**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное автономное образовательное учреждение

высшего образования «Уральский федеральный университет

имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий - РТФ

Центр ускоренного обучения

Отчет по лабораторной работе № 1

по дисциплине «Программирование»

Старший преподаватель: Н.А. Архипов

Студент гр. РИЗ-220938у А.А. Фищев

Екатеринбург 2023

2. Цель работы: Создание простейшего веб-приложения с помощью фреймворка Spring Boot.

3. Описание задачи:

1. 1. Реализовать простейшее Spring Boot приложение
   2. Реализовать дополнительный функционал в разработанном по примеру Spring Boot приложении
   3. Ответить письменно на вопросы в соответствии с номером в списке
   4. Оформить отчет и прикрепить его на сайте https://edu.itlearn.ru/ в соответствующем курсе, в соответствующем разделе
2. Ход выполнения:
   1. Реализовал Spring Boot приложение
   2. Реализовал дополнительный функционал приложения
   3. Ответил письменно на вопросы
      1. Чем отличаются системы сборки maven и gradle?

Maven и Gradle — это две разные среды разработки программного обеспечения, которые используются для автоматизации процесса сборки приложений на Java. Основное отличие между Maven и Gradle заключается в том, что Maven использует систему контроля версий (например, JIRA), чтобы управлять зависимостями проекта, а также предоставляет возможность настройки конфигурации приложения. Gradle же является более простым инструментом для автоматической сборки и управления проектами. Кроме того, Maven имеет больше возможностей для настройки конфигурации приложения, таких как использование плагинов или интеграция с другими инструментами. В то время как Gradle предлагает более простой интерфейс для создания новых проектов и настройки существующих. В целом, выбор между Maven и Gradle зависит от конкретных потребностей разработчика и его предпочтений по отношению к инструментам для сборки приложений

* + 1. Чем отличаются языки программирования Java и Groovy?

Java и Groovy — это два разных языка программирования, которые имеют некоторые сходства, но также и отличия. Вот несколько основных различий между ними: Синтаксис: Java использует синтаксис, основанный на правилах, а Groovy — более гибкий язык с возможностью использования блоков кода. Типизация: В Java все переменные должны быть объявлены как объекты класса, что делает его более строгим языком по сравнению с Groovy. В то время как Groovy позволяет использовать переменные без явного указания типа. Библиотеки: Java имеет большое количество библиотек для работы со стандартными классами операционной системы (например, JDBC), тогда как Groovy не имеет такой же большой библиотеки. Грамматика: Оба языка имеют похожую грамматику, но есть различия в том, какие конструкции могут использоваться и как они обрабатываются. Безопасность: В Java используется строгая типизация, которая требует проверки типов при выполнении операций. В Groovy можно использовать переменные без явного указания типа, что может привести к ошибкам. В целом, выбор между Java и Groovy зависит от конкретных потребностей разработчика и требований проекта. Если вам нужна строгость и безопасность, то лучше выбрать Java. Если же вы хотите иметь больше гибкости и возможности использовать переменные без явного указания типа, то Groovy может подойти лучше.

* + 1. Что нового добавилось в Java 11 в отличии от Java 8?

1. Новый класс: Apache Commons Lang был добавлен в Java 11, предоставляя набор инструментов для обработки строк, коллекций и других типов данных. 2. Функция высшего порядка: В Java 11 была добавлена новая функция высшего порядка, называемая "super", которая позволяет вызывать методы других объектов напрямую, без необходимости создания объекта-обертки. 3. Многопоточная обработка: В Java 11 была добавлена улучшенная поддержка многопоточной обработки, включая новые классы для работы с многопоточностью и улучшенную производительность. 4. Объектно-ориентированное программирование: В Java 11 была добавлена улучшенная поддержка объектно-ориентированного программирования, включая новые классы для работы с наследованием и улучшенную поддержку полиморфизма. 5. Виртуальная машина: В Java 11 была добавлена улучшенная поддержка виртуальной машины, включая новые классы для работы с виртуальными машинами и улучшенную производительность. 6. Стандарт XML: В Java 11 была добавлена улучшенная поддержка стандарта XML, позволяя создавать структурированные данные и работать с XML-документами. 7. Расширенная поддержка безопасности: В Java 11 была добавлена улучшенная поддержка безопасности, включая новые классы для работы с безопасностью и улучшенную производительность.

* 1. Так же прикрепляю ссылку на [GIT РЕПОЗИТОРИЙ](https://github.com/papakarlorabotaet/lrForStudySecondYear)

5. Вывод: за время выполнения лабораторной работы укрепил навыки программирования и создал простейшее веб-приложение с помощью фреймворка SpringBoot.

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание