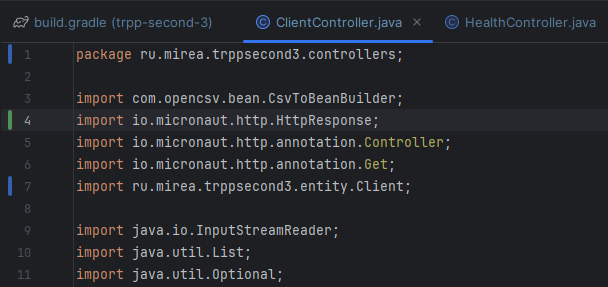


Рисунок 8 ⎯ Ручное добавление импортов в соответствии с правилами оформления кода, часть 2

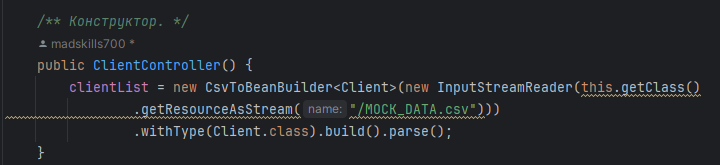


Рисунок 9 ⎯ Разбиение языковой операции на три строки в соответствии с правилами оформления кода (<120 символов на строку)

После исправления данных проблем (добавления импорта, переименования пакета, добавление зависимости в build.gradle и разбиения строки на три части) повторная сборка прошла успешно (Рисунок 10)

