



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт Информационных технологий (ИТ)

Кафедра Математического обеспечения и стандартизации информационных
технологий (МОСИТ)

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 3

по дисциплине

«Технология разработки программных приложений»

Тема: «Системы сборки»

Выполнил студент группы ИКБО-66-23


Тарасова В.А.

Принял

Петренко А.А.

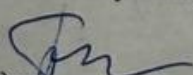
Практическая работа выполнена

«4» 04 2025 г.


(подпись студента)

«Зачтено»

«4» 04 2025 г.


(подпись руководителя)

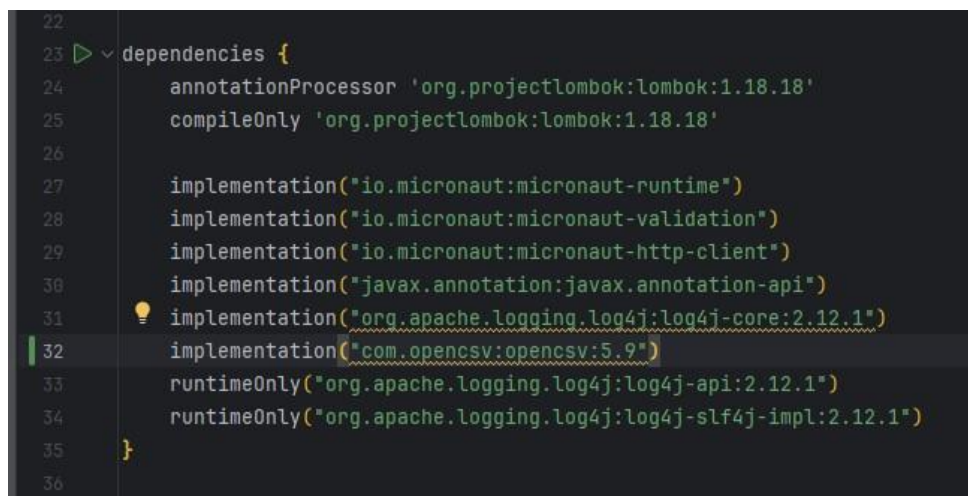
Москва 2025

Оглавление

| | |
|---|----|
| 1. Изменения которые были проведены в исходном коде проекте для всех пунктов задания..... | 3 |
| 2. Скриншоты результатов, полученного в п.5-8 задания | 7 |
| 3. Ответы на вопросы..... | 11 |
| 4. Вывод | 13 |

1. Изменения которые были проведены в исходном коде проекте для всех пунктов задания

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

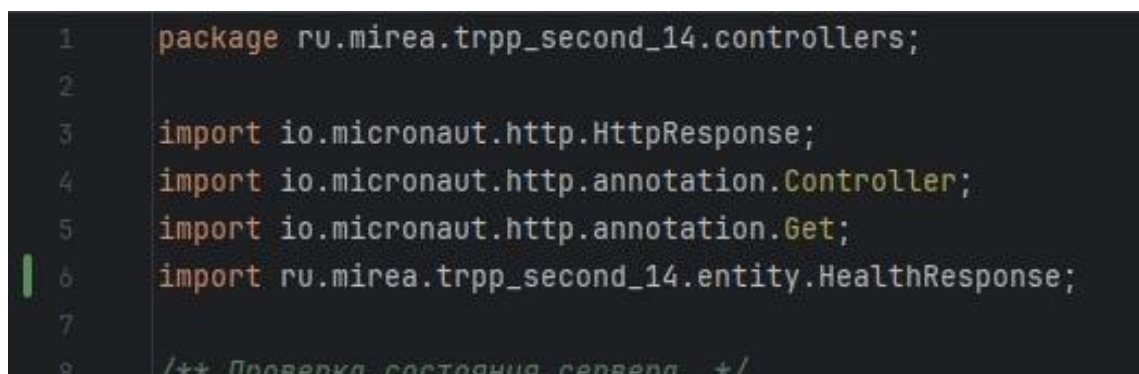
A screenshot of an IDE showing the 'dependencies' block in a build.gradle file. The code is as follows:

```
22
23 dependencies {
24     annotationProcessor 'org.projectlombok:lombok:1.18.18'
25     compileOnly 'org.projectlombok:lombok:1.18.18'
26
27     implementation("io.micronaut:micronaut-runtime")
28     implementation("io.micronaut:micronaut-validation")
29     implementation("io.micronaut:micronaut-http-client")
30     implementation("javax.annotation:javax.annotation-api")
31     implementation("org.apache.logging.log4j:log4j-core:2.12.1")
32     implementation("com.opencsv:opencsv:5.9")
33     runtimeOnly("org.apache.logging.log4j:log4j-api:2.12.1")
34     runtimeOnly("org.apache.logging.log4j:log4j-slf4j-impl:2.12.1")
35 }
36
```

