|  |
| --- |
|  |

ОГЛАВЛЕНИЕ

[Выполнение работы 2](#_Toc128936326)

[Ответы на контрольные вопросы 7](#_Toc128936327)

[Вывод 8](#_Toc128936328)

## Выполнение работы

Вариант 10 - репозиторий: https://github.com/rtu-mirea/trpp-second-10, сущность ru.mirea.entity.Game

1. Найти отсутствующую зависимость и указать ее в соответствующем блоке в build.gradle, чтобы проект снова начал собираться.

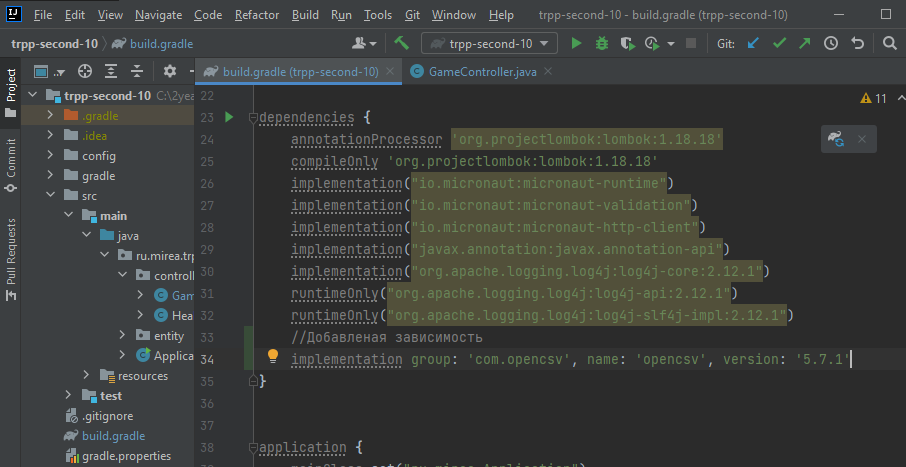


Рисунок 1 - Вставленная зависимость

2. В некоторых классах поправить имя пакета

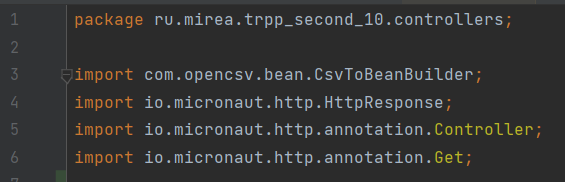


Рисунок 2 - Недостающий import

3. Собрать документацию проекта, найти в ней запросы состояния и сущности по идентификатору.

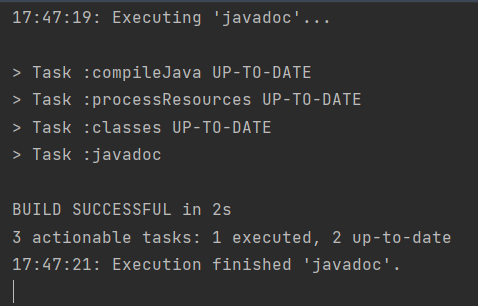


Рисунок 3 - Выполнение команды javadoc



Рисунок 4 - Запрос состояния сервера

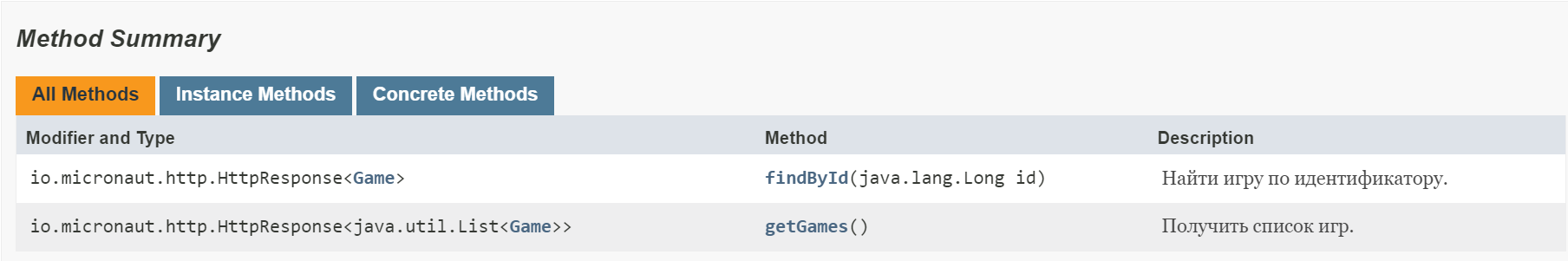


Рисунок 5 - Запрос на получение сущности по идентификатору

4. Собрать jar со всеми зависимостями (так называемый UberJar), после чего запустить приложение. По умолчанию, сервер стартует на порту 8080

