|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Инструментального и прикладного программного обеспечения

**ПРАКТИЧЕСКАЯ РАБОТА №5-7**

по дисциплине «Разработка серверных частей интернет-ресурсов»

**Студент группы** ИКБО-10-21 Никонова Е. А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель практической работы** ассистентБлагирев М. М.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Допущен к работе «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Москва 2023

Содержание

[Практическая работа № 5 3](#__RefHeading___Toc835_1872065658)

[Практическая работа № 6 11](#__RefHeading___Toc835_18720656581)

[Список использованной литературы: 17](#__RefHeading___Toc926_4009055764)

# Практическая работа № 5

**Теоретическое введение**

Java — это строго типизированный объектно-ориентированный язык программирования. Приложения на Java транслируются в байт-код, поэтому их можно запускать на любом устройстве, где установлена виртуальная машина Java (JVM).

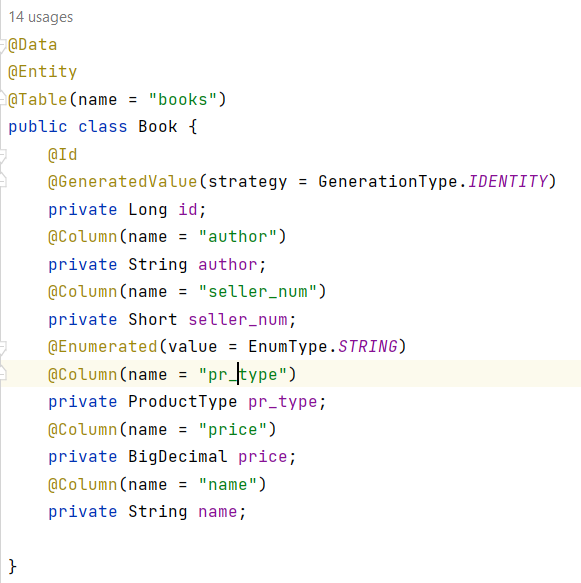
По сути Spring Framework представляет собой просто контейнер внедрения зависимостей, с несколькими удобными слоями (например: доступ к базе данных, прокси, аспектно-ориентированное программирование, RPC, веб-инфраструктура MVC). Это все позволяет вам быстрее и удобнее создавать Java-приложения.

Несмотря на огромное число разнообразных сайтов, практически всю веб разработку можно свести к CRUD операциям. CRUD – широко распространенный термин, означающий 4 стандартные операции над любой сущностью (ресурсом): создание, чтение, обновление и удаление.

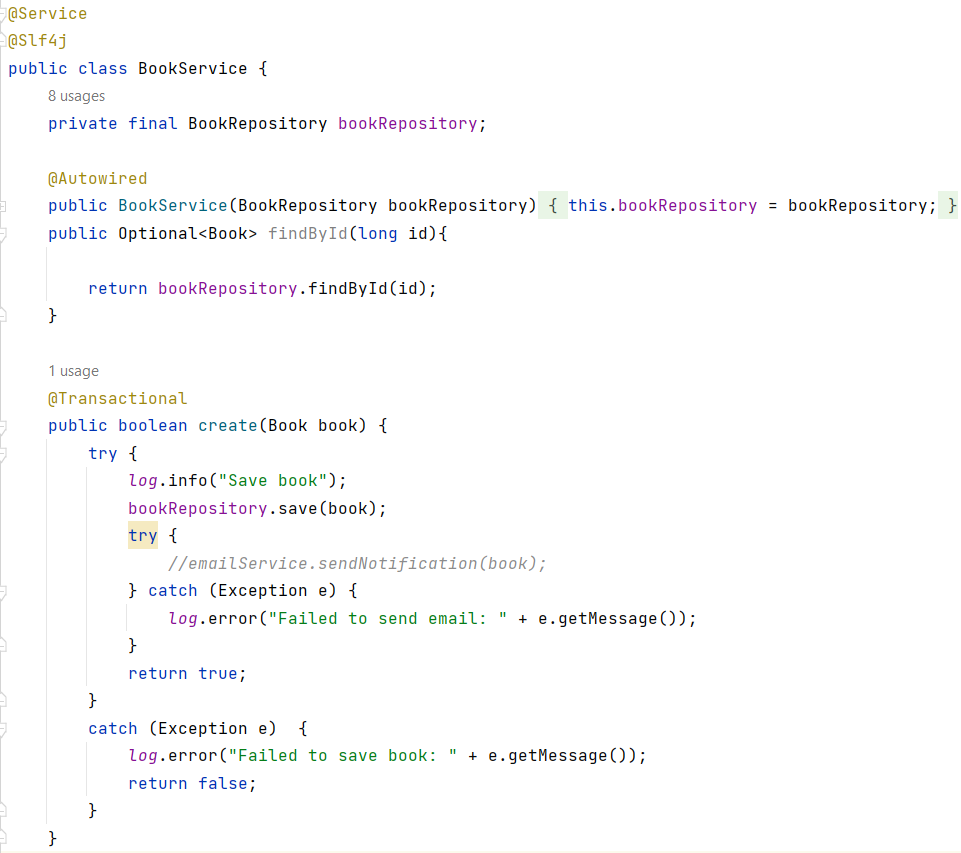
**Цель работы:** создать серверное CRUD-приложение — основу для маркетплейса.

