



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 № 2

по дисциплине

«Технология разработки программных приложений»

Тема: «Системы сборки»

Выполнил студент группы ИКБО-20-21

Алинбеков А.Т.

Принял

Петренко А.А.

Практическая работа выполнена

«31» марта 2023 г.

(подпись студента)

«Зачтено»

«31» марта 2023 г.

(подпись руководителя)

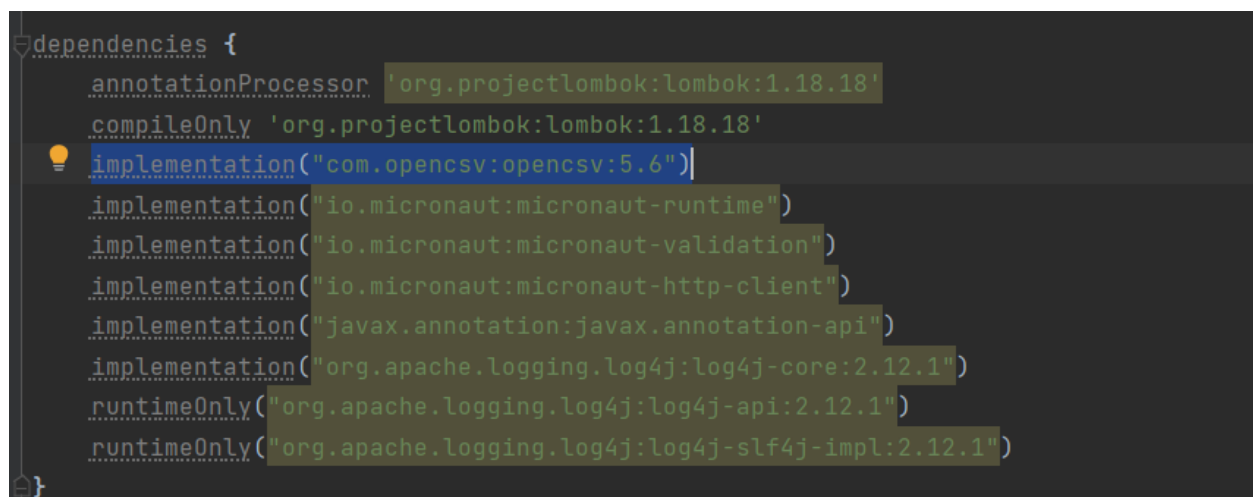
Москва 2023

Ход работы.

Вариант 1 (репозиторий: <https://github.com/rtu-mirea/trpp-second-1>, сущность ru.mirea.entity.Employee)

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

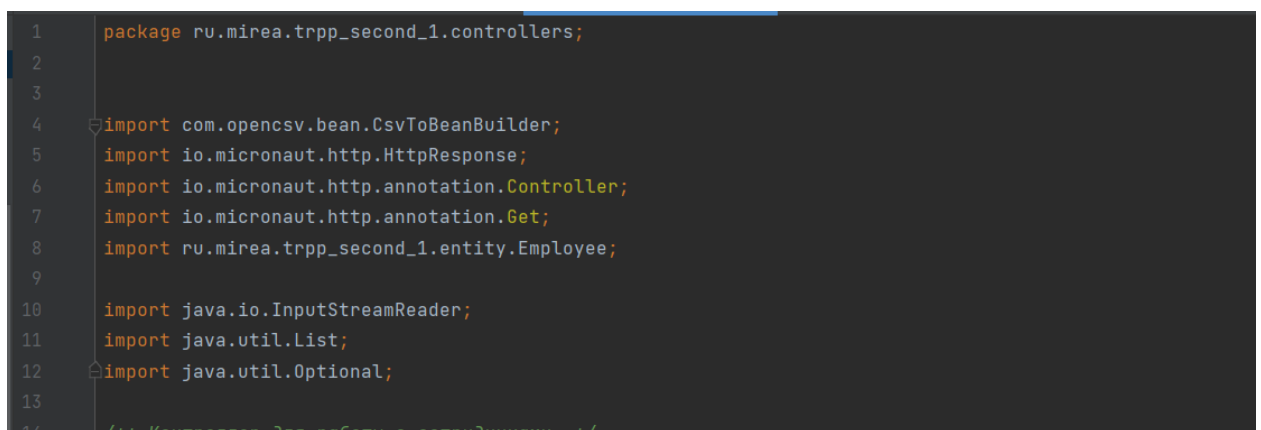
Отсутствующей зависимостью является opencsv, она была добавлена в build.gradle и проект был пересобран – рисунок 1.



```
dependencies {
    annotationProcessor 'org.projectlombok:lombok:1.18.18'
    compileOnly 'org.projectlombok:lombok:1.18.18'
    implementation("com.opencsv:opencsv:5.6")
    implementation("io.micronaut:micronaut-runtime")
    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 – Добавление зависимости opencsv

2. В некоторых классах поправить имя пакета.



