|  |
| --- |
|  |
| |  |  |  | | --- | --- | --- | |  |  |  | | 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»  по теме «Основы работы с Bash Scriptами»   |  |  | | --- | --- | | **Выполнил:**  Студент группыИКБО-20-22 | Шумахер М. Е. | | **Проверил:** | преподаватель Мельников Д. А. |   МОСКВА 2024 г. |

**СОДЕРЖАНИЕ**

[Задание (Вариант 15) 3](#_Toc160410799)

[Выполнение заданий 4](#_Toc160410800)

[Ответы на вопросы 9](#_Toc160410801)

[Вывод 10](#_Toc160410802)

Задание

**Цель работы**: получить навыки по работе с Bash Script’ами.

**Задание для выполнения**

Часть 1. Базовые Bash скрипты

1. Напишите сценарий, который выводит дату, время, список зарегистрировавшихся пользователей, и uptime системы и сохраняет эту информацию в файл.

2. Напишите сценарий, который выводит содержимое любого каталога или сообщение о том, что его не существует.

3. Напишите сценарий, который с помощью цикла прочитает файл и выведет его содержимое.

4. Напишите сценарий, который с помощью цикла выведет список файлов и директорий из текущего каталога, укажет, что есть файл, а что директория.

5. Напишите сценарий, который подсчитает объем диска, занимаемого директорией. В качестве директории можно выбрать любую директорию в системе.

6. Напишите сценарий, который выведет список всех исполняемых файлов в директории, для которых у текущего пользователя есть права на исполнение.

Часть 2. Развертка и запуск проекта при помощи Bash Script

1. Определение зависимостей проекта

2. Создание виртуального окружения

3. Написание скрипта запуска приложения на новой системе

Выполнение заданий

Найдем отсутствующую зависимость (рис. 1) и укажем ее в соответствующем блоке в build.gradle (рис. 2), чтобы проект снова начал собираться. Проверим соответствующую ошибку (рис. 3).

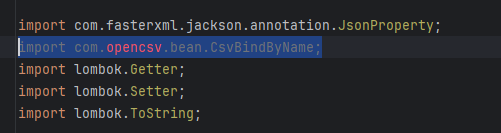


Рисунок 1 – Нахождение отсутствующей зависимости

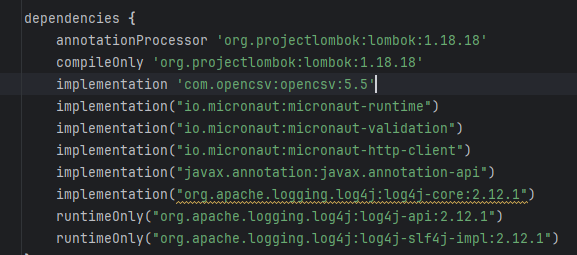


Рисунок 2 – Указание зависимости в build.gradle

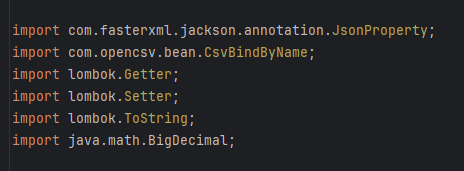


Рисунок 3 – Проверка ошибки в build.gradle

Исправим имя пакета в классе (рис. 4-5). Проверим сборку проекта (рис. 6).