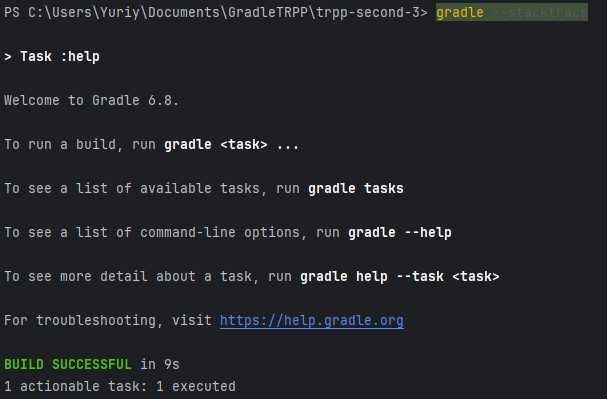


Рисунок 10 ⎯ Успешная сборка проекта

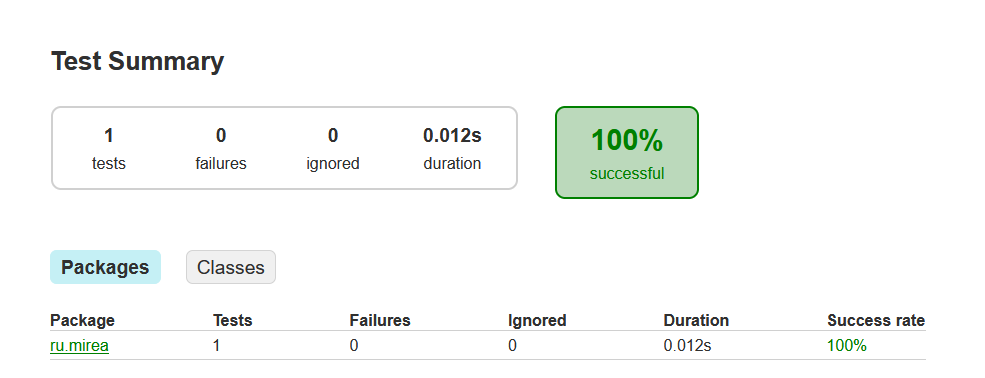


Рисунок 11 ⎯ Успешное прохождение теста

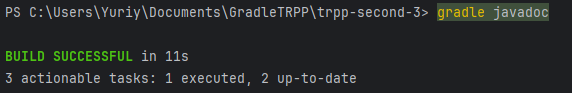


