
УТВЕРЖДЕНА

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

**Рабочая программа дисциплины (модуля)
«Язык программирования Java»**

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	9
6. Материально-техническое обеспечение	9
7. Методические и оценочные материалы	11

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Язык программирования Java» 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) «Язык программирования Java» формирует системные знания и практические навыки программирования на языке Java. Дисциплина (модуль) охватывает синтаксис, объектно-ориентированное программирование, работу с коллекциями, исключениями, стримами и современными возможностями языка, включая лямбда-выражения и рефлексия.

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 в 1 семестре.

Цель изучения дисциплины (модуля): сформировать у студентов прочную теоретическую и практическую базу для разработки, отладки и сопровождения программных решений на языке Java, соответствующих современным стандартам промышленной разработки.

Задачи изучения дисциплины (модуля):

- освоить синтаксис языка Java и базовые управляющие конструкции;
- изучить принципы объектно-ориентированного программирования и их реализацию в Java;
- научиться обрабатывать исключения и использовать механизмы try-catch-finally и try-with-resources;
- освоить работу с коллекциями и Stream API для эффективной обработки данных;
- развить навыки написания читаемого, структурированного и поддерживаемого кода с использованием современных возможностей Java.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- синтаксис языка Java, базовые и управляющие конструкции;
- основы ООП: понятия классов и объектов, наследование, полиморфизм и инкапсуляция, интерфейсы и абстрактные классы;
- принципы работы с исключениями в Java;
- основные коллекции и структуры данных в Java, принципы сравнения объектов;
- принципы работы с дженериками, стримами, лямбда-выражениями и функциональными интерфейсами;
- принципы работы аннотаций и рефлексии в Java.

уметь:

- устанавливать и настраивать окружение Java для разработки;
- находить ошибки в коде и исправлять их;
- работать с исключениями, обрабатывать их с использованием try-catch;
- работать с данными при помощи коллекций и Stream API;
- создавать приложения, соответствующие принципам ООП.

владеть:

- навыком уверенно пользоваться IDE;
- навыком читать чужой код на Java и понимать, что он делает;

— навыком писать читаемый и поддерживаемый код на Java.

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	Базовые конструкции языка	4	4		18	Домашние задания
2	Проектирование типов и объектно-ориентированная модель	4	4	1	18	Домашние задания Контрольная работа
3	Обработка ошибок и внешнее взаимодействие	5	4		18	Домашние задания
4	Типобезопасность и структуры данных	5	4		18	Домашние задания
5	Функциональный стиль программирования в Java	4	4	1	18	Домашние задания Контрольная работа
6	Метаданные и анализ программы во время выполнения	4	4		18	Домашние задания
7	Виртуальная машина Java	4	4		18	Домашние задания
	Экзамен			4		
Итого:		30	28	6	126	
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	Базовые конструкции языка	Основы языка Java Работа с объектами
2	Проектирование типов и объектно-ориентированная модель	ООП Наследование
3	Обработка ошибок и внешнее взаимодействие	Исключения Ввод-вывод и работа с файлами Сокеты и работа с сетью
4	Типобезопасность и структуры данных	Дженерики Дженерики и коллекции Коллекции
5	Функциональный стиль программирования в Java	Лямбда-выражения и функциональные интерфейсы Optional и Stream API
6	Метаданные и анализ программы во время выполнения	Аннотации Reflection

7	Виртуальная машина Java	Внутреннее устройство JVM
---	-------------------------	---------------------------

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) «Язык программирования Java» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, домашние задания, контрольные работы, 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.

Контрольная работа – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы – получить специальные знания по одной или нескольким темам дисциплины и продемонстрировать навыки их практического применения.

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) «Язык программирования Java»

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	

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
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) «Язык программирования Java» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашние задания	40%	7	Набор задач по темам недели
Контрольные работы	30%	2	Письменная работа с набором задач, которые нужно решить за ограниченное время
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) «Язык программирования Java»: $0,4 \times \text{среднее за домашние задания} + 0,3 \times \text{среднее за контрольные работы} + 0,3 \times \text{экзамен}$.

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание 1. Базовые конструкции и ООП

1. Напишите класс Student с полями: name, age, grade. Реализуйте конструктор, геттеры и метод toString().
2. Создайте объект Student, выведите его в консоль.
3. Напишите метод, который принимает массив студентов и возвращает самого старшего.
4. Реализуйте интерфейс Comparable для сортировки студентов по оценке.
5. Оформите проект в IDE, добавьте комментарии.