**Ход работы:**

1. Создадим модели для сущности книга, клиент, стиральная машина, телефон (рис. 1).

Рисунок 1 — Модель сущности «Книга»

2. Создадим сервис для сущностей книга, клиент, стиральная машина, телефон (рис. 2).

Рисунок 2 — Сервис для сущности «Книга»

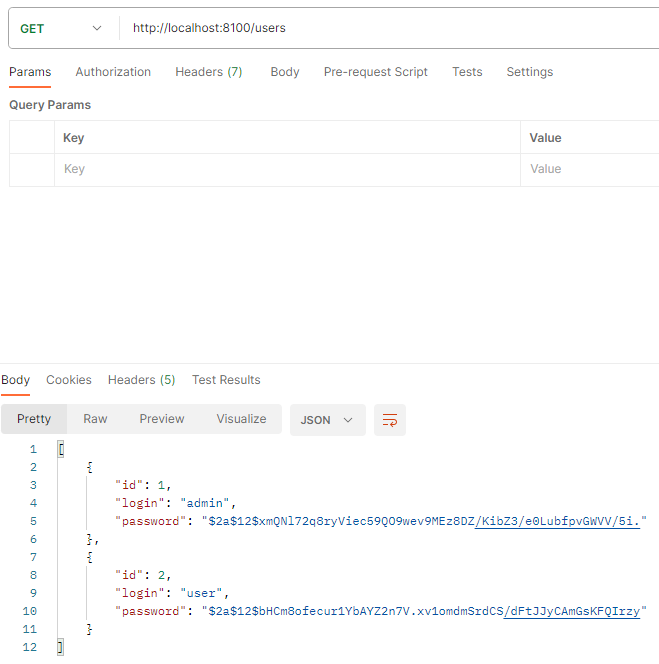
3. Создадим контроллер для управления сущностями книга, клиент, стиральная машина, телефон (рис. 3).

Рисунок 3 — Контроллер для сущности «Книга»

4. Создадим файл docker-compose для созданного приложения (рис. 4).

Рисунок 4 — Файл docker-compose

5. Протестируем работоспособность созданного приложения при помощи Postman (рис. 5).

Рисунок 5 — GET-запрос к сущности User

**Вывод:** было создано серверное CRUD-приложение — основа для маркетплейса.

**Ответы на вопросы:**

1. Что такое JDBC и JPA, в чем отличия.

JDBC позволяет нам писать команды SQL для чтения данных и обновления данных в реляционной базе данных. JPA, в отличие от JDBC, позволяет разработчикам создавать программы Java, управляемые базой данных, с использованием объектно-ориентированной семантики. Аннотации JPA описывают, как данный класс Java и его переменные сопоставляются с данной таблицей и ее столбцами в базе данных.

2. Основные интерфейсы для реализации Spring Data.

Основное понятие в Spring Data — это репозиторий. Это несколько интерфейсов которые используют JPA Entity для взаимодействия с ней. Репозитории Spring Data обычно расширяются за счет интерфейсов Repository или CrudRepository.

3. Аннотации @Entity, @Table. В чем разница.

Аннотация @Entity говорит Hibernate о том, что этот класс является сущностью. Аннотация @Table позволяет указать название таблицы в БД, которая связана с этой сущностью. Если имя класса и имя таблицы совпадает, то эту аннотацию можно не указывать.

4. Что такое Lombok. Приведите несколько примеров использования.

Lombok – это основанная на аннотациях библиотека Java, позволяющая сократить шаблонный код. В Lombok предлагаются различные аннотации, цель которых – заменить ненужный повторяющийся код, писать который утомительно.

Когда после аннотировано @Getter и/или @Setter, Lombok автоматически сгенерирует заданный по умолчанию геттер и/или сеттер, соответственно. Когда класс аннотирован @ToString, Lombok позаботится о том, чтобы сгенерировать нужную реализацию метода toString(). По умолчанию будет возвращаться строка, в которой содержится имя класса, за которой следуют значения всех полей, разделенные запятыми. Можно аннотировать при помощи @NonNull компонент записи, параметр, метод или целый конструктор, В таком случае Lombok сгенерирует для вас команды для проверки на нуль.

5. Назовите способы инициализации бинов в Java.

Аннотация @Bean используется для указания того, что метод создает, настраивает и инициализирует новый объект, управляемый Spring IoC контейнером.

6. Что такое миграции и как они работаю в библиотеке Flyway.

В терминологии flyway любые изменения в базе данных называются миграциями. Миграции могут быть *версионированными* (versioned) и *повторяемыми (*repeatable).

# Практическая работа № 6

**Теоретическое введение**

При разработке систем с пользовательским интерфейсом, следуя паттерну MVC нужно разделять систему на три составные части. Их, в свою очередь, можно называть модулями или компонентами. Говори как хочешь, но дели на три. У каждой составной компоненты будет свое предназначение.

Model. Первая компонента/модуль — так называемая модель. Она содержит всю бизнес-логику приложения.

View. Вторая часть системы — вид. Данный модуль отвечает за отображение данных пользователю. Все, что видит пользователь, генерируется видом.

Controller. Третьим звеном данной цепи является контроллер. В нем хранится код, который отвечает за обработку действий пользователя (любое действие пользователя в системе обрабатывается в контроллере).

**Цель работы:** добавить к существующему серверному CRUD-приложению реализацию «корзины».

**Ход работы:**

1. Создадим модель для сущности корзина (рис. 6).

Рисунок 6 — Модель сущности «Корзина»

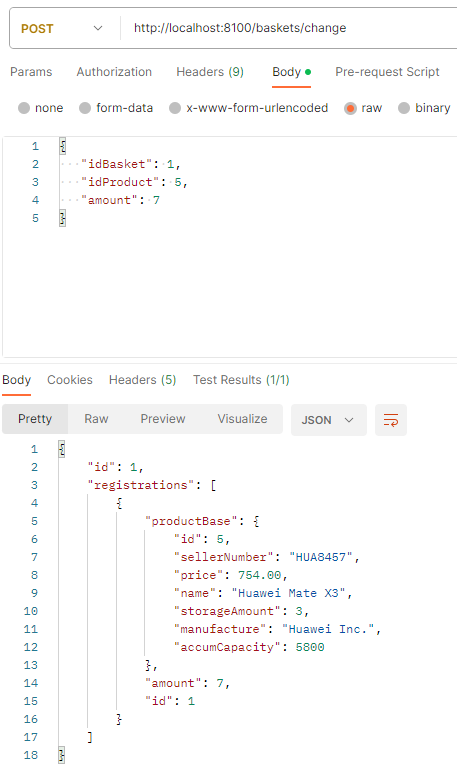
2. Создадим сервис для сущности «корзина» (рис. 7).

Рисунок 7 — Сервис сущности «Корзина»

3. Создадим контроллер для управления сущностью «корзина» (рис. 8).

Рисунок 8 — Контроллер сущности «Корзина»

4. Протестируем работоспособность приложения при помощи Postman (рис. 9).

Рисунок 9 — POST-запрос к сущности Basket

**Вывод:** к ранее созданной основе маркетплейса была добавлена функция корзины.

**Ответы на вопросы:**

1. Аннотация Service.

Эта аннотации, которая выделяет слой бизнес логики — сервисы. Аннотация @Service создает слой для реализации бизнес логики. Классы, на которых весит эта аннотация, автоматически становятся bean’ами.

2. Паттерн MVC.

MVC — это именно набор архитектурных идей и принципов для построения сложных систем с пользовательским интерфейсом.

При разработке систем с пользовательским интерфейсом, следуя паттерну MVC нужно разделять систему на три составные части.

Model. Первая компонента/модуль — так называемая модель. Она содержит всю бизнес-логику приложения.

View. Вторая часть системы — вид. Данный модуль отвечает за отображение данных пользователю. Все, что видит пользователь, генерируется видом.

Controller. Третьим звеном данной цепи является контроллер. В нем хранится код, который отвечает за обработку действий пользователя (любое действие пользователя в системе обрабатывается в контроллере).

3. Паттерн «Чистая архитектура».

**Основная идея заключается в том, чтобы разделить систему на уровни, основанные на ценности для бизнеса**. Следовательно, самый высокий уровень имеет бизнес-правила, причем каждый нижний уровень приближается к устройствам ввода-вывода.

4. Описать все слои RestFull приложения и описать зачем каждый из них.

Веб-сервер является уровнем представления и предоставляет пользовательский интерфейс. Обычно это веб-страница или веб-сайт, но никто не запрещает использование мобильных или декстопных приложений. Этот уровень по существу взаимодействует с клиентом, отображая данные различного рода. Лишь немногие из логики приложения выполняются на стороне клиента, имея ограниченные возможности;

Сервер приложений соответствует среднему уровню, в котором размещена бизнес-логика, используемая для обработки пользовательских вводов. По сути, он играет роль объединения. При технической обработке различных входных данных и выборок, полученных клиентами, он играет роль взаимодействия с обширной базой данных, представленной на следующем уровне;

Сервер базы данных – это уровень данных или серверной части веб-приложения.

# Практическая работа № 7

**Теоретическое введение**

Spring Security – это мощная и настраиваемая платформа аутентификации и контроля доступа. Это стандарт де-факто для обеспечения безопасности приложений на базе Spring.

JSON Web Token (JWT) — это JSON объект, который определен в открытом стандарте [RFC 7519](https://tools.ietf.org/html/rfc7519). Он считается одним из безопасных способов передачи информации между двумя участниками. Для его создания необходимо определить заголовок (header) с общей информацией по токену, полезные данные (payload), такие как id пользователя, его роль и т.д. и подписи (signature).

Микросервисная архитектура — это подход, который помогает не только ускорить разработку продукта, но и сделать ее гибкой и управляемой: проект из неделимого целого превращается в систему связанных между собой блоков — сервисов.

Приложение с микросервисной архитектурой разделено на небольшие не зависящие друг от друга компоненты — микросервисы. У каждого из них своя бизнес-задача: например, управлять каталогом, хранить и обновлять содержимое корзины или проводить оплату заказа.

**Цель работы:** создать приложение на микросервисной архитектуре, где один сервис — это приложение шестой практики, а второй сервис — это сервис авторизации.

**Ход работы:**

1. Создадим модель для сущности корзина (рис. 6).

**Вывод:** было создано приложение на микросервисной архитектуре, где один сервис — это приложение шестой практики, а второй сервис — это сервис авторизации.

**Ответы на вопросы:**

1. Spring Security.

Spring Security – это мощная и настраиваемая платформа аутентификации и контроля доступа. Это стандарт де-факто для обеспечения безопасности приложений на базе Spring.

Spring Security – это платформа, которая фокусируется на обеспечении аутентификации и авторизации для приложений Java. Как и во всех проектах Spring, реальная мощь Spring Security заключается в том, насколько легко ее можно расширить для удовлетворения пользовательских требований.

2. Что такое JWT Token и как он работает.

JSON Web Token (JWT) — это JSON объект, который определен в открытом стандарте [RFC 7519](https://tools.ietf.org/html/rfc7519). Он считается одним из безопасных способов передачи информации между двумя участниками. Для его создания необходимо определить заголовок (header) с общей информацией по токену, полезные данные (payload), такие как id пользователя, его роль и т.д. и подписи (signature).

Простыми словами, JWT — это лишь строка в следующем формате: header.payload.signature.

3. Микросервисная архитектура. Плюсы и минусы перед монолитной системой.

Микросервисная архитектура — это подход, который помогает не только ускорить разработку продукта, но и сделать ее гибкой и управляемой: проект из неделимого целого превращается в систему связанных между собой блоков — сервисов.

Приложение с микросервисной архитектурой разделено на небольшие не зависящие друг от друга компоненты — микросервисы. У каждого из них своя бизнес-задача: например, управлять каталогом, хранить и обновлять содержимое корзины или проводить оплату заказа.

Благодаря тому, что части приложения автономны, его, как и любую распределённую систему, легко развивать и обновлять: добавление или улучшение отдельных функций никак не повлияет на остальные компоненты. И это главное отличие микросервисного приложения от монолитного, в котором все блоки кода связаны между собой, и даже небольшие изменения хотя бы в одном из них поменяют работу всей системы.

4. Варианты масштабируемости.