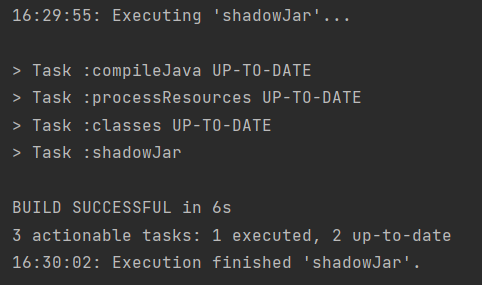


Рисунок 6 - Выполнение команды shadowJar

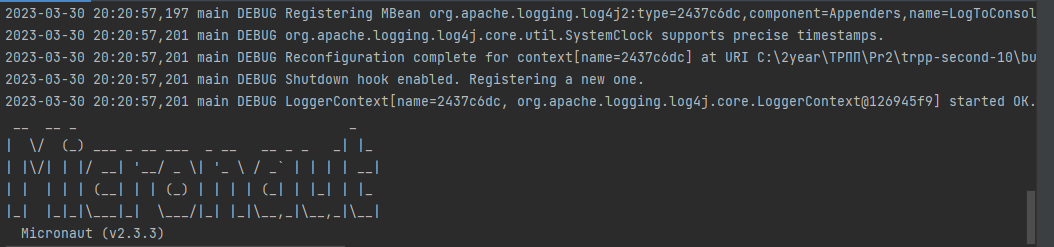


Рисунок 7 - Запуск приложения

5. Запросить состояние запущенного сервера (GET запрос по адресу http://localhost:8080)

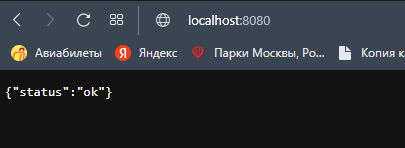


Рисунок 8 - Запрос состояния сервера

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.

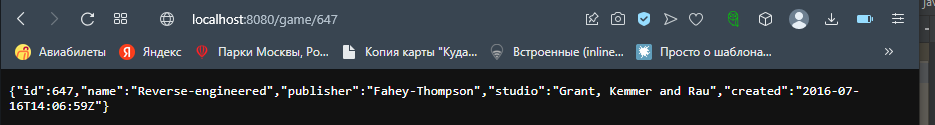
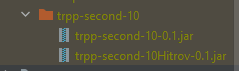


Рисунок 9 - Запрос на получение сущности по идентификатору

7. В задаче shadowJar добавить к jar-файлу вашу фамилию

  
Рисунок 10 - Добавление фамилии к jar-файлу

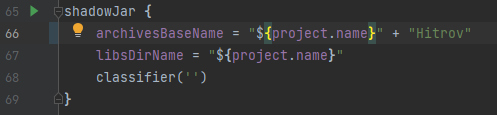


Рисунок 11 - Добавление фамилии к jar-файлу

8. Выполнить задачу checkstyleMain. Посмотреть сгенерированный отчет. Устранить ошибки оформления кода.

