



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»

Выполнил:

Студент группы ИКБО-10-23

Лазаренко С. А.

Проверил:

Преподаватель к.э.н., доцент

Петросян Л. Э.

Москва 2025 г.

СОДЕРЖАНИЕ

МОДИФИКАЦИЯ ИСХОДНОГО КОДА ПРОЕКТА	3
СКРИНШОТЫ РЕЗУЛЬТАТОВ К ЗАДАНИЯМ 5-8.....	6
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	8
ВЫВОД.....	10

МОДИФИКАЦИЯ ИСХОДНОГО КОДА ПРОЕКТА

1. Нахождение отсутствующей зависимости и указание ее в соответствующем блоке в `build.gradle`, чтобы проект снова начал собираться

Чтобы конфигурационный файл `gradle` подавал хотя бы какие-то признаки жизни, пришлось откатить JDK, потому что новые версии не поддерживались Micronaut и Gradle. Далее добавил недостающую зависимость `implementation("io.micronaut:micronaut-inject")`, который позволял Micronaut найти модуль для инъекции зависимостей. После внесенных изменений я запустил команду `./gradlew build` для сборки:

```
dependencies {
    annotationProcessor 'org.projectlombok:lombok:1.18.18'
    compileOnly 'org.projectlombok:lombok:1.18.18'

    implementation("io.micronaut:micronaut-runtime")
    implementation("io.micronaut:micronaut-inject")
    implementation("com.opencsv:opencsv:5.9")
    implementation("io.micronaut:micronaut-validation")
    implementation("io.micronaut:micronaut-http-client")
    implementation("javax.annotation:javax.annotation-api")
    implementation("org.apache.logging.log4j:log4j-core:2.12.1")
    runtimeOnly("org.apache.logging.log4j:log4j-api:2.12.1")
    runtimeOnly("org.apache.logging.log4j:log4j-slf4j-impl:2.12.1")
}
```

Рисунок 1 – Изменения в файле `build.gradle`

2. В некоторых классах поправить имя пакета

До изменений:

```
package same_text;

import lombok.AllArgsConstructor;
import lombok.Getter;
import lombok.Setter;
```

Рисунок 2 – Файл `HealthResponse` до изменений

После изменений:

```
package ru.mirea.trpp_second_2.entity;

import lombok.AllArgsConstructor;
import lombok.Getter;
import lombok.Setter;
```

Рисунок 3 – Файл HealthResponse после изменений

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

Выполним команду `./gradlew javadoc`:

```
gwynbleidd@MacBook-Air-Sergej-3 trpp-second-2 % ./gradlew javadoc

> Task :compileJava
Note: Creating bean classes for 2 type elements
warning: No processor claimed any of these annotations: /io.micronaut.http.annotation.Controller,/io.micronaut.http.annotation.Set,/com.fasterxml.jackson.annotation.JsonProperty,/com.opencsv.bean.CsvBindByName
1 warning

BUILD SUCCESSFUL in 5s
3 actionable tasks: 2 executed, 1 up-to-date
gwynbleidd@MacBook-Air-Sergej-3 trpp-second-2 % ls build/docs/javadoc

allclasses-index.html      index-all.html           overview-summary.html     search.js
allclasses.html            index.html                overview-tree.html        stylesheet.css
allpackages-index.html     jquery                   package-search-index.js   type-search-index.js
constant-values.html       jquery-ui.overrides.css   package-search-index.zip  type-search-index.zip
deprecated-list.html       legal                     resources
element-list               member-search-index.js    ru
help-doc.html              member-search-index.zip   script.js
```

Рисунок 4 – Результат выполнения скрипта

4. Собрать jar со всеми зависимостями(так называемый UberJar), после чего запустить приложение

Выполним команду `./gradlew shadowJar`:

```
shadowJar {
    archiveBaseName = "${project.name}"
    libsDirName = "${project.name}"
    archiveClassifier.set("") // Этот метод заменяет classifier('')
}
```

Рисунок 5 – shadowJar

```
> tmp
  > trpp-second-2
    > trpp-second-2-0.1.jar
```

Рисунок 6 – Результат выполнения скрипта

```

null null Initializing configuration \n[Configuration[location=jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-14/build/trpp-second-14-0.1.jar!/log4j2.xml]
Installed 1 script engine
to be removed from a future JRE release
Oracle Nashorn version: 11.9.21, language: ECMAScript, threading: Not thread safe, compile: true, names: [nashorn, Nashorn, js, JS, JavaScript, javascript, ECMAScript, ecmasc
JS scripting NashornScriptEngineFactory
PluginManager 'Core' found 118 plugins
PluginManager 'Level' found 6 plugins
PluginManager 'Layout' found 13 plugins
Building Plugin[name=layout, class=org.apache.logging.log4j.core.layout.PatternLayout].
PluginManager 'TypeConverter' found 26 plugins
PatternLayoutBuilder(pattern="%d{HH:mm:ss} [%t] %-5level %logger{36} - %msg%n", PatternSelector=null, Configuration[jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-1
14-0.1.jar!/log4j2.xml], Replace=null, charset="null", alwaysWriteExceptions="null", disableAnsi="null", noConsoleNoAnsi="null", header="null", footer="null")
PluginManager 'Converter' found 66 plugins
Building Plugin[name=appender, class=org.apache.logging.log4j.core.appender.ConsoleAppender].
ConsoleAppenderBuilder(target="SYSTEM_OUT", #follow="null", direct="null", bufferSize="null", bufferSize="null", immediateFlush="null", ignoreExceptions="null", PatternLayout
Appender[36] - %msg%n), name="LogToConsole", Configuration[jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-14/build/trpp-second-14-0.1.jar!/log4j2.xml], Filter

Starting OutputStreamManager SYSTEM_OUT false false
Building Plugin[name=appender, class=org.apache.logging.log4j.core.config.AppendersPlugin].
createAppenders([LogToConsole])
Building Plugin[name=appenderRef, class=org.apache.logging.log4j.core.config.AppenderRef].
createAppenderRef(ref="LogToConsole", level="null", filter=null)
Building Plugin[name=logger, class=org.apache.logging.log4j.core.config.LoggerConfig].
createLogger(additivity=false, level="DEBUG", name="ru.mirea", includeLocation="null", [LogToConsole], []). Configuration[jar:file:/C:/Users/Ramash/IdeaProjects/trpp-seco
nd-14-0.1.jar!/log4j2.xml], filter=null)
Building Plugin[name=appenderRef, class=org.apache.logging.log4j.core.config.AppenderRef].
createAppenderRef(ref="LogToConsole", level="null", filter=null)
Building Plugin[name=logger, class=org.apache.logging.log4j.core.config.LoggerConfig].
createLogger(additivity="null", level="TRACE", includeLocation="null", [LogToConsole], []). Configuration[jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-14/build/trpp-s
econd-14-0.1.jar!/log4j2.xml], filter=null)
Building Plugin[name=loggers, class=org.apache.logging.log4j.core.config.LoggerPlugin].
createLoggers([ru.mirea, root])
Configuration \n[Configuration[location=jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-14/build/trpp-second-14-0.1.jar!/log4j2.xml] Initialized
Starting configuration \n[Configuration[location=jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-14/build/trpp-second-14-0.1.jar!/log4j2.xml]
Started configuration \n[Configuration[location=jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-14/build/trpp-second-14-0.1.jar!/log4j2.xml] OK
Shutting down OutputStreamManager SYSTEM_OUT false false 1
Shut down OutputStreamManager SYSTEM_OUT false false 1, all resources released: true
Appender DefaultConsole-1 stopped with status true
Stopped org.apache.logging.log4j.core.config.DefaultConfiguration[2f280884] OK
org.apache.logging.log4j.core.util.SystemClock does not support precise timestamps
Reconfiguration complete due context[name=Default] at URI jar:file:/C:/Users/Ramash/IdeaProjects/trpp-second-14/build/trpp-second-14-0.1.jar!/log4j2.xml (arg.
ment$504fb6d) with optional classloader: null
Shutdown mode enabled. Registering a new one.
LoggerContext[name=Default, org.apache.logging.log4j.core.LoggerContext$504fb6d] started OK.