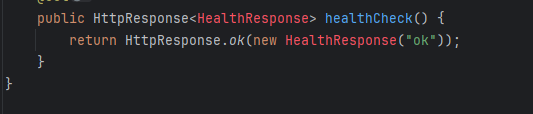


Рисунок 4 – Неправильное имя пакета



Рисунок 5 – Добавление имени пакета

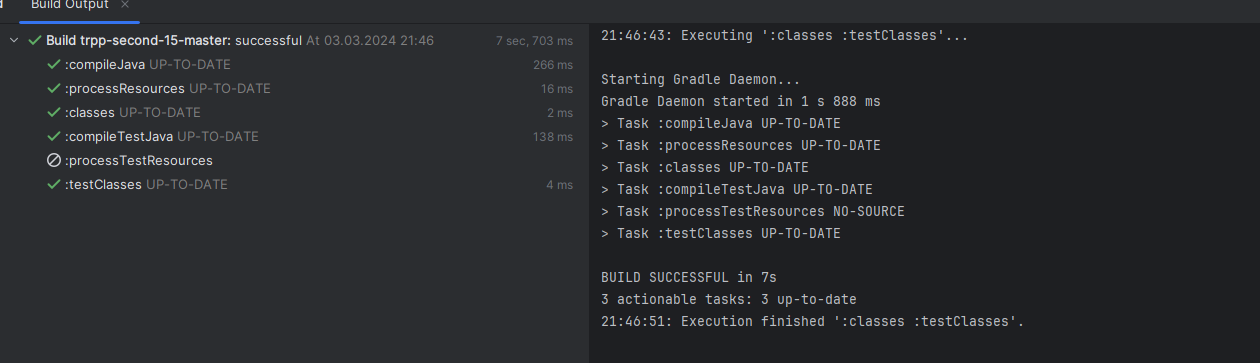


Рисунок 6 – Проверка сборки проекта

Соберем документацию проекта посредством Javadoc, найдем в ней запросы состояния и сущности по идентификатору (рис. 7).

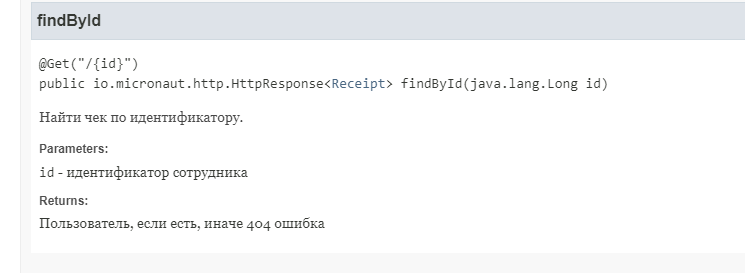


Рисунок 7 – Поиск метода в файле документации

Соберем jar-файл со всеми зависимостями (так называемый UberJar), после чего запустим приложение (рис. 8). По умолчанию, сервер стартует на порту 8080.

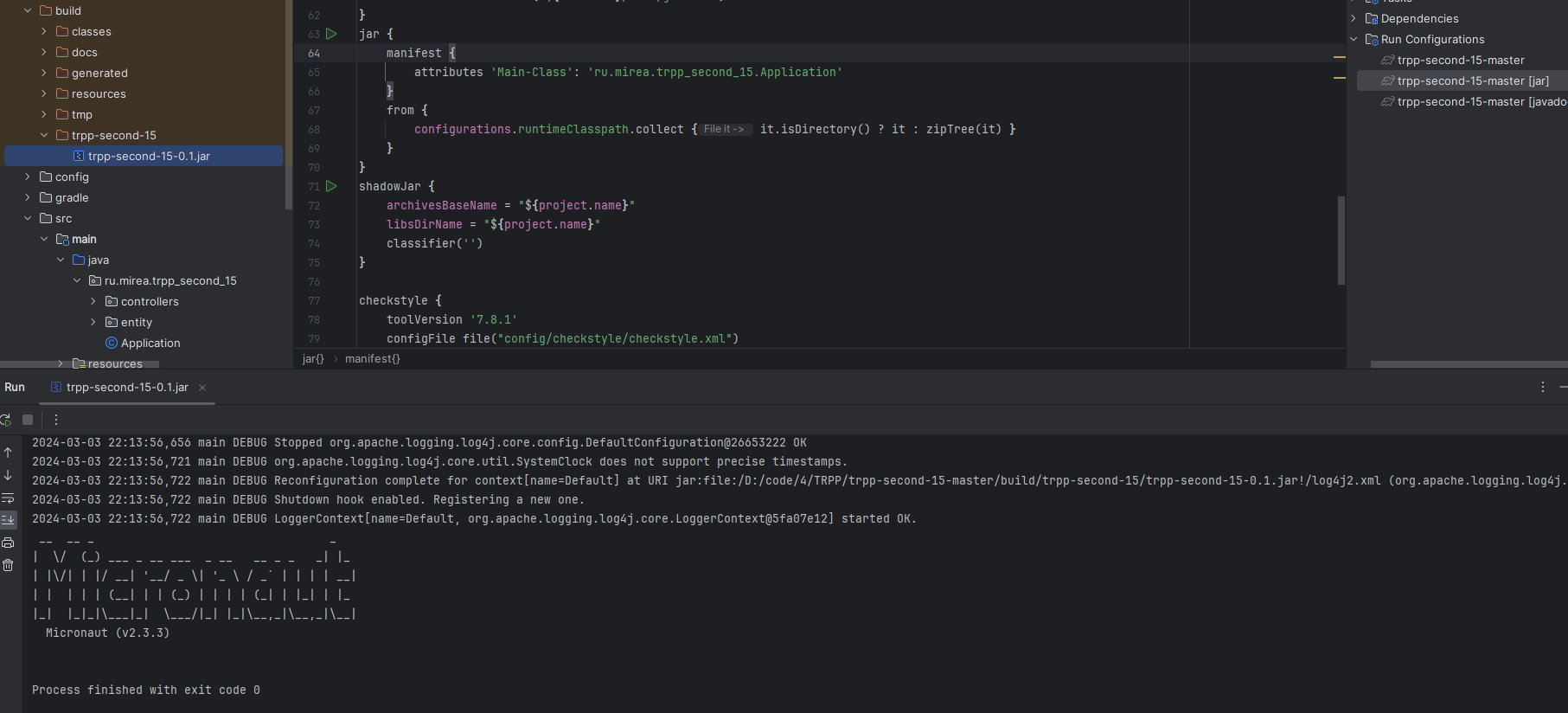


Рисунок 8 – Запуск приложения на 8080 порту

Запросим состояние запущенного сервера (GET запрос по адресу [http://localhost:8080](http://localhost:8080/)) (рис. 9).

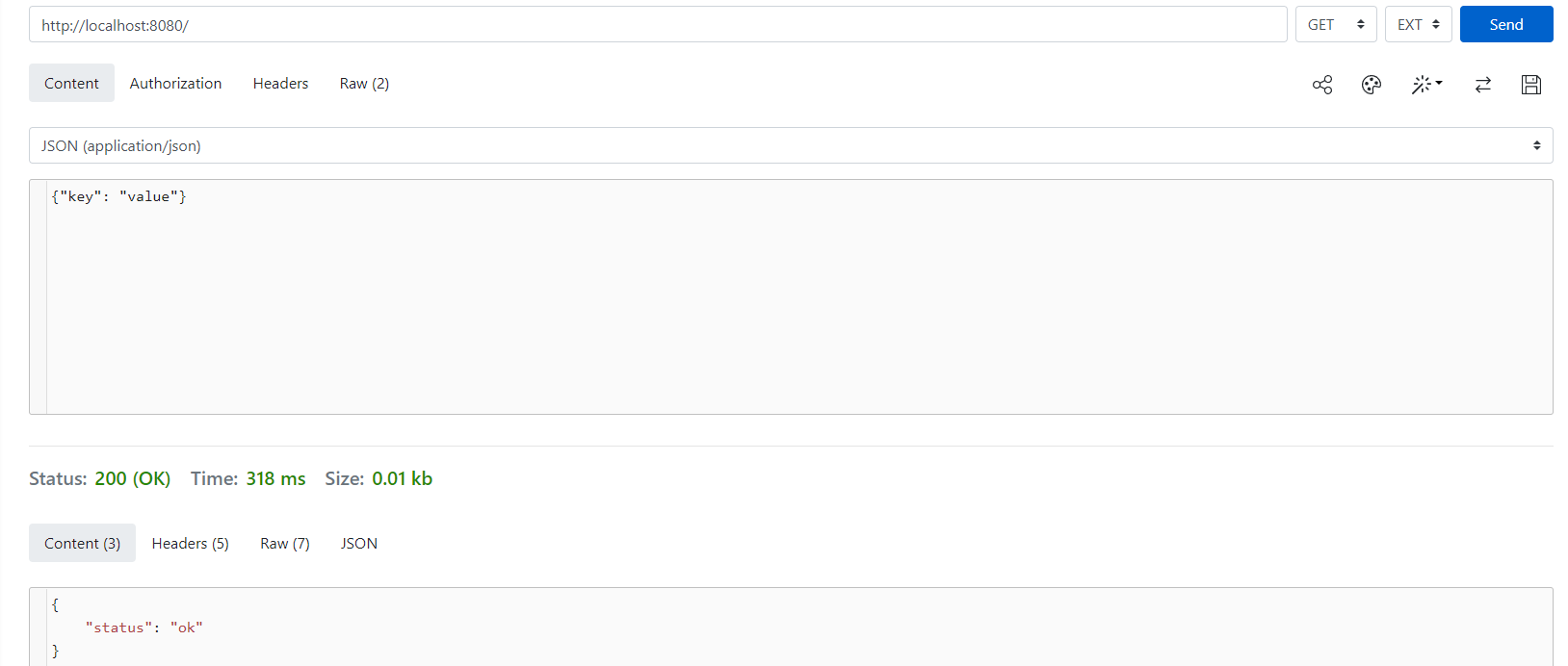


Рисунок 9 – Запрос состояния сервера

Запросим сущность по идентификатору (GET запрос по адресу: [http://localhost:8080/сущность/идентификатор](http://localhost:8080/%D1%81%D1%83%D1%89%D0%BD%D0%BE%D1%81%D1%82%D1%8C/%D0%B8%D0%B4%D0%B5%D0%BD%D1%82%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%82%D0%BE%D1%80)) (рис. 10). Идентификатором будут 3 последних цифры в серийном номере студенческого билета.

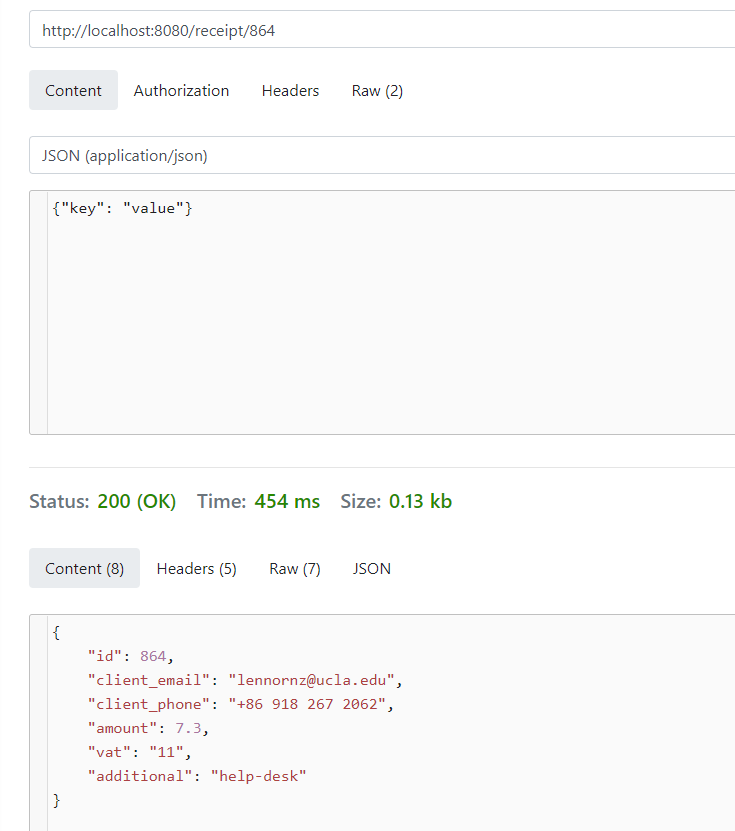


Рисунок 10 – Запрос сущности по идентификатору

В задаче shadowJar добавим к jar-файлу свою фамилию (рис. 11-12).