```
1 package ru.mirea.trpp_second_1.controllers;
2
3
4 import com.opencsv.bean.CsvToBeanBuilder;
5 import io.micronaut.http.HttpResponse;
6 import io.micronaut.http.annotation.Controller;
7 import io.micronaut.http.annotation.Get;
8 import ru.mirea.trpp_second_1.entity.Employee;
9
10 import java.io.InputStreamReader;
11 import java.util.List;
12 import java.util.Optional;
13
14 /* Контроллер для работы с сотрудниками */
```

Рисунок 2 – Исправление ошибок в названии пакета

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

Был собран Javadoc и найдены запросы состояния и сущности – рисунок 3,4.

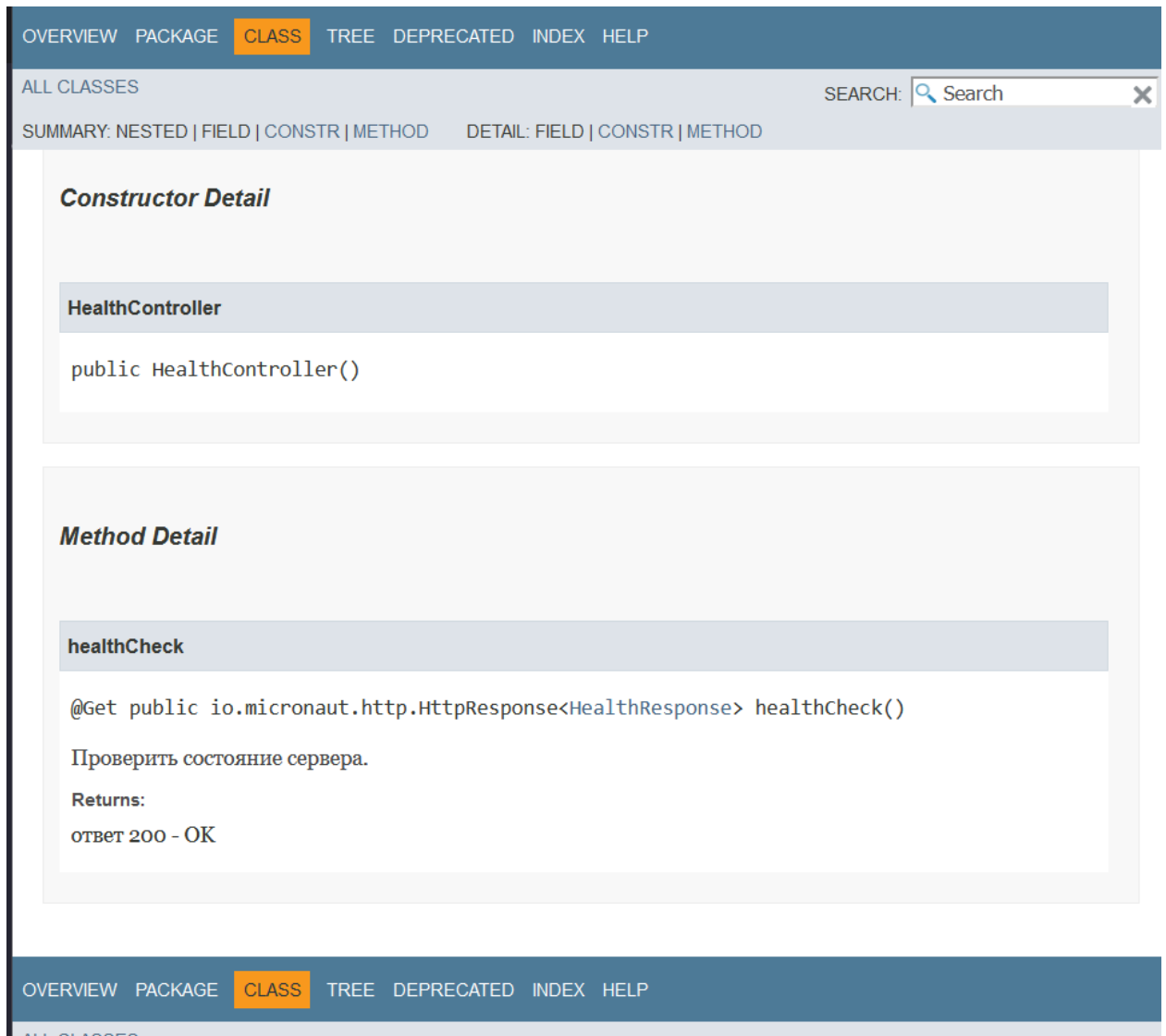


Рисунок 3 - Запрос состояния сервера

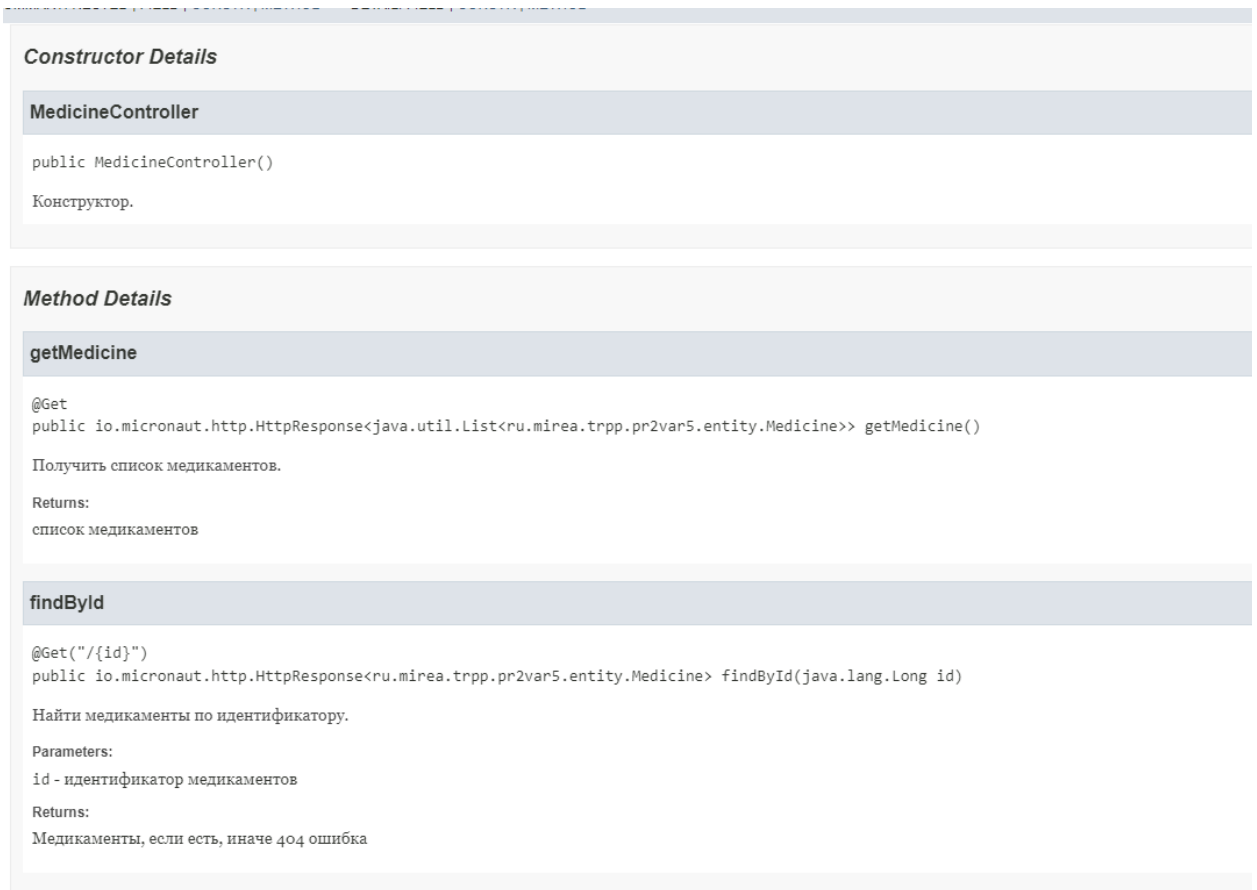


Рисунок 4 – запрос сущности

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

Был собран jar и запущен – рисунок 5,6.

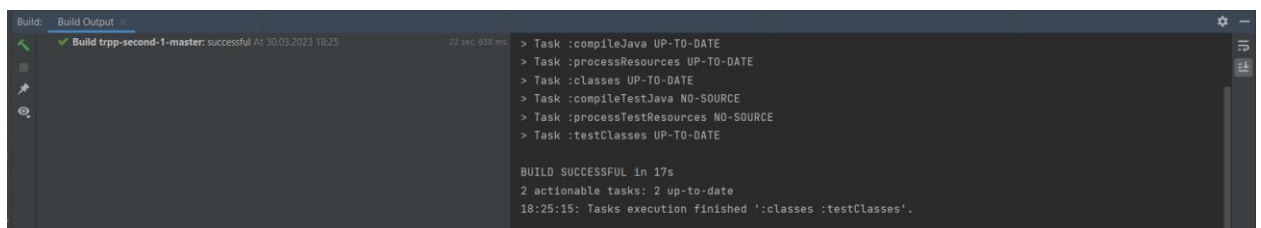


