

**Федеральное государственное образовательное бюджетное учреждение
высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финуниверситет)**

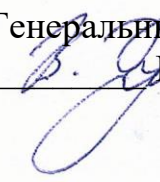
Тульский филиал Финуниверситета

Кафедра «Математика и информатика»

СОГЛАСОВАНО

ЗАО «ЛИМ»

Генеральный директор

 В.Б. Куликов

«26» мая 2026 г.



УТВЕРЖДАЮ

Директор филиала

 Г.В. Кузнецов

«26» мая 2026 г.



Е.В. Манохин

ПРОГРАММИРОВАНИЕ НА JAVA SCRIPT

Рабочая программа дисциплины

для студентов, обучающихся по направлению подготовки
38.03.05 – Бизнес-информатика, профиль - «ИТ-менеджмент в бизнесе»

Рекомендовано Ученым советом Тульского филиала
(протокол № 40 от 26 мая 2026 г.)

Одобрено кафедрой «Математика и информатика»
(протокол № 11 от 05 мая 2026 г.)

Тула 2026

Содержание

№ п/п	Наименование разделов РПД	стр.
1.	Наименование дисциплины	3
2.	Перечень планируемых результатов освоения образовательной программы с указанием индикаторов их достижения, соотнесенных с планируемыми результатами обучения по дисциплине.....	3
3.	Место дисциплины в структуре образовательной программы.....	4
4.	Объем дисциплины в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	4
5	Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий	4
5.1.	Содержание дисциплины	4
5.2.	Учебно-тематический план	7
5.3.	Содержание семинаров, практических занятий.....	9
6.	Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.....	11
6.1.	Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы	11
6.2.	Перечень вопросов, заданий, тем для подготовки к текущему контролю	13
7.	Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине.....	14
7.1.	Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний.....	15
8.	Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.....	17
9.	Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	17
10.	Методические указания для обучающихся по освоению дисциплины	18
11.	Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем.....	19
11.1.	Комплект лицензионного программного обеспечения:	19
11.2.	Современные профессиональные базы данных и информационные справочные системы	19
11.3.	Сертифицированные программные и аппаратные средства защиты информации	19
12.	Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	19

1. Наименование дисциплины

Факультатив «Программирование на Java Script»

2. Перечень планируемых результатов освоения образовательной программы с указанием индикаторов их достижения, соотнесенных с планируемыми результатами обучения по дисциплине

Цель дисциплины - дать современные теоретические знания и сформировать устойчивые практические навыки эффективного применения современных компьютерных технологий в науке, а также проведения собственных научных исследований в финансово-экономической сфере.

Основные задачи дисциплины «Программирование на Java Script»:

- приобретение представлений о современных способах формализации, хранения, поиска и обработки аналитических данных;
- приобретение практических навыков использования информационных систем и их компонентов для отбора, структуризации, анализа экономической информации, построения прогнозных моделей и представления итоговых результатов;
- приобретение навыков интеллектуального анализа информации.

Дисциплина «Программирование на Java Script» обеспечивает формирование компетенций, представленных в таблице 1

Таблица 1

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
УК-4	Способность использовать прикладное программное обеспечение при решении профессиональных задач	1.Использует основные методы и средства получения, представления, хранения и обработки данных.	Знать: основные языковые конструкции Java. Уметь: применять основные языковые конструкции Java.
		2.Демонстрирует владение профессиональными пакетами прикладных программ.	Знать: объектную модель Java: пакеты, классы и объекты. Уметь: применять соответствующие пакеты и функционал языка Java.
		3.Выбирает необходимое прикладное программное обеспечение в зависимости от решаемой задачи.	Знать: методы реализации интерфейсов языка Java для решения типовых задач в различных прикладных областях. Уметь: применять соответствующие методы реализации интерфейсов языка Java для решения типовых задач в различных прикладных областях.

