
УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«25» мая 2026 г.
Протокол № 2

**Рабочая программа дисциплины (модуля)
«Java-фреймворк Spring»**

Направление подготовки: 02.04.01 Математика и компьютерные науки

Направленность (профиль) подготовки: Программная инженерия в
ритейле

Квалификация (степень) выпускника: магистр

Форма обучения: очная

Срок освоения программы: 2 года

Год набора: 2026

**Москва
2026**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения.....	3
3. Тематический план.....	7
4. Содержание дисциплины (модуля).....	7
5. Учебно-методическое обеспечение	8
6. Материально-техническое обеспечение	8
7. Методические и оценочные материалы	10

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Java-фреймворк Spring» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – магистратура по направлению 02.04.01 Математика и компьютерные науки, профиль Программная инженерия в ретейле, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 810 от 23.08.2017 года.

Изучение дисциплины (модуля) «Java-фреймворк Spring» позволяет сформировать целостное понимание экосистемы Spring и её роли в разработке enterprise-приложений, а также практическую компетенцию в создании масштабируемых, поддерживаемых backend-решений.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки магистратуры по направлению 02.04.01 Математика и компьютерные науки, профиль Программная инженерия в ретейле и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений как дисциплина по выбору. Дисциплина (модуль) доступна к изучению на 1 курсе во 2 семестре.

Цель изучения дисциплины (модуля): формирование у студентов глубокого понимания архитектуры и принципов работы фреймворка Spring, а также развитие практических навыков разработки современных backend-приложений на платформе Spring Boot с использованием стандартных и продвинутых подходов.

Задачи изучения дисциплины (модуля):

- освоить ключевые концепции Spring: IoC, DI, контекст, бины, аннотации и автоматическую конфигурацию;
- научиться проектировать и реализовывать RESTful API с обработкой запросов, исключений и данных;
- овладеть технологиями доступа к данным: JDBC, JPA (Hibernate), Spring Data;
- развить навыки работы с реактивными типами (Mono/Flux) и событийно-ориентированными архитектурами;
- сформировать умение создавать, настраивать и разворачивать полноценные Spring Boot-приложения.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- базовые принципы и компоненты экосистемы Spring: для чего нужны Inversion of Control (IoC), Dependency Injection (DI), Spring Context, бины, и чем Spring Boot отличается от Spring Framework;
- ключевые подходы к работе с данными: понимает разницу между JDBC и JPA (Hibernate);
- принципы реализации и работы RESTful API;
- основы реактивной парадигмы (Mono/Flux) и событийно-ориентированной архитектуры (EDA/Kafka).

уметь:

- создавать приложения на Spring Boot, конфигурировать их, управлять биными;
- реализовывать RESTful API: создавать эндпоинты для обработки запросов и формирования ответов (включая работу с DTO, файлами, исключениями);
- подключаться к базам данных и работать с ними из Spring-приложений (JDBC/JPA, миграции).

владеть:

- навыком создавать полноценные приложения на Spring Boot "с нуля": от инициализации проекта через Spring Initializr до развертывания рабочего

backend-приложения;

- навыком применять продвинутые возможности Spring Data JPA для эффективной работы с БД: создавать запросы с помощью методов репозитория, аннотаций `@Query`, понимать их ограничения и знать альтернативы (JOO).

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
ОПК-1.	Способен находить, формулировать и решать актуальные и значимые проблемы прикладной и компьютерной математики	ОПК-1.1.	Знает современные методы прикладной математики, а также алгоритмические основы решения задач в условиях реальных данных ретейла; владеет инструментами и технологиями, применяемыми для анализа и обработки больших объёмов информации (включая отечественные программные средства)
		ОПК-1.2.	Умеет формализовать бизнес-задачи ретейла в виде математических моделей, выбирать и применять алгоритмы и методы решения, интерпретировать результаты и представлять их в форме, доступной для технических и управленческих стейкхолдеров
		ОПК-1.3.	Имеет практический опыт решения прикладных задач в сфере ретейла в рамках учебных, научных или проектных работ, включая обработку реальных данных, разработку аналитических моделей и интеграцию решений в программные системы, а также участие в командных проектах, направленных на повышение эффективности бизнес-процессов
ОПК-3.	Способен самостоятельно создавать прикладные программные средства на основе современных информационных технологий и сетевых ресурсов, в том числе отечественного производства	ОПК-3.1.	Знает принципы проектирования и архитектуры программного обеспечения, современные языки программирования, фреймворки и инструменты разработки, а также особенности применения отечественных программных продуктов и платформ в условиях цифровой экономики
		ОПК-3.2.	Умеет разрабатывать, тестировать и сопровождать

			прикладные программные средства, интегрировать их с базами данных, аналитическими системами и внешними сервисами, а также адаптировать программные решения под требования бизнеса в сфере ретейла
		ОПК-3.3.	Имеет практический опыт разработки программных модулей или систем, используемых для реализации математических моделей и анализа данных в ретейле, включая участие в командных проектах, сопровождение кодовой базы, документирование решений и внедрение в производственную среду
ПК-6.	Способен представлять результаты профессиональной деятельности и формулировать практические рекомендации по развитию цифровых решений в ретейле	ПК-6.1.	Знает требования к подготовке технической и аналитической документации, принципы визуализации данных и стандарты профессиональной коммуникации
		ПК-6.2.	Умеет формулировать обоснованные выводы и рекомендации по результатам разработки и анализа, адаптируя их для технической и управленческой аудитории
		ПК-6.3.	Имеет практический опыт подготовки отчетов, презентаций и публичной защиты результатов проектов в области программной инженерии в ретейле

3. Тематический план

№ п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контроль	Самостояте льная работа	
		Лекции	Семинары (практичес кие занятия)			
1	Введение в Spring. Spring Boot	7	7		32	Домашние задания
2	Работа с API и базами данными	9	9		35	Домашние задания
3	Тестирование и профилирование	6	6		30	Домашние задания
4	Реактивный Spring	8	8		31	Домашние задания
	Экзамен			2		
Итого:		30	30	2	128	
Объем дисциплины (модуля) (в ак. ч.)		190				
Объем дисциплины (модуля) (в зач. ед.)		5				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Введение в Spring. Spring Boot	Экосистема Spring. Spring Framework. Spring Boot Работа с бинами. Жизненный цикл бинов Стереотипные аннотации. Конфигурирование
2	Работа с API и базами данными	RESTful API. Spring MVC. Обработка запросов RESTful API. Spring MVC. Формирование ответов Работа с БД. Spring JDBC. JdbcTemplate Работа с БД. Spring Data JPA. Hibernate Продвинутые возможности Spring Data JPA Spring Security Выполнение запросов. RestTemplate. HttpClient
3	Тестирование и профилирование	Тестирование Spring-приложений Профилирование и мониторинг Spring-приложений
4	Реактивный Spring	Реактивный Spring. Spring WebFlux Event-driven architecture. Работа с Kafka Асинхронная работа с базами данных

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Васильев А. Н. Java. Объектно-ориентированное программирование. — СПб. : Питер, 2021. <https://znanium.ru/catalog/document?id=441974>.
2. Леонард А. Java. Решение практических задач. — СПб. : БХВ-Петербург, 2021. <https://znanium.ru/catalog/document?id=441975>.

Дополнительная литература:

1. Оукс С. Эффективный Java. Тюнинг кода на Java 8, 11 и дальше. — 2-е междунар. изд. — СПб. : Питер, 2021. <https://znanium.ru/catalog/document?id=378710>.

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- механическими калькуляторами;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое

Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Java-фреймворк Spring» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, домашние задания, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар – это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал, использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Java-фреймворк Spring»

Оценивание уровня учебных достижений обучающихся по дисциплине (модулю) осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме **экзамена**, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом, если иное не указано в силлабусе дисциплины:

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
9	Отлично	
8	Отлично	
7	Хорошо	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	
5	Удовлетворительно	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	
3	Не сдан	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	
1	Не сдан	

Дисциплина (модуль) «Java-фреймворк Spring» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашние задания	70%	5	Набор задач по темам недели
Экзамен	30%	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