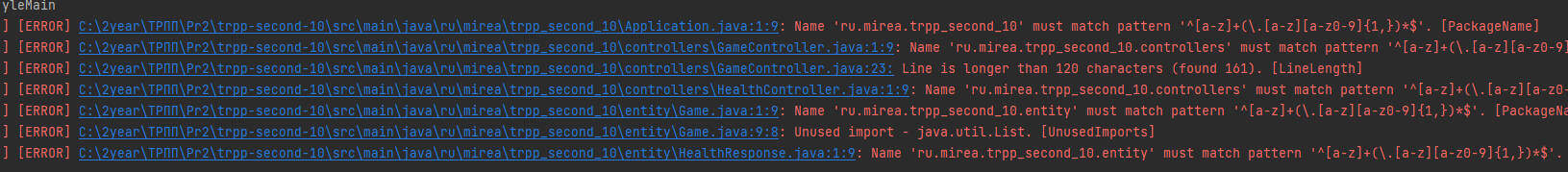


Рисунок 12 - Выполнение команды checkstyleMain

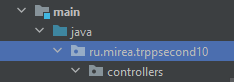


Рисунок 13 - Исправление ошибки в названии пакета

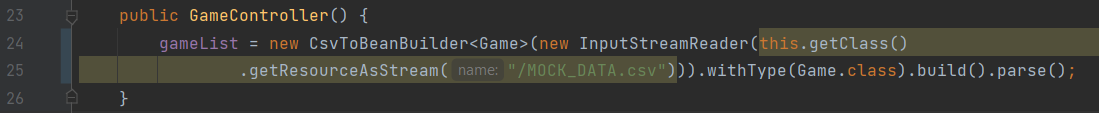


Рисунок 14 - Исправление ошибки в длине строки кода

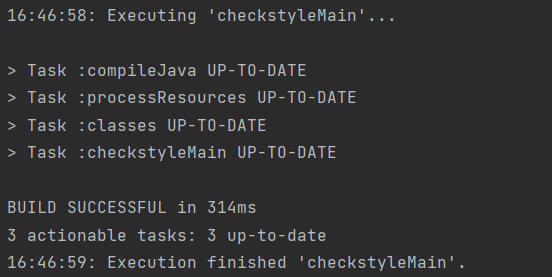


Рисунок 15 - Выполнение команды checkstyleMain, отсутствие ошибок

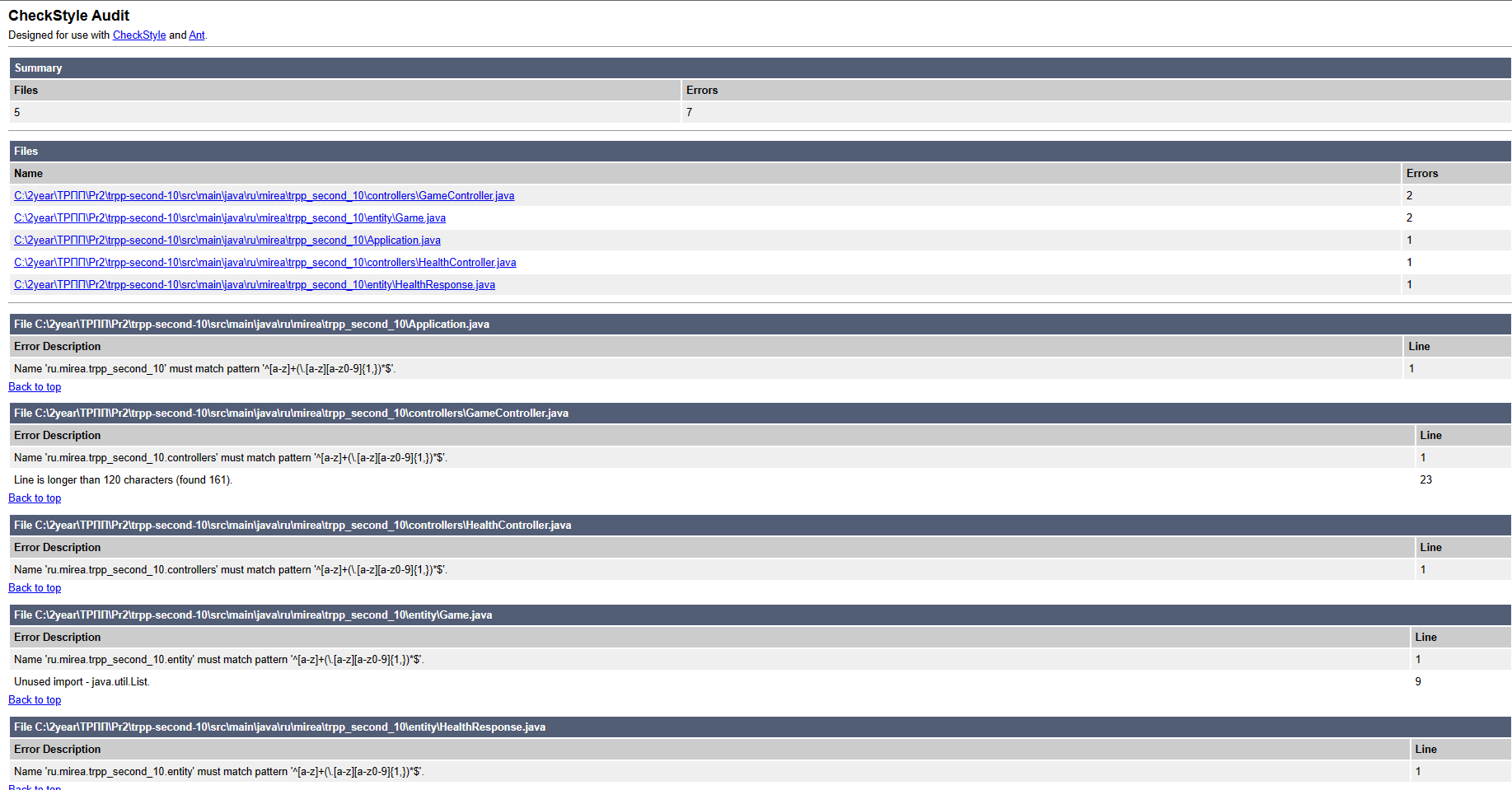


Рисунок 16 - Сгенерированный отчет

## Ответы на контрольные вопросы

**1. Что такое gradle?**

Gradle — система сборки проектов с открытым исходным кодом, в которой основной упор идет на гибкость и производительность. В настоящее время поддерживаются 2 языка программирования для написания скриптов сборки: groovy и kotlin.

**2. Что делает задача clean?**

Очищает все сборочные директории.

**3. Что делает задача build?**

Собирает приложение целиком.

**4. Что делает задача run?**

Запускает приложение.

**5. Что делает задача test?**

Запускает прогон тестов.

**6. Что такое javadoc?**

Генерирует JavaDoc документацию.

## Вывод

В ходе проделанной практической работы были получены навыки работы с системой сборки Gradle. Были изучены возможности и управление зависимостями Gradle.