Line 32 is highlighted with a green cursor. A yellow lightbulb icon is visible next to line 31, indicating a suggestion or warning.

Рисунок 1 – Изменения в файле build.gradle

2. В некоторых классах поправить имя пакета

A screenshot of an IDE showing the import statements at the top of a Java file. The code is as follows:

```
1 package ru.mirea.trpp_second_14.controllers;
2
3 import io.micronaut.http.HttpResponse;
4 import io.micronaut.http.annotation.Controller;
5 import io.micronaut.http.annotation.Get;
6 import ru.mirea.trpp_second_14.entity.HealthResponse;
7
8 /** Проверка состояния сервера. */
```

Line 6 is highlighted with a green cursor.

Рисунок 2 – Изменения в файле HealthController.java

Запустим проект с исправленными именами пакетов. Видим, что все собралось без проблем.

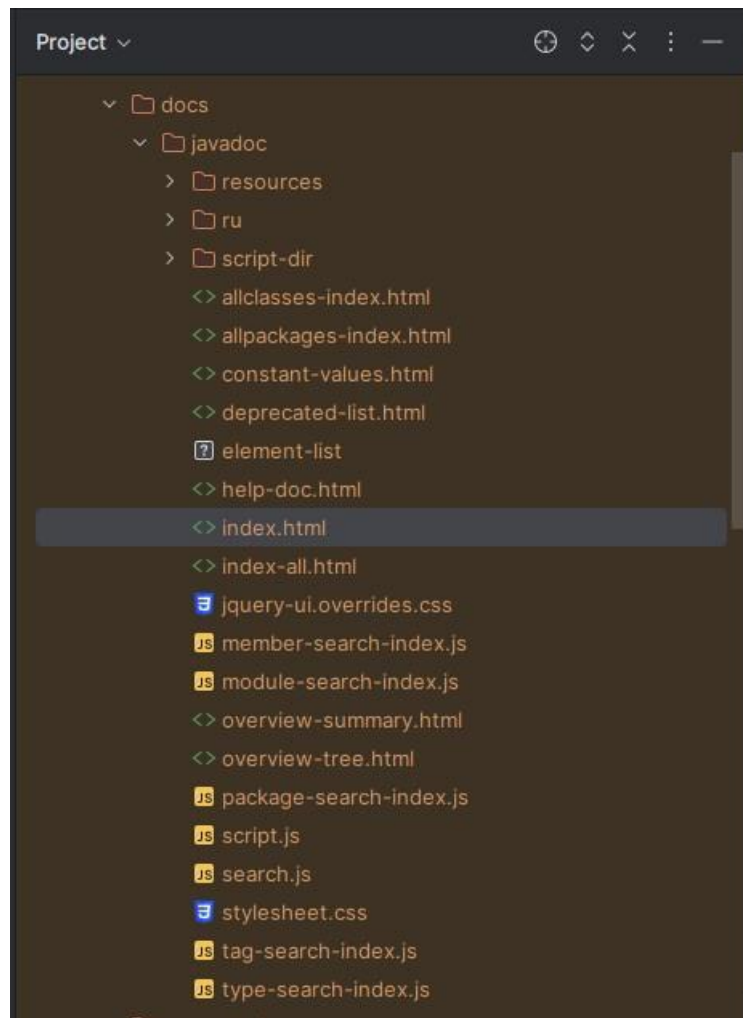


Рисунок 5 – Результат выполнения скрипта

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

```
65 ▶ shadowJar {  
66     archivesBaseName = "${project.name}"  
67     libsDirName = "${project.name}"  
68     classifier('')  
69 }
```

Рисунок 6 – shadowJar

2. Скриншоты результатов, полученного в п.5-8 задания

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

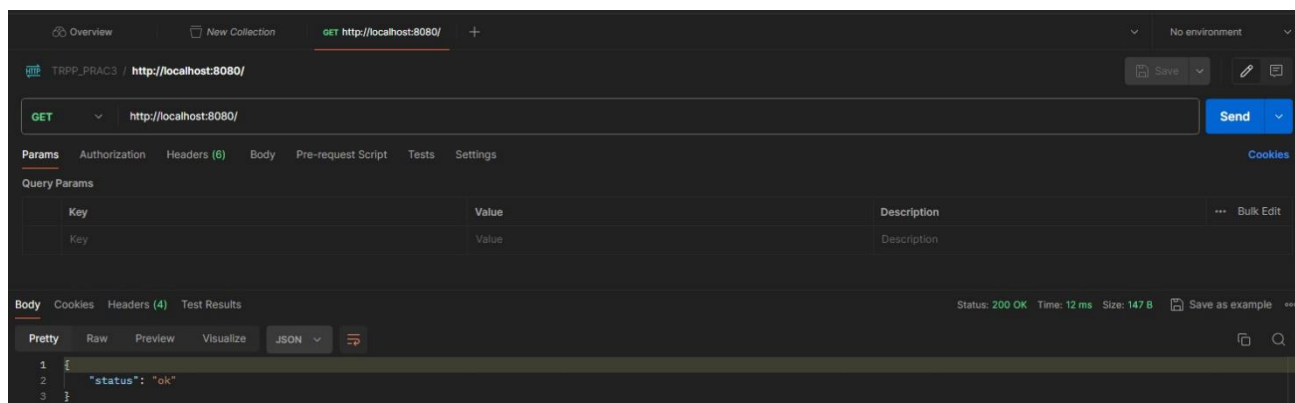


Рисунок 9 - Состояние запущенного сервера

6. Запросить сущность по идентификатору (GET запрос по адресу: `http://localhost:8080/сущность/идентификатор`)