Рисунок 5 - Jar

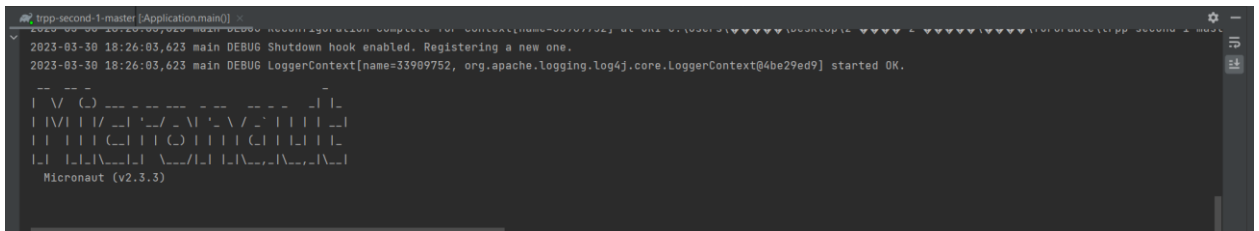


Рис 6 – Запуск приложения

5-6. Запросить состояние запущенного сервера (GET запрос по адресу <http://localhost:8080>). Запросить сущность по идентификатору (GET запрос по адресу: <http://localhost:8080/сущность/идентификатор>)
Идентификатором будут 3 последних цифры в серийном номере вашего студенческого билета.

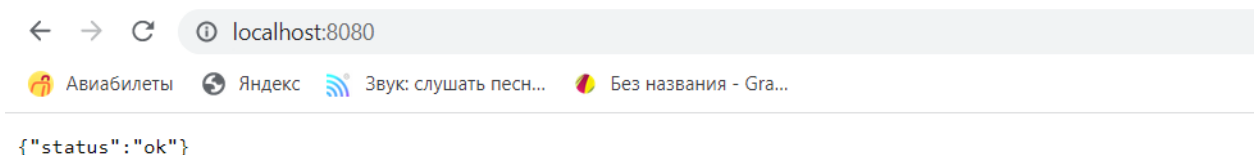


Рисунок 7 – Запрос к серверу

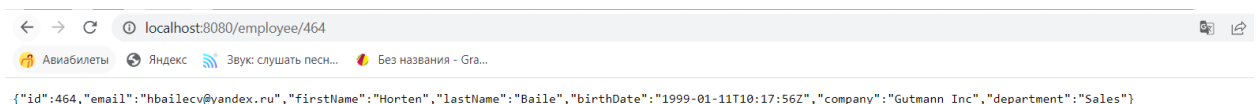


Рисунок 8- Запрос к серверу

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

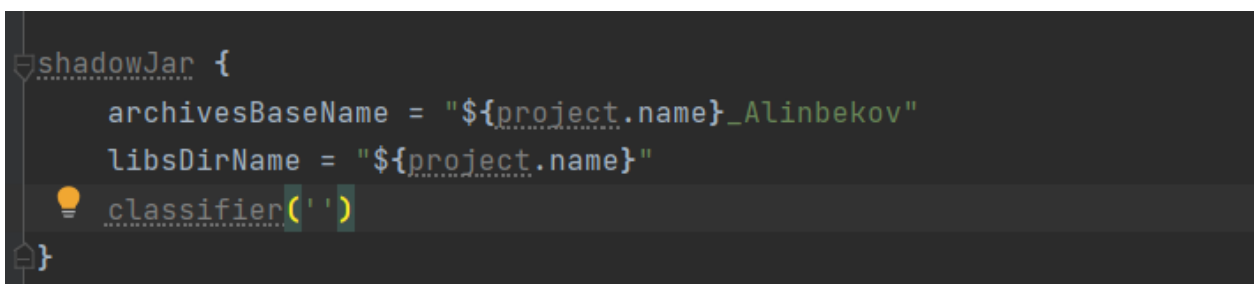


Рисунок 9 – ShadowJar

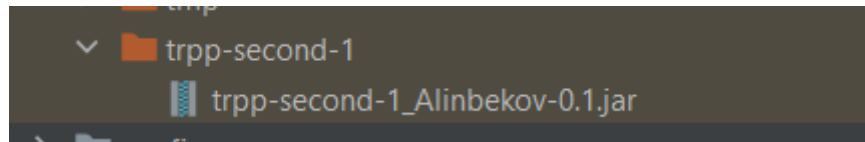


Рисунок 10 – Сгенерированные JAR файлы

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

CheckStyle Audit
Designed for use with [CheckStyle](#) and [Ant](#)

Summary	
Files	Errors
5	6

Files	
Name	Errors