```

Рисунок 7 – Результат выполнения сборки

СКРИНШОТЫ РЕЗУЛЬТАТОВ К ЗАДАНИЯМ 5-8

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

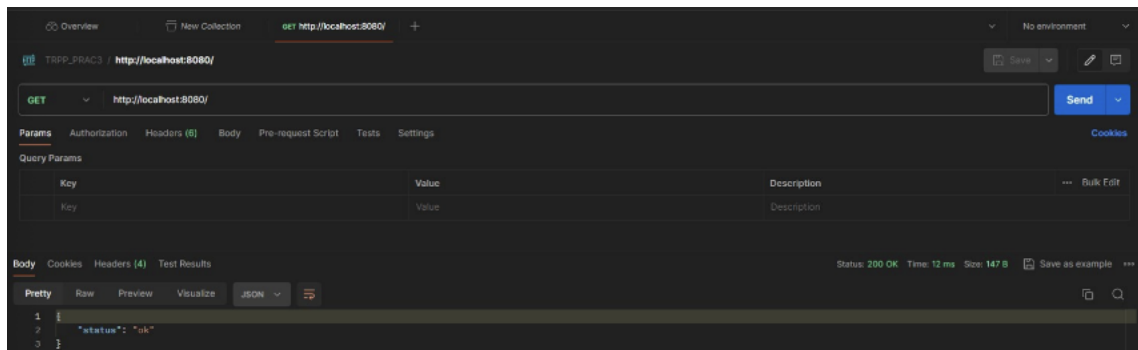


Рисунок 8 - Состояние запущенного сервера

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

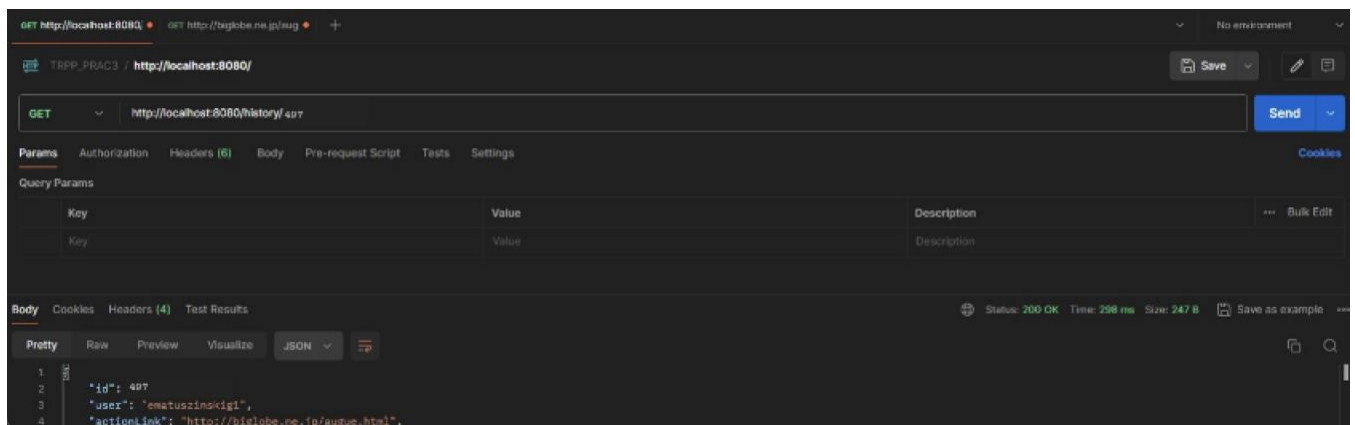


Рисунок 9 – Запрос сущности по идентификатору

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

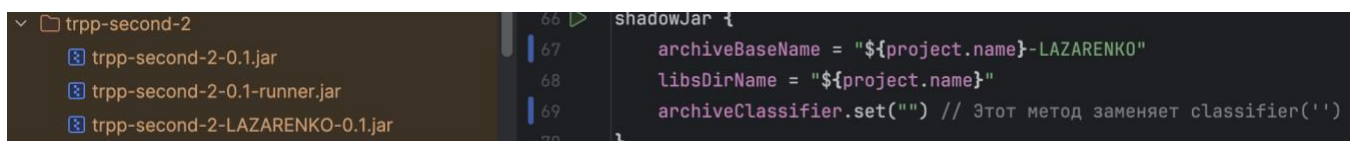


Рисунок 10 – Добавление фамилии к jar-файлу

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

Выполним команду `./gradlew checkstyleMain`:

```
gwynbleidd@MacBook-Air-Sergej-3 trpp-second-2 % ./gradlew checkstyleMain
```

Рисунок 11 – Запуск checkstyleMain

Summary	
Files	Errors
5	7
Files	
Name	Errors
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\controllers\HistoryController.java	2
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\entity\History.java	2
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\Application.java	1
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\controllers\HealthController.java	1
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\entity\HealthResponse.java	1
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\Application.java	
Error Description	Line
Name 'nu mirea trp_second_14' must match pattern "[a-z]+([a-z]([a-z0-9]{1,})?)\$".	1
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\controllers\HealthController.java	
Error Description	Line
Name 'nu mirea trp_second_14 controllers' must match pattern "[a-z]+([a-z]([a-z0-9]{1,})?)\$".	1
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\controllers\HistoryController.java	
Error Description	Line
Name 'nu mirea trp_second_14 controllers' must match pattern "[a-z]+([a-z]([a-z0-9]{1,})?)\$".	1
Line is longer than 120 characters (found 170).	22
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\entity\HealthResponse.java	
Error Description	Line
Name 'nu mirea trp_second_14 entity' must match pattern "[a-z]+([a-z]([a-z0-9]{1,})?)\$".	1
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trp_second_14\entity\History.java	
Error Description	Line
Name 'nu mirea trp_second_14 entity' must match pattern "[a-z]+([a-z]([a-z0-9]{1,})?)\$".	1
Unused import - java util List.	9
Back to top	

Рисунок 12 – Выявленные ошибки

CheckStyle Audit	
Designed for use with CheckStyle and Ant	
Summary	
Files	Errors
5	0
Files	
Name	Errors
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\Application.java	0
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\controllers\HealthController.java	0
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\controllers\HistoryController.java	0
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\entity\HealthResponse.java	0
C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\entity\History.java	0
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\Application.java	
Error Description	Line
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\controllers\HealthController.java	
Error Description	Line
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\controllers\HistoryController.java	
Error Description	Line
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\entity\HealthResponse.java	
Error Description	Line
Back to top	
File C:\Users\Ramesh\IdeaProjects\trp-second-14src\main\java\nu\mirea\trpsecond14\entity\History.java	
Error Description	Line
Back to top	

Рисунок 13 – Отсутствие ошибок

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. 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.

2. 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.

3. 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.

4. 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.

5. Что такое javadoc?

Javadoc – это инструмент, который автоматически генерирует документацию API на основе комментариев в исходном коде Java.

6. Что такое checkstyle?

Checkstyle – инструмент статического анализа кода для языка Java, который используется для обеспечения согласованности и соответствия кода определенным стандартам оформления.

ВЫВОД

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