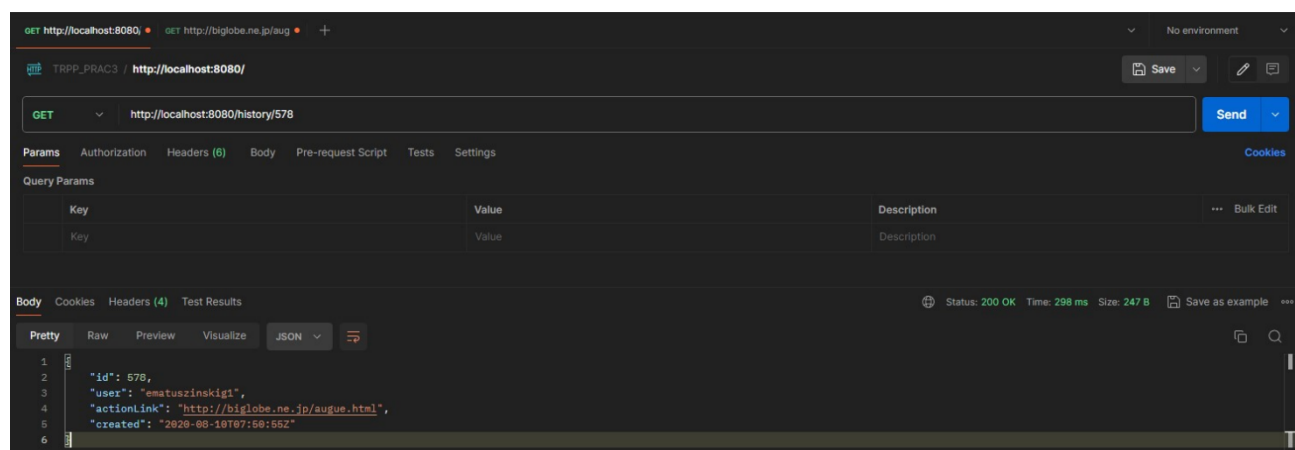


Рисунок 10 – Запрос сущности по идентификатору

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

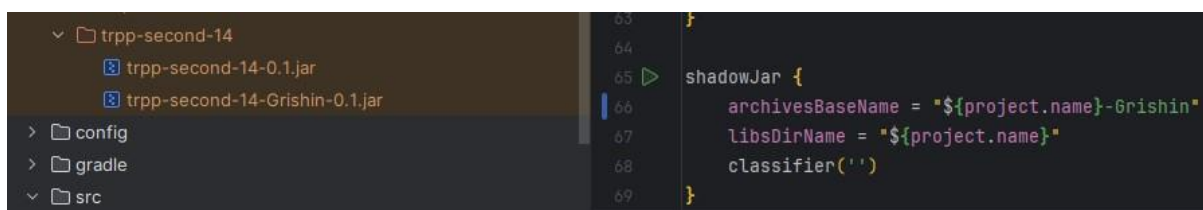


Рисунок 11 – Добавление фамилии к jar-файлу

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

```
Execution failed for task ':checkstyleMain'.
> Checkstyle rule violations were found. See the report at: file:///C:/Users/Rarmash/IdeaProjects/trpp-second-14/build/reports/checkstyle/main.html
Checkstyle files with violations: 5
Checkstyle violations by severity: [error:7]

* Try:
Run with --stacktrace option to get the stack trace. Run with --info or --debug option to get more log output. Run with --scan to get full insights.
```

Рисунок 12 – Запуск checkstyleMain

| Summary | | |
|---------|--------|--|
| Files | Errors | |
| 5 | 7 | |

| Files | | |
|--|--------|--|
| Name | Errors | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\controllers\HistoryController.java | 2 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\entity\History.java | 2 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\Application.java | 1 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\controllers\HealthController.java | 1 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\entity\HealthResponse.java | 1 | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\Application.java | | |
|---|------|--|
| Error Description | Line | |
| Name 'ru.mireael.trpp_second_14' must match pattern '^[a-z]+\.[a-z]{0-9}[1,]*\$'. | 1 | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\controllers\HealthController.java | | |
|--|------|--|
| Error Description | Line | |
| Name 'ru.mireael.trpp_second_14.controllers' must match pattern '^[a-z]+\.[a-z]{0-9}[1,]*\$'. | 1 | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\controllers\HistoryController.java | | |
|---|------|--|
| Error Description | Line | |
| Name 'ru.mireael.trpp_second_14.controllers' must match pattern '^[a-z]+\.[a-z]{0-9}[1,]*\$'. | 1 | |
| Line is longer than 120 characters (found 170). | 22 | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\entity\HealthResponse.java | | |
|---|------|--|
| Error Description | Line | |
| Name 'ru.mireael.trpp_second_14.entity' must match pattern '^[a-z]+\.[a-z]{0-9}[1,]*\$'. | 1 | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trpp_second_14\entity\History.java | | |
|--|------|--|
| Error Description | Line | |
| Name 'ru.mireael.trpp_second_14.entity' must match pattern '^[a-z]+\.[a-z]{0-9}[1,]*\$'. | 1 | |
| Unused import - java.util.List. | 9 | |
| Back to top | | |