Рисунок 12 ⎯ Генерация JavaDoc документации

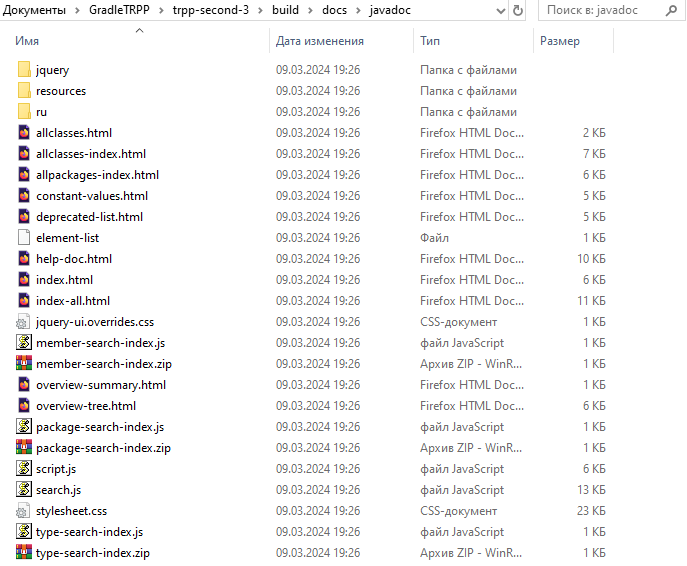


Рисунок 13 ⎯ Сгенерированная документация в папке с проектом

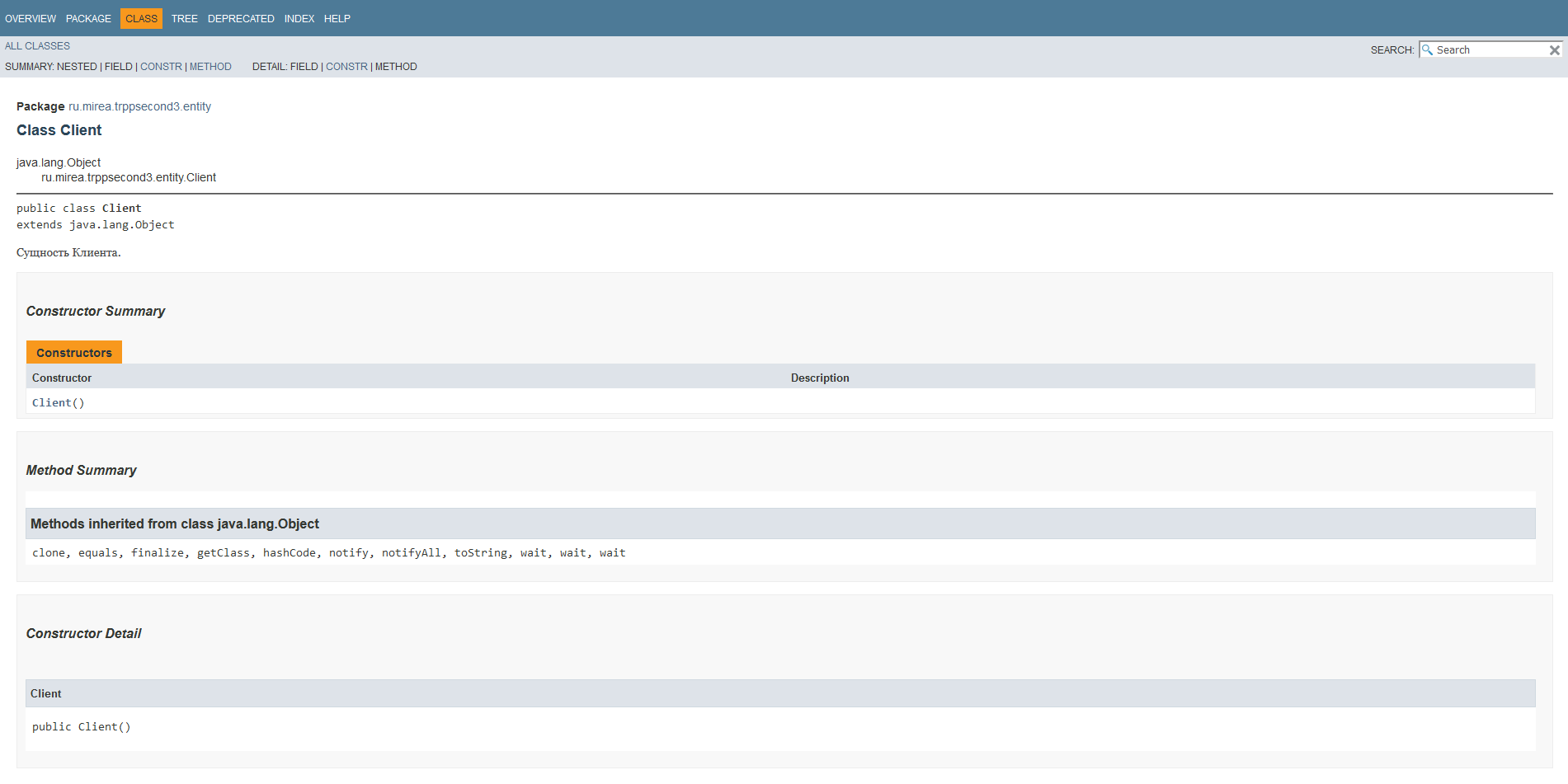


Рисунок 14 ⎯ Документация по сущности (Client)