Формула расчёта итоговой оценки по дисциплине (модулю) «Java-фреймворк Spring»: « $0,7 \times \text{среднее за домашние задания} + 0,3 \times \text{экзамен}$ ».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание 1: Введение в Spring Boot и управление бинами

1. Создайте Spring Boot-приложение с помощью Spring Initializr (веб-интерфейс или IDE).
2. Реализуйте три сервиса: UserService, OrderService, NotificationService, используя аннотации @Service, @Component, @Repository.
3. Настройте внедрение зависимостей через @Autowired.
4. Продемонстрируйте жизненный цикл бина: используйте @PostConstruct и @PreDestroy.
5. Создайте конфигурационный класс с @Configuration и определите бин вручную через @Bean.

Домашнее задание 2: REST API и работа с базой данных

1. Создайте REST-контроллер с эндпоинтами: GET /users, GET /users/{id}, POST /users, PUT /users/{id}, DELETE /users/{id}.
2. Реализуйте DTO-классы для передачи данных (например, UserDto, UserCreateDto).
3. Подключите базу данных H2 (в памяти) и реализуйте доступ через Spring Data JPA.
4. Создайте репозиторий UserRepository и добавьте методы: findByEmail, findByActiveTrue.
5. Обработайте исключения с помощью @ControllerAdvice и @ExceptionHandler.

Домашнее задание 3: Реактивность, события и тестирование

1. Перепишите один из контроллеров на WebFlux (используя Mono и Flux).
2. Настройте Kafka: создайте топик, отправьте и получите сообщение через KafkaTemplate и @KafkaListener.
3. Реализуйте асинхронную отправку уведомления при создании пользователя (@Async).
4. Напишите модульные тесты для сервиса с использованием @SpringBootTest и @MockBean.
5. Настройте профилирование: включите Actuator и проверьте эндпоинты /health, /metrics.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Какой принцип позволяет Spring управлять объектами, а не программисту?	Inversion of Control (IoC)	ОПК-1
2.	Какой механизм используется для передачи зависимостей в объекты Spring?	Dependency Injection (DI)	ОПК-3
3.	Что представляет собой основной контейнер, управляющий бинами в Spring?	Spring Context	ПК-6
4.	Какая аннотация используется для автоматического внедрения зависимостей?	@Autowired	ОПК-1
5.	Какой компонент Spring Boot автоматически настраивает приложение на основе зависимостей?	авто-конфигурация	ОПК-3
6.	Какая аннотация помечает класс как бин-сервис в Spring?	@Service	ПК-6
7.	Какой интерфейс в Spring Data JPA позволяет реализовать CRUD-операции без написания кода?	JpaRepository	ОПК-1
8.	Какой класс используется для выполнения SQL-запросов вручную в Spring JDBC?	JdbcTemplate	ОПК-3
9.	Какой аннотацией помечается REST-контроллер в Spring MVC?	@RestController	ПК-6
10.	Какой метод HTTP используется для создания нового ресурса?	POST	ОПК-1
11.	Какой объект используется для передачи данных между клиентом и сервером без exposing сущности БД?	DTO	ОПК-3
12.	Какой компонент Spring обрабатывает исключения на уровне контроллеров?	@ControllerAdvice	ПК-6
13.	Какой реактивный тип представляет 0 или 1 элемент?	Mono	ОПК-1
14.	Какой реактивный тип представляет поток из 0 или более элементов?	Flux	ОПК-3
15.	Какой фреймворк используется для реализации событийно-ориентированной архитектуры в Spring?	Kafka	ПК-6
16.	Какая аннотация используется для асинхронного выполнения метода?	@Async	ОПК-1
17.	Какой инструмент Spring позволяет мониторить состояние приложения в продакшене?	Actuator	ОПК-3
18.	Какой класс используется для отправки HTTP-запросов из Spring-приложения?	RestTemplate / WebClient	ПК-6
19.	Какой метод вызывается после инициализации бина?	@PostConstruct	ОПК-1

20.	Какой профиль используется для настройки поведения приложения в разных средах (dev, prod)?	профили Spring	ОПК-3
-----	--	----------------	-------