		4.Использует прикладное программное обеспечение для решения конкретных прикладных задач.	Знать: чтение и запись файлов с использованием байтовых потоков. Уметь: применять потоковую организацию системы ввода-вывода Java.
--	--	--	---

3. Место дисциплины в структуре образовательной программы

Дисциплина «Программирование на Java Script» относится к факультативным дисциплинам для направления подготовки 38.03.05 Бизнес-информатика, профиль «ИТ-менеджмент в бизнесе».

4. Объем дисциплины в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

Таблица 2

Вид учебной работы по дисциплине	Всего (в з/е и часах)	Семестр (модуль) 3 (в часах)
Общая трудоёмкость дисциплины	(2 з/е) 72	72
Аудиторные занятия	50	50
Лекции	16	16
Семинары, практические занятия	34	34
Самостоятельная работа	22	22
Вид текущего контроля	-	-
Вид промежуточной аттестации	зачет	зачет

5 Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Обзор платформы и языка программирования Java.

Общая информация о языке программирования Java. История языка программирования, его связь с другими языками программирования, распространенность Java и основные сферы его применения. Знакомство с первыми примерами кода на Java.

Базовая информация о языке программирования Java. Базовая информация о синтаксисе. Базовая информация о типизации. Информация о примитивных типах данных. Математические операции над числовыми типами данных.

Преобразование типов данных. Переменные и специфика их объявления. Именование переменных.

Основная информация о Java Development Kit. Основная информация о Java Runtime Environment. Информация о JIT-компиляции. Виртуальная машина Java. Сборщик мусора. Интегрированная среда разработки Java.

Тема 2. Основные языковые конструкции Java.

Строковый тип данных языка Java. StringBuilder, StringBuffer. Идентификаторы, ключевые слова, литералы, операторы, разделители.

Управляющие конструкции в Java. Булев тип: объявление и операции. Операции сравнения в Java. Условные операторы в Java. Оператор switch-case в Java.

Массивы, особенности инициализации.

Циклы в Java: while, do-while, for, for each. Специфика циклов в Java.

Тема 3. Базовые принципы объектно-ориентированного программирования.

Базовые принципы объектно-ориентированного программирования: наследование, инкапсуляция, полиморфизм. Объектная модель Java: пакеты, классы и объекты. Спецификации классов (модификаторы доступа). Отношения между классами. Конструкторы, методы, переменные и поля классов. Объекты и объектные переменные класса. Ключевое слово final. Статические поля и методы классов.

Тема 4. Классы, методы и объекты Java

Пакеты в Java. Импорт пакетов и классов. Вложенные и внутренние классы. Полиморфизм. Перегрузка методов и конструкторов. Ключевое слово this. Классы java.util. Суперклассы и подклассы. Ключевое слово super. Наследование и доступ к членам класса. Иерархии наследования. Восходящее и нисходящее преобразование типов. Ключевое слово instanceof. Переопределение методов. Абстрактные классы и методы. Типизированные классы и методы (Generic Class, Generic Method). Наследование Generics. Ограничения, налагаемые на параметры. Совместимость на уровне байт-кода.

Тема 5. Интерфейсы. Исключения. Отладка программ.

Реализация интерфейсов. Свойства интерфейсов. Переменные интерфейсов. Множественная реализация интерфейсов. Наследование интерфейсов. Методы интерфейсов: по умолчанию, статические. Обработка исключений. Классификация исключений. Классы Throwable, Exception, RuntimeException, IOException.

Тема 6. Коллекции.

Описание Java Collection. Классификация коллекций java: Collection, Map. Основные методы интерфейса Collection. Interface Iterator<E>, ListIterator<E>. Списки (Interface List<E>), имплементации List: ArrayList, LinkedList.

Интерфейс Set. Реализации интерфейса Set: HashSet, TreeSet. Очереди java, интерфейс Queue, основные методы. Интерфейс Deque.

Тема 7. Потокковая организация системы ввода-вывода. Многопоточность.