C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\controllers\EmployeeController.java	2
C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\application.java	1
C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\controllers\HealthController.java	1
C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\entity\Employee.java	1
C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\entity\HealthResponse.java	1

File C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\application.java	
Error Description	Line
Name 'ru.mirea.trpp_second_1' must match pattern '^[a-z]{1}([a-z][a-z0-9]{1})*\$'.	1
Back to top	

File C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\controllers\EmployeeController.java	
Error Description	Line
Name 'ru.mirea.trpp_second_1.controllers' must match pattern '^[a-z]{1}([a-z][a-z0-9]{1})*\$'.	1
Line is longer than 120 characters (found 173).	23
Back to top	

File C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\controllers\HealthController.java	
Error Description	Line
Name 'ru.mirea.trpp_second_1.controllers' must match pattern '^[a-z]{1}([a-z][a-z0-9]{1})*\$'.	1
Back to top	

File C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\entity\Employee.java	
Error Description	Line
Name 'ru.mirea.trpp_second_1.entity' must match pattern '^[a-z]{1}([a-z][a-z0-9]{1})*\$'.	1
Back to top	

File C:\Users\A\app\Desktop\2\src\2\main\TPP\forGradle\trpp-second-1-master\src\main\java\ru\mirea\trpp_second_1\entity\HealthResponse.java	
Error Description	Line
Name 'ru.mirea.trpp_second_1.entity' must match pattern '^[a-z]{1}([a-z][a-z0-9]{1})*\$'.	1
Back to top	

Рисунок 11 – Отчет до исправления ошибок

Было изменено название пакета для исправления стилистической ошибки – рисунок 12

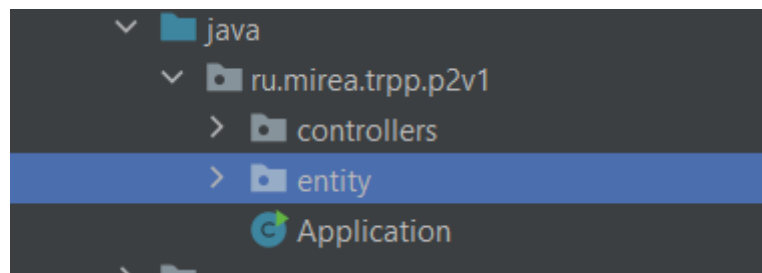


Рисунок 12 – Новое название пакета