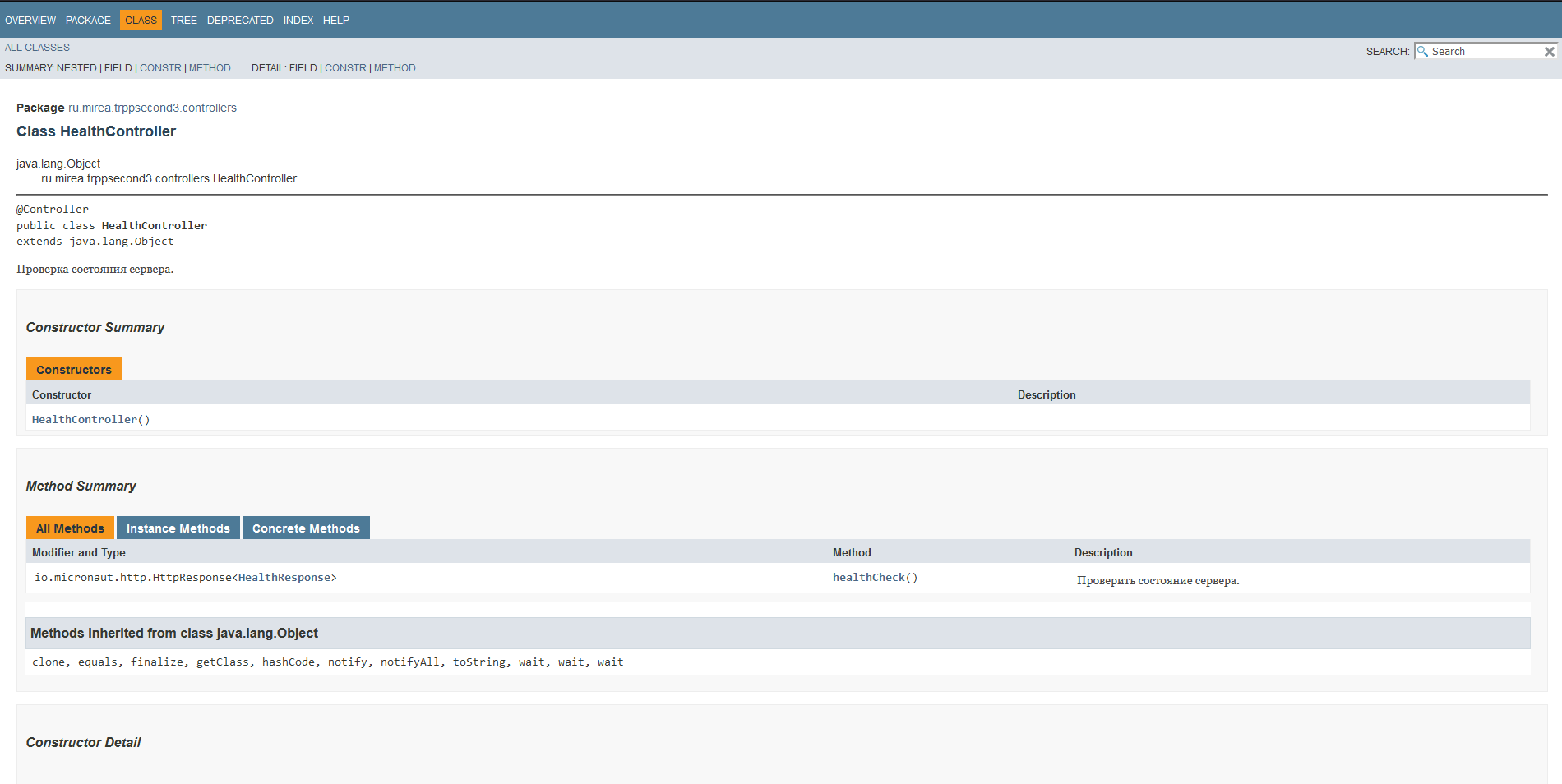


Рисунок 15 ⎯ Документация по запросу состояния, часть 1

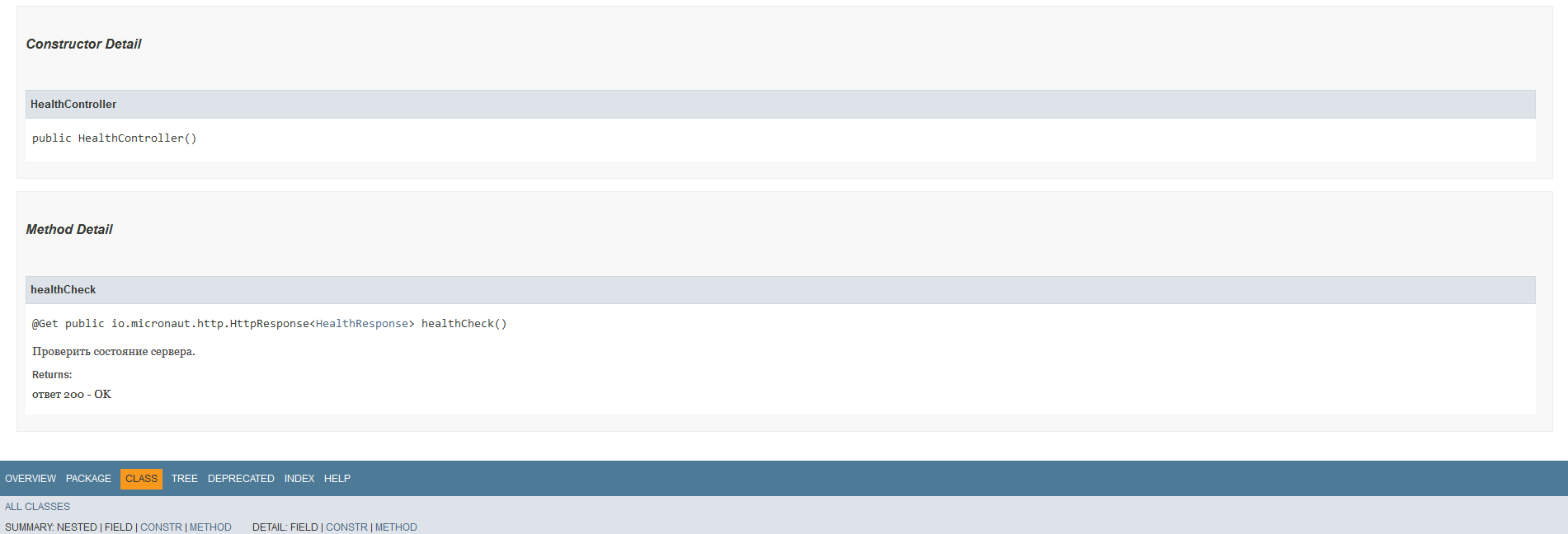


Рисунок 16 ⎯ Документация по запросу состояния, часть 2

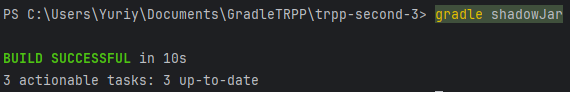


Рисунок 17 ⎯ Сборка Jar со всеми зависимостями

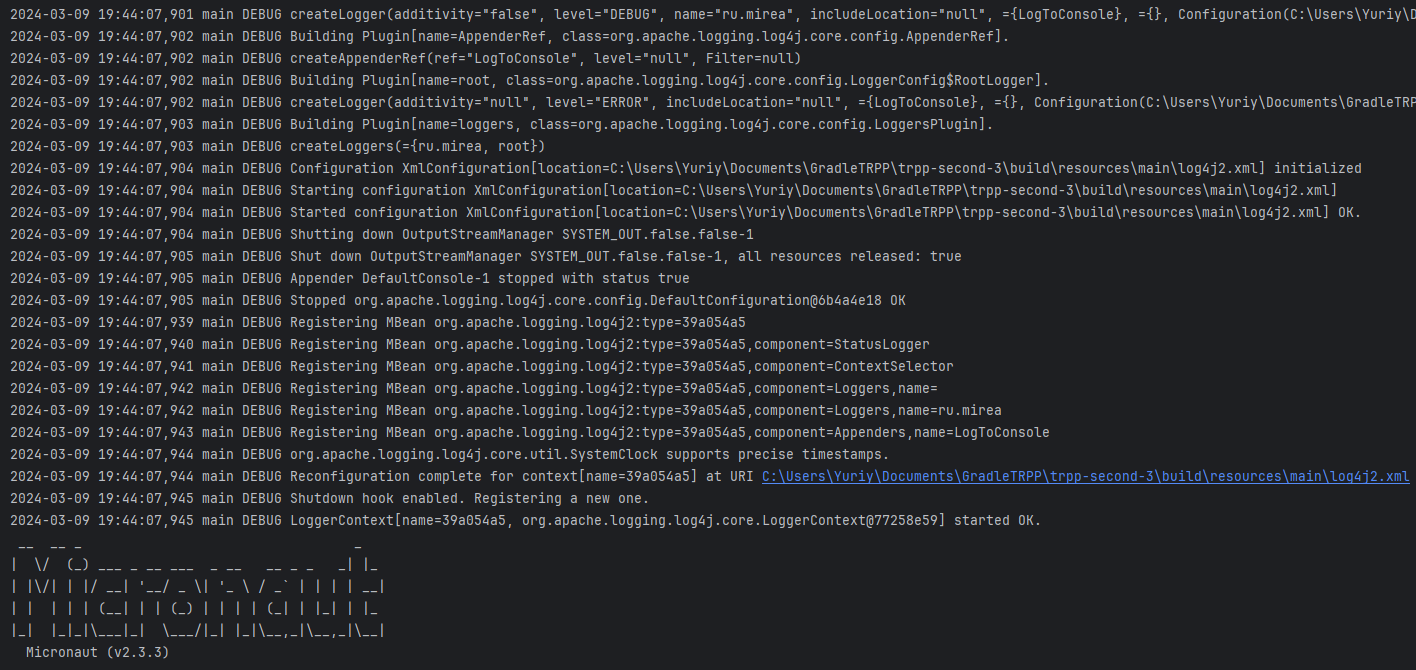


Рисунок 18 ⎯ Запуск данного проекта

На рисунке 19 можно видеть запрос на локальный сервер с помощью команды curl, которая отправляет запрос на сервер, и возвращает ответ сервера на запрос.

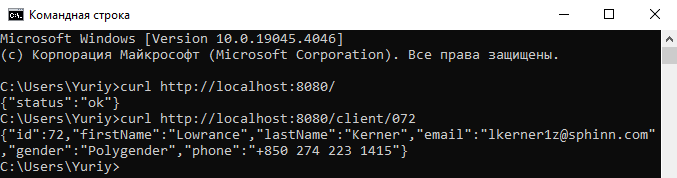


Рисунок 19 ⎯ Статус запущенного сервера (GET запрос по данному URL адресу), а также запрос сущности по идентификатору