Домашнее задание 2. Исключения, файлы, коллекции

1. Напишите программу, которая читает текстовый файл input.txt и подсчитывает количество строк.
2. Обработайте возможные исключения: FileNotFoundException, IOException.
3. Сохраните результат в output.txt с использованием try-with-resources.
4. Загрузите список строк в ArrayList, отфильтруйте пустые, отсортируйте и выведите.
5. Добавьте логирование (по желанию — через java.util.logging).

Домашнее задание 3. Stream API, аннотации, работа с сетью

1. Используя Stream API:
 - Преобразуйте список строк в верхний регистр
 - Отфильтруйте строки короче 5 символов
 - Подсчитайте количество
2. Напишите кастомную аннотацию @Author, примените её к классу, выведите информацию через рефлексию.
3. Напишите простой TCP-клиент, который подключается к httpbin.org:80 и отправляет GET / HTTP/1.1.
4. Выведите ответ сервера.

5. Упакуйте всё в JAR-файл и запустите.

Примерные задания для контрольных работ

Контрольная работа 1:

1. Напишите код, который создаёт объект класса ArrayList для хранения строк и добавляет в него три элемента.
2. Напишите метод, который принимает int a и int b и выбрасывает IllegalArgumentException, если b == 0.
3. Напишите лямбда-выражение, реализующее интерфейс Predicate<String>, проверяющий, что строка начинается с "Java".
4. Напишите команду, которая отфильтровывает чётные числа из List<Integer> и возвращает их сумму через Stream API.
5. Напишите код, который читает содержимое файла data.txt построчно с использованием try-with-resources.

Контрольная работа 2:

1. Напишите аннотацию @Deprecated, объясните, для чего она используется.
2. Напишите код, который получает все публичные методы класса String с помощью рефлексии.
3. Напишите тип, который используется в Stream API для защиты от NullPointerException.
4. Напишите имя интерфейса, который имеет один абстрактный метод и может быть реализован лямбдой.
5. Напишите имя компонента JVM, который отвечает за сборку мусора.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Какой ключевой элемент Java определяет поведение объекта через методы и состояние через поля?	класс	ОПК-1
2.	Как называется механизм, позволяющий одному классу наследовать поля и методы другого?	наследование	ОПК-3
3.	Какой принцип ООП предполагает сокрытие внутренней реализации объекта?	инкапсуляция	ПК-6
4.	Какой блок кода используется для обработки исключительной ситуации?	try-catch	ОПК-1
5.	Какой интерфейс в Java используется для чтения и записи файлов?	InputStream / OutputStream	ОПК-3
6.	Какой класс позволяет установить сетевое соединение по протоколу TCP?	Socket	ПК-6
7.	Какой механизм в Java позволяет создавать типы, параметризованные другими типами?	дженерики	ОПК-1
8.	Какая коллекция в Java хранит элементы в порядке вставки и допускает дубликаты?	ArrayList	ОПК-3
9.	Какой интерфейс представляет последовательность элементов для функциональных операций?	Stream	ПК-6

10.	Какой функциональный интерфейс принимает один аргумент и возвращает результат?	Function	ОПК-1
11.	Какой контейнер используется для безопасной обработки значений, которые могут отсутствовать?	Optional	ОПК-3
12.	Какой оператор в Stream API фильтрует элементы по заданному условию?	filter	ПК-6
13.	Какой интерфейс в Java представляет действие без параметров и возвращаемого значения?	Runnable	ОПК-1
14.	Как называется механизм, позволяющий добавлять метаданные к элементам кода?	аннотации	ОПК-3
15.	Какой пакет в Java используется для работы с рефлексией?	java.lang.reflect	ПК-6
16.	Какой компонент JVM отвечает за выполнение байт-кода?	интерпретатор / JIT-компилятор	ОПК-1
17.	Какая область памяти в JVM хранит объекты?	куча (heap)	ОПК-3
18.	Какой сборщик мусора освобождает память, на которую нет ссылок?	GC (Garbage Collector)	ПК-6
19.	Какой метод вызывается перед удалением объекта сборщиком мусора?	finalize	ОПК-1
20.	Какой интерфейс в Stream API используется для преобразования элементов?	map	ОПК-3