Чтение и запись файлов с использованием байтовых потоков. Закрывание файлов. Файлы с произвольным доступом. Файловый ввод-вывод с использованием символьных потоков. Классы FileWriter и FileReader. Потокковая организация системы ввода-вывода Java. Абстрактные классы InputStream и OutputStream. Многопоточное программирование, модель потока в Java. Многопоточность в одно- и много процессорных системах. Класс Thread и интерфейс Runnable: основные методы, создание, приостановка, прерывание, завершение потока. Группы потоков. Жизненный цикл потока. Фоновые задачи. Управление потоками и приоритеты потоков. Синхронизация.

Тема 8. Работа с базами данных в Java. Основы объектно-реляционного отображения.

Архитектура API JDBC (Java DataBase Connectivity), основные компоненты, модели обработки. Преимущества и недостатки применения JDBC. Драйверы JDBC. Установление соединения, классы DriverManager и DataSource, интерфейс Connection. Типы данных SQL и JDBC. Классы Statement, PreparedStatement. Запросы к базе данных. Добавление, удаление, изменение информации в базе данных. Обработка результатов выполнения запроса к базе данных, интерфейс ResultSet. Закрывание соединения.

Основы Java Persistence API: Entity, встраиваемые классы, управление связями, стратегии наследования маппинга. Основы Hibernate Framework. Основы объектно-реляционного отображения. Основы языка запросов HQL. Отличия Entity в Hibernate и Java Persistence API.

Тема 9. Системы автоматической сборки пакетов. Реализация графического интерфейса в языке программирования Java.

Автоматизация сборки. Компиляция, управление зависимостями, автоматическое тестирование, пакетирование приложений. Основные особенности Maven. Структура языка разметки при помощи XML. Основные особенности Gradle. Структура языка разметки при помощи Groovy. Сравнение систем автоматической сборки.

Библиотека JavaFX. Классы Stage, Scene. Узлы и граф сцены. Класс Application и жизненный цикл приложения. Компоненты JavaFX: CheckBox, ListView, TextField. JavaFX SceneBuilder.

Тема 10. Spring Framework. Архитектура, основные сведения.

Основные фреймворки и задачи, решаемые Spring. Внедрение зависимостей. Интерфейс ApplicationContext. Inversion of Control – инверсия управления. Dependency Injection – внедрение зависимости между объектами приложения. Внедрение зависимости при помощи конструктора класса и методов set.

Конфигурирование метаданных: xml конфигурация, Java-аннотация, Java-код. Объект Bean. Область видимости объектов Bean: singleton, prototype, request, session, application. Жизненный цикл объекта Bean. Спецификация, применение init- и destroy-методов. Аннотации @Component, @Autowired. Основы интерфейса Servlet.

Тема 11. RESTful API приложения.

Архитектура REST (Representational State Transfer — «передача состояния представления») и RESTful API. Spring MVC (Model-View-Controller), общая схема работы. Класс DispatcherServlet. Аннотации @Controller, @Component, @ComponentScan. Маппинг (определение соответствия): аннотации @GetMapping, @PostMapping, @RequestMapping. Обработка HTTP-запросов: интерфейсы HttpServletRequest, Model.

Классификация функций по манипуляции данными CRUD (create, read, update, delete). Взаимосвязь CRUD и REST. Связь с базой данных на основе паттерна DAO (Data Access Object). Применение класса JdbcTemplate.

Тема 12. Генерация веб-страниц. Основы Thymeleaf

Интеграция Thymeleaf с Spring. SpringStandart диалект. Views и View Resolvers. Бизнес-слой. Контроллер. Конфигурирование Conversion Service. Создание форм: обработка командного объекта, поля ввода, динамические поля. Проверка и сообщения об ошибках. Служба конверсии. Отрисовка фрагментов шаблона.

Тема 13. Тестирование. Документирование в Java проектах.

JUnit, библиотека модульного тестирования. JUnit аннотации: @Test, @Before, @BeforeClass, @AfterClass, @Ignore. JUnit утверждения, класс Assert. Правила, @Rule. Генератор документирования Javadoc. Виды комментариев, форматирование. Javadoc на уровне класса, метода и поля. Использование тегов. Генерируемые исключения. Применение гиперссылок.

5.2. Учебно-тематический план

Таблица 3

№ п/п	Наименование темы (раздела) дисциплины	Трудоемкость в часах					Формы текущего контроля успеваемости
		Всего	Аудиторная работа			Самостоя тельная работа	
			Общая	Лекции	Семинары, практические занятия		
1	Обзор платформы и языка программирования Java	4	3	1	2	1	Опрос, проверка практических заданий
2	Основные языковые конструкции Java	5	3	1	2	2	Опрос, проверка практических заданий
3	Базовые принципы	8	6	2	4	2	Опрос,

№ п/п	Наименование темы (раздела) дисциплины	Трудоемкость в часах					Формы текущего контроля успеваемости
		Всего	Аудиторная работа			Самостоя тельная работа	
			Общая	Лекции	Семинары, практические занятия		
	объектно-ориентированного программирования						проверка практических заданий
4	. Классы, методы и объекты Java	8	6	2	4	2	Опрос, проверка практических заданий
5	Интерфейсы. Исключения. Отладка программ.	5	3	1	2	2	Опрос, проверка практических заданий
6	Коллекции.	5	3	1	2	2	Опрос, проверка практических заданий
7	Потоковая организация системы ввода-вывода. Многопоточность.	5	3	1	2	2	Опрос, проверка практических заданий
8	Работа с базами данных в Java. Основы объектно-реляционного отображения.	8	6	2	4	2	Опрос, проверка практических заданий
9	Системы автоматической сборки пакетов. Реализация графического интерфейса в языке программирования Java.	5	3	1	2	2	Опрос, проверка практических заданий
10	Spring Framework. Архитектура, основные сведения.	5	3	1	2	2	Опрос, проверка практических заданий
11	RESTful API приложения	4	3	1	2	1	Опрос, проверка практических заданий
12	Генерация веб-страниц. Основы Thymeleaf	4	3	1	2	1	Опрос, проверка практических заданий
13	Тестирование. Документирование в Java проектах	6	5	1	4	1	Опрос, проверка практических заданий
В целом по дисциплине		72	50	16	34	22	Согласно учебному плану: не

№ п/п	Наименование темы (раздела) дисциплины	Трудоемкость в часах					Формы текущего контроля успеваемости
		Всего	Аудиторная работа			Самостоя тельная работа	
			Общая	Лекции	Семинары, практические занятия		
							предусмотрен
Итого в %		100	69	32	68	31	

5.3. Содержание семинаров, практических занятий

Таблица 4

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9 (указывается раздел и порядковый номер источника)	Формы проведения занятий
Обзор платформы и языка программирования Java	Комплект разработчика (Java development kit), среда выполнения Java (Java Runtime Environment). Интегрированная среда разработки (Integrated development Environment) IntelliJ Idea. Установка. Создание нового проекта, системы сборки проектов Maven. Структура проекта. Интерфейс IDE IntelliJ Idea. Создание исполняемого файла. Использование инструментов командной строки. Прimitives и ссылочные типы данных, литералы. Операции. Преобразование типов. Переменные: объявление, инициализация. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.
Основные языковые конструкции Java	Консольный ввод-вывод данных, команды ввода- вывода. Переменные и основные типы переменных. Объявление и инициализация переменных. Присвоение данных между разными типами переменных. Инициализация переменных. Специальные операторы. Операторы «вычисление остатка» и «вычисление частного». Логический тип (булевы значения). Ветвление (конструкция «if – else»). Класс Math. Циклы for и while. Массивы и методы. Строковые и символьные переменные. Классы String, StringBuilder, StringBuffer. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.
Базовые принципы объектно- ориентированного программирования	Объекты и классы. Состояние и поведение. Построение классов в соответствии с принципами инкапсуляции, полиморфизма и наследование. Перегрузка конструкторов, методов. Области видимости, модификаторы доступа. Применение ключевого слова static. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.
Классы, методы и объекты Java	Объявление классов. Метод main. Отношения между классами. Имена классов, импорт. Инициализаторы. Преобразование ссылочных типов. Ключевые слова this и super. Обобщенные методы и конструкторы. Применение ключевого слова final к переменным	Выполнение практических заданий по программированию.

	экземпляра класса. Рекомендуемые источники: 8.1, 8.2	
Интерфейсы. Исключения. Отладка программ	Объявление, реализация и применение интерфейсов. Наследование интерфейсов. Методы интерфейсов: по умолчанию, статические. Ошибки при работе программы. Исключения (Exceptions). Конструкция try-catch-finally. Использование оператора throw. Отладка программ в среде программирования IDE IntelliJ IDEA. Запуск отладчика, точки останова. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.
Коллекции	Интерфейсы Collection, Set, List, Map, Iterator. Каркас коллекций. Представления и оболочки. Унаследованные коллекции. Абстрактные классы, используемые при работе с коллекциями. Интерфейс Comparator. Применение ArrayList. TreeMap. Применение очередей. Алгоритмы: сортировка, двоичный поиск. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.
Потоковая организация системы ввода-вывода. Многопоточность.	Консольный ввод, запись консольного вывода. Классы-реализации потоков данных. Классы ByteArrayInputStream и ByteArrayOutputStream. Классы FileInputStream и FileOutputStream. StringBufferInputStream. Классы FilterInputStream и FilterOutputStream. BufferedInputStream и BufferedOutputStream. DataInputStream и DataOutputStream. Класс Thread и интерфейс Runnable. Жизненный цикл потока. Фоновые задачи. Управление потоками и приоритеты потоков. Синхронизация. Приостановка, возобновление и остановка потока. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.
Работа с базами данных в Java. Основы объектно-реляционного отображения.	Установление соединения с системами управления баз данных MySQL и PostgreSQL. Драйверы JDBC. Установление соединения, классы DriverManager и DataSource, интерфейс Connection. Классы Statement, PreparedStatement. Запросы к базе данных. Добавление, удаление, изменение информации в базе данных. Обработка результатов выполнения запроса к базе данных, интерфейс ResultSet. Закрытие соединения. Драйверы JPA. Hibernate framework. Классы Entity, встраиваемые классы, наследование маппингов, fetch стратегии. Жизненный цикл Entity, EntityManager. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.
Системы автоматической сборки пакетов. Реализация графического интерфейса в языке программирования Java.	Сборка проекта при помощи Maven. Основы синтаксиса Maven. Сборка проекта при помощи Gradle. Синтаксис Groovy. Классы JavaFX Stage, Scene. Узлы и граф сцены. Класс Application и жизненный цикл приложения. Компоненты JavaFX: CheckBox, ListView, TextField. JavaFX SceneBuilder. Обработка событий. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.

Spring Framework. Архитектура, основные сведения	Конфигурация приложения при помощи xml файлов, java-аннотации. Внедрение зависимостей на основе конструкторов и set-методов. Спецификация, применение init- и destroy-методов. Аннотации @Component, @Autowired. Автоматическое связывание свойств компонентов. Рекомендуемые источники: 8.2	Выполнение практических заданий по программированию.
RESTful API приложения.	Использование классов поддержки DAO. Класс DispatcherServlet. Аннотации @Controller, @Component, @ComponentScan. Маппинг (определение соответствия): аннотации @GetMapping, @PostMapping, @RequestMapping. Обработка HTTP-запросов: интерфейсы HttpServletRequest, Model. Рекомендуемые источники: 8.2	Выполнение практических заданий по программированию.
Генерация веб-страниц. Основы Thymeleaf	Интеграция Thymeleaf с Spring. SpringStandart диалект. Views и View Resolvers. Бизнес-слой. Контроллер. Конфигурирование Conversion Service. Создание форм: обработка командного объекта, поля ввода, динамические поля. Проверка и сообщения об ошибках. Служба конверсии. Отрисовка фрагментов шаблона. Реализация базовой веб страницы с использованием Thymeleaf. Рекомендуемые источники: 8.2.	Выполнение практических заданий по программированию.
Тестирование. Документирование в Java проектах	JUnit, библиотека модульного тестирования. JUnit аннотации: @Test, @Before, @BeforeClass, @AfterClass, @Ignore. JUnit утверждения, класс Assert. Правила, @Rule. Генератор документирования Javadoc. Виды комментариев, форматирование. Javadoc на уровне класса, метода и поля. Использование тегов. Генерируемые исключения. Применение гиперссылок. Рекомендуемые источники: 8.1, 8.2	Выполнение практических заданий по программированию.