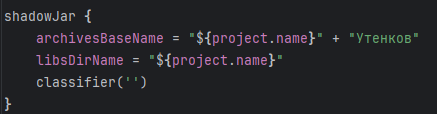


Рисунок 20 ⎯ Изменённая версия jar файла

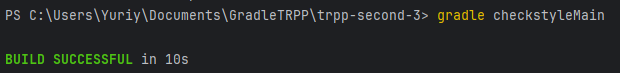


Рисунок 21 ⎯ Запуск задачи checkstyle и генерация отчёта

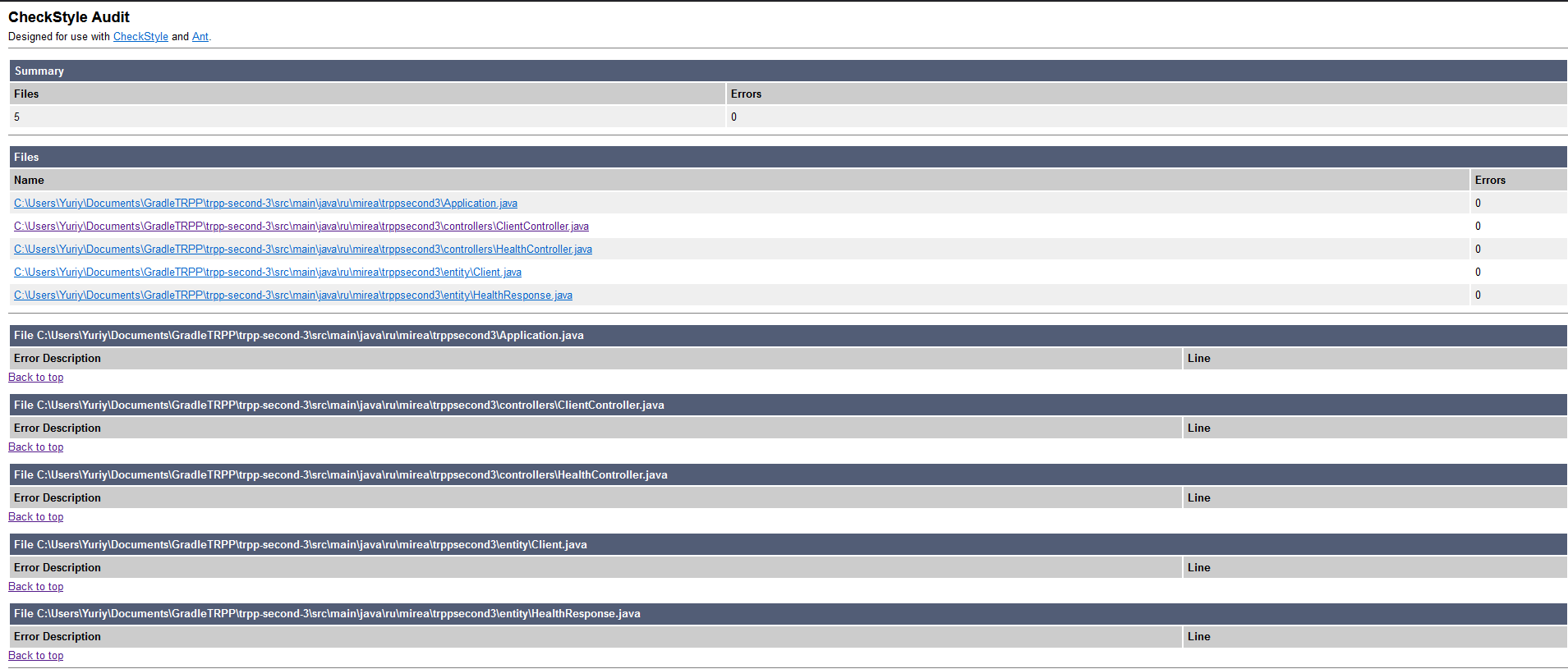


Рисунок 22 ⎯ Отчёт об оформлении кода в main

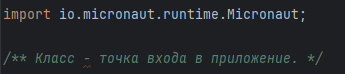


Рисунок 23 ⎯ Ошибка оформления и исправление её

# Контрольные вопросы.

1. Что такое gradle?

Gradle — система сборки проектов с открытым исходным кодом, в которой основной упор идет на гибкость и производительность.

1. Что такое maven?

Apache Maven — фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM, являющемся подмножеством XML.

1. Что такое mavencentral?

Mavencentral – репозиторий для проектов на Maven.

1. Что делает задача clean?

clean — очистка всех сборочных директорий.

1. Что такое javadoc?

Javadoc — генератор документации в HTML-формате из комментариев исходного кода на Java от Sun Microsystems. Javadoc — стандарт для документирования классов Java. Большинство сред разработки программного обеспечения автоматически генерируют HTML-документацию, используя Javadoc.

1. Что такое micronaut?

Micronaut – это основанный на JVM фреймворк с открытым исходным кодом для разработки лёгких, модульных приложений и микросервисов.

# Выводы

В ходе выполнения данной работы я узнал про сборку проектов с помощью gradle. Gradle позволяет удобно собирать проекты, генерировать отчёт про каждый этап сборки, gradle очень популярен для сборки проектов, использующих JVM. Я научился запускать gradle через переменные среды, исправлять зависимости в проекте.