Крайне важно понять, что такое масштабируемость, прежде чем обсуждать вертикальное и горизонтальное масштабирование. Количество клиентских запросов, которые приложение может обрабатывать одновременно, является мерой его масштабируемости. Тогда термин “предел масштабируемости” относится к точке, в которой аппаратный ресурс исчерпан и больше не может обслуживать запросы.

**Масштабируемость базы данных — это способность базы данных управлять изменяющимися запросами путем добавления и удаления данных.** Когда ресурс исчерпан, программа больше не может обрабатывать новые запросы. Затем администраторы могут масштабировать инфраструктуру, добавляя дополнительные ресурсы, такие как оперативная память, центральный процессор, хранилище и сетевые устройства, для эффективного удовлетворения дополнительных требований.

5. Что такое вертикальная масштабируемость

Вертикальное масштабирование — это разновидность масштабируемости, при которой производительность машины повышается за счет добавления дополнительных вычислительных мощностей. Кроме того, вертикальное масштабирование помогает нам расширить пропускную способность машины, сохраняя ресурсы внутри одного логического блока.

6. Что такое горизонтальная масштабируемость

Итак, горизонтальное масштабирование, по определению, относится к практике добавления дополнительных устройств в инфраструктуру с целью увеличения пропускной способности и эффективного управления растущими потребностями в трафике. **Действительно, горизонтальное масштабирование, как следует из названия, - это процесс увеличения пропускной способности по горизонтали за счет добавления дополнительных серверов.** Затем балансировщик нагрузки распределяет нагрузку и вычислительную мощность между многочисленными серверами в системе.

7. Объяснить принцип сбора микросервисной архитектуры в Docker.

Развертывание и оркестровка контейнеров Docker - неотъемлемая часть управления архитектурой микросервисов. Средства оркестровки контейнеров автоматизируют развертывание, управление и масштабирование отдельных контейнеров, обеспечивая эффективную совместную работу микросервисов.

8. NoSQL DB. Почему в сервисе авторизации удобнее использовать Redis.

9. Многопотчность — что это такое. Как оно работает в Java.

Java является многопоточным языком программирования, а это значит, что с помощью него мы можем разрабатывать многопоточные программы. Многопоточная программа состоит из двух или более частей, которые могут выполняться одновременно, и каждая часть может одновременно обрабатывать разные задачи, оптимально используя доступные ресурсы, особенно если компьютер имеет несколько ЦП.

10. Класс Thread, расскажите про его методы.

Чтобы запустить поток, необходимо вызвать специальный метод start(). Для приостановки выполнения текущего потока на какое-то время, используем метод sleep(). Метод join() приостанавливает выполнение текущего потока до тех пор, пока не завершится другой поток.

**Ссылка на удаленный репозиторий проекта:** [**Elisaveta-N/rschir (github.com)**](https://github.com/Elisaveta-N/rschir)

# Список использованной литературы:

1. Документация // PHP URL: [https://www.php.net/manual/ru/index.php](https://vk.com/away.php?to=https%3A%2F%2Fwww.php.net%2Fmanual%2Fru%2Findex.php&cc_key=) (дата обращения: 16.10.2021). – Текст: электронный.

2. Дергачев, А. М. Проблемы эффективного использования сетевых сервисов / А. М. Дергачев // Научно-технический вестник СПбГУ ИТМО. –2011. – № 1 (71). С. 83-87.

3. Розенфельд, Л. Информационная архитектура в Интернете / Л. Розенфельд, П. Морвиль, 2 е издание. – Пер. с англ. – СПб: Символ Плюс, 2005 – 544 с.

4. Спинеллис, Д. Идеальная архитектура. Ведущие специалисты о красоте программных архитектур / Д. Спинеллис, Г. Гусиос. – Пер. с англ. –СПб.: Символ Плюс, 2010 – 528 с.

5. Фаулер, М. Архитектура корпоративных программных приложений / М. Фаулер Пер. с англ. − М.: Издательский дом "Вильяме", 2006 − 544 с.

6. Информационные технологии / О.Л. Голицына, Н.В. Максимов, Т.Л. Партыка, И.И. Попов. − 2-е изд. − Москва: ФОРУМ − ИНФРА-М, 2008. – 395 с.

7. Таненбаум, Э. Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. ван Стеен. − СПб.: Питер, 2003. − 93 с.

8. Котеров, Д.В. PHP 7 / Д.В. Котеров, И. В. Симдянов. - СПб.: БХВПетербург, 2021. – 1088 с.