6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Таблица 5

Наименование темы (раздела) дисциплины или № темы	Указание разделов и тем, отводимых на самостоятельное освоением обучающимися	Формы внеаудиторной самостоятельной работы
Обзор платформы и языка программирования Java	Установка IDE IntelliJ Idea. Переменные среды JDK. Регистрация github, создание проекта для выполнения семинарских занятий.	Изучение литературы и веб-источников.
Основные языковые конструкции Java	Создание блоков кода. Стилизовое оформление программного кода. Ключевые слова. Укороченные логические операции. Многоступенчатые конструкции «if – else – if».	Изучение литературы и веб-источников.

	Оператор Switch. Цикл без тела. Вложенные циклы.	
Базовые принципы объектно-ориентированного программирования	Поиск пакета и переменная среды CLASSPATH.	Изучение литературы и веб-источников.
Классы, методы и объекты Java	Защищенные члены классов. Класс Object. Многоуровневая иерархия классов. Передача объектов методам. Инициализация полей по умолчанию. Вызов одного конструктора из другого.	Изучение литературы и веб-источников.
Интерфейсы. Исключения. Отладка программ	Интерфейсы и абстрактные классы. Разрешение конфликтов в методах по умолчанию. Лямбда-выражения. Генерация исключений в лямбда-выражениях. Переопределение методов и исключения.	Изучение литературы и веб-источников.
Коллекции	Связные списки. Хэш-множества. Очереди по приоритету. Древовидные множества. Взаимные преобразования списков и множеств. Стеки. Битовые множества.	Изучение литературы и веб-источников.
Потоковая организация системы ввода-вывода. Многопоточность.	Классы PipedInputStream и PipedOutputStream. SequenceInputStream. LineNumberInputStream. PushBackInputStream. PrintStream. Синхронизированные блоки. Параллельное программирование.	Изучение литературы и веб-источников.
Работа с базами данных в Java. Основы объектно-реляционного отображения.	Исключения (SQL Exception). Потоки в JDBC. Основные этапы работы с базой данных при использовании JDBC. Уровни изоляции транзакций в JDBC. Хранимая процедура.	Изучение литературы и веб-источников.
Системы автоматической сборки пакетов. Реализация графического интерфейса в языке программирования Java.	Эффекты и преобразования в JavaFX. Класс List. Диспетчеры сеточно-контейнерной, групповой компоновки в Swing. Пиктограммы в меню.	Изучение литературы и веб-источников.
Spring Framework. Архитектура, основные сведения	Аспектно-ориентированный Spring. Выбор точек сопряжения в описаниях срезков. Объявление аспектов в XML. Аннотирование аспектов. Внедрение аспектов AspectJ.	Изучение литературы и веб-источников.
RESTful API приложения.	URI шаблоны с регулярными выражениями. Сборка исполняемого JAR-файла. Шаблоны с Placeholder. Spring Boot.	Изучение литературы и веб-источников.
Генерация веб-страниц. Основы Thymeleaf	Конфигурирование веб-страницы с использованием Thymeleaf. Реализация форм ввода.	Изучение литературы и веб-источников.
Тестирование. Документирование в Java проектах	Дескрипторы javadoc. Общая форма документирующих комментариев.	Изучение литературы и веб-источников.

6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

Пример тестовых заданий

Что такое ООП?

1. Объектно-ориентированное программирование — методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования.

2. Объектно-ориентированное программирование — так называют любой тип программирования, в котором используются понятия высокого уровня и, в отличие от *Assembler*, в котором не работают напрямую с ячейками памяти ПК.