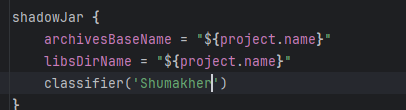


Рисунок 11 – Добавление фамилии к названию jar-файла

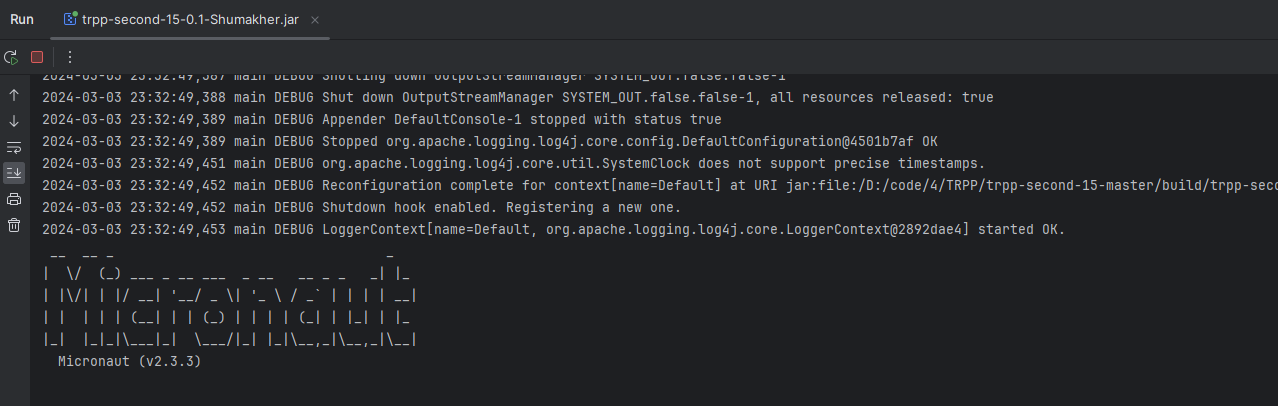


Рисунок 12 – Запуск jar-файла

Выполним задачу checkstyleMain. Посмотрим сгенерированный отчет и устраним ошибки оформления кода (рис. 13-14).



Рисунок 13 – Вывод ошибок по оформлению кода

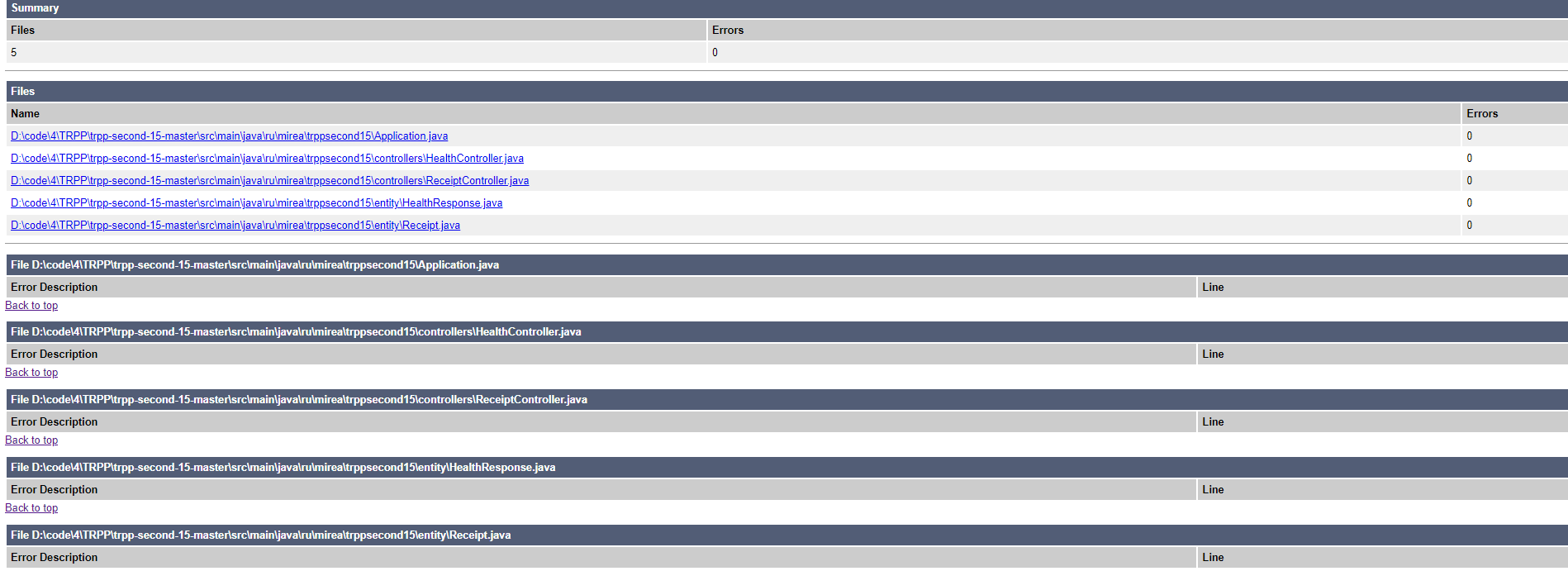


Рисунок 14 – Вывод отсутствия ошибок по оформлению кода

Вывод

В результате выполнения работы были получены навыки по работе с Gradle, Javadoc и Micronaut.