```
> Task :processResources UP-TO-DATE
> Task :classes
> Task :checkstyleMain

BUILD SUCCESSFUL in 4s
3 actionable tasks: 2 executed, 1 up-to-date
10:31:17: Task execution finished 'checkstyleMain --stacktrace'.
```

Рисунок 13 – отчет после исправления ошибок

Ответы на вопросы

Вопрос 1 Чем компиляция отличается от системы сборки?

«Сборка» — это довольно общий термин, и он может относиться ко всему, что необходимо для перехода от редактируемого исходного материала (исходный код, сценарии, файлы необработанных данных и т. Д.) К готовому программному продукту. Сборка может (и обычно имеет) несколько этапов, таких как предварительная обработка, компиляция, компоновка, преобразование файлов данных, запуск автоматических тестов, упаковка и т.д.

«Компиляция» более специфична и почти всегда относится к процессу, который принимает исходный код в качестве входных данных и выводит что-то работоспособное, обычно машинный код для физической или виртуальной машины, или исходный код на другом языке.

Вопрос 2 Что такое система сборки?

Система сборки – это программное обеспечение, обеспечивающее автоматизацию сборки проекта. Основное отличие от IDE в том, что конфигурационный файл для системы сборки вы описываете в текстовом виде.

Вопрос 3 Что такое репозиторий?

Репозиторий- место, где хранятся и поддерживаются какие-либо данные. Чаще всего данные в репозитории хранятся в виде файлов, доступных для дальнейшего распространения по сети.

Вопрос 5 Что такое gradle?

Gradle — система автоматической сборки, построенная на принципах Apache Ant и Apache Maven, но предоставляющая DSL на языках Groovy и Kotlin вместо традиционной XML-образной формы представления конфигурации проекта.

Вопрос 6 Что такое Maven?

Maven — это инструмент автоматической сборки проектов, который используется в различных областях разработки программного обеспечения, включая Java-приложения и серверные приложения. Он является популярным инструментом для управления зависимостями, сборки проектов и автоматизации процесса разработки.

Вопрос 7 Что такое mavencentral?

Maven Central — это крупнейший репозиторий для Java-библиотек и плагинов, который используется в системе автоматической сборки Maven. Этот репозиторий хранит множество библиотек и плагинов, которые могут быть использованы в проектах Java.

Вывод

Была изучена система сборки Gradle и ее возможности