Рисунок 13 – Выявленные ошибки

| CheckStyle Audit | | |
|--|--|--|
| Designed for use with CheckStyle and Ant . | | |

| Summary | | |
|---------|--------|--|
| Files | Errors | |
| 5 | 0 | |

| Files | | |
|--|--------|--|
| Name | Errors | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\Application.java | 0 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\controllers\HealthController.java | 0 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\controllers\HistoryController.java | 0 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\entity\HealthResponse.java | 0 | |
| C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\entity\History.java | 0 | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\Application.java | | |
|---|------|--|
| Error Description | Line | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\controllers\HealthController.java | | |
|--|------|--|
| Error Description | Line | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\controllers\HistoryController.java | | |
|---|------|--|
| Error Description | Line | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\entity\HealthResponse.java | | |
|---|------|--|
| Error Description | Line | |
| Back to top | | |

| File C:\Users\Rarmash\IdeaProjects\trpp-second-14\src\main\java\ru\mirael\trppsecond14\entity\History.java | | |
|--|------|--|
| Error Description | Line | |
| Back to top | | |

Рисунок 14 – Теперь нет ошибок

3. Ответы на вопросы

5) Что такое Gradle?

- **Gradle** - это инструмент сборки проектов, который используется в основном для разработки программного обеспечения на языке Java, хотя его также можно использовать и для других языков программирования. Он предоставляет гибкий способ управления проектами и их зависимостями, а также автоматизации сборки, тестирования и развертывания приложений. Gradle использует DSL (Domain Specific Language) на основе Groovy или Kotlin для описания сборочных скриптов, что делает его более удобным и гибким по сравнению с другими инструментами сборки.

6) Что такое Maven?

- **Maven** - это инструмент управления проектами, который обеспечивает сборку, отчетность и управление зависимостями в проектах на Java. Он использует XML-файлы для описания структуры проекта, его зависимостей и сборочных целей. Maven автоматически загружает зависимости из центрального репозитория и обеспечивает стандартизированные структуры проектов, что делает его широко используемым инструментом в сообществе Java.

9) Что делает задача build?

- Задача **build** (сборка) в инструментах сборки, таких как Gradle или Maven, отвечает за компиляцию и сборку исходного кода проекта в исполняемый или дистрибутивный файл. Это включает в себя выполнение различных этапов, таких как компиляция исходного кода, копирование ресурсов, создание исполняемых файлов, упаковку и т. д.

10) Что делает задача compileJava?

- Задача **compileJava** отвечает за компиляцию исходного кода Java в байт-код Java (файлы .class). Она часто является частью процесса сборки проекта и выполняется перед другими этапами сборки, такими как создание исполняемых файлов или дистрибутивов.

13) Что такое javadoc?

- Javadoc - это инструмент, входящий в состав JDK (Java Development Kit), который автоматически генерирует документацию API на основе комментариев в исходном коде Java. Эти комментарии должны следовать определенным соглашениям о стиле и содержать информацию о классах, методах, полях и других элементах API. Сгенерированная документация предоставляет полезную справочную информацию для разработчиков, использующих библиотеку или API.

14) Что такое checkstyle?

- Checkstyle - это инструмент статического анализа кода для языка Java, который используется для обеспечения согласованности и соответствия кода определенным стандартам оформления. Он проверяет исходный код на соответствие правилам, определенным в конфигурационных файлах, и предоставляет отчет о найденных стилистических или структурных нарушениях. Checkstyle помогает обеспечить высокое качество кода, улучшить его читаемость и уменьшить количество потенциальных ошибок.

4. Вывод

В ходе выполнения данной работы были приобретены и освоены навыки работы с gradle, сборкой проекта, работы с сервером.