3. Объектно-ориентированное программирование — просто красивое понятие. Если вдуматься, оно не несет дополнительной смысловой нагрузки, просто программисты любят аббревиатуры, так области их знаний выглядят сложнее.

4. Очень одинокий программист.

Что такое класс в Java?

1. Уровень сложности программы. Все операторы делятся на классы в зависимости от сложности их использования.

2. Базовый элемент объектно-ориентированного программирования в языке Java.

3. Просто одно из возможных названий переменной.

4. Такое понятие есть только в C++, в Java такого понятия нет.

Как объявить класс в коде?

1. `class MyClass {}`

2. `new class MyClass {}`

3. `select * from class MyClass {}`

4. `MyClass extends class {}`

Примерный перечень вопросов для текущего контроля

1. Новейшие направления в области создания технологий программирования. Законы эволюции программного обеспечения.

2. Программирование в средах современных информационных систем: создание модульных программ, элементы теории модульного программирования объектно-ориентированные проектирование и программирование

3. Среда *Net. Framework*. Общезыковая исполняющая среда (CLR). Стандарты *JAVA SCRIPT*. Технологии отладки

4. Объектно-ориентированный подход к проектированию и разработке про-грамм. Сущность объектно-ориентированного подхода
5. Инкапсуляция. Понятие класса. Управление доступом к элементам класса. Конструкторы и деструкторы. Переменные объектного типа.
6. Друзья класса.
7. Наследование. Базовый класс. Управление доступом при наследовании.
8. Простое наследование. Виртуальные функции.
9. Чистые виртуальные функции. Абстрактные классы. Непрямые базовые классы. Виртуальные деструкторы. Виртуальные базовые классы
10. Полиморфизм.
11. Методы общего полиморфизма: перегрузка операций. Реализация перегруженной операции. Перегрузка операции присваивания, инкремента, декремента, бинарных арифметических операций.
12. Методы общего полиморфизма: перегрузка операций. Перегрузка операций ввода-вывода в потоки, индексации массивов, операций выделения и освобождения памяти.
13. Методы общего полиморфизма: Преобразования типов, определяемые классом.
14. Методы общего полиморфизма: перегрузка функций. Чистый полиморфизм.
15. Параметрический полиморфизм. Шаблоны функций.
16. Параметрический полиморфизм. Шаблоны классов

Критерии балльной оценки различных форм текущего контроля успеваемости

Организация текущего контроля успеваемости обучающихся по итогам семестра (модуля), по программам бакалавриата и магистратуры в соответствии с учебными планами по направлениям подготовки высшего образования утверждена приказом ректора Финуниверситета по основной деятельности от 23.03.2017 №0557/о «Об утверждении Положения о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся по программам бакалавриата и магистратуры в Финансовом университете».

7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине содержится в разделе «2. Перечень планируемых результатов освоения образовательной программы

(перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».

7.1. Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний

Таблица 7

Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые задания
УК-4 Способность использовать прикладное программное обеспечение при решении профессиональных задач	1. Использует основные методы и средства получения, представления, хранения и обработки данных.	Знать: основные языковые конструкции Java. Уметь: применять основные языковые конструкции Java.	Задание 1. Разработать шаблон генерации технической документации.
	2. Демонстрирует владение профессиональным и пакетами прикладных программ.	Знать: объектную модель Java: пакеты, классы и объекты. Уметь: применять соответствующие пакеты и функционал языка Java.	Задание 1. Применить шаблонизатор для генерации документации.
	3. Выбирает необходимо прикладное программное обеспечение в зависимости от решаемой задачи.	Знать: методы реализации интерфейсов языка Java для решения типовых задач в различных прикладных областях. Уметь: применять соответствующие методы реализации интерфейсов языка Java для решения типовых задач в различных прикладных областях.	Задание 1. Разработать программу, которая производит проверку кода на соответствие спецификации java.
	4. Использует	Знать: чтение и	Задание 1. Разработать

	прикладное программное обеспечение для решения конкретных прикладных задач.	запись файлов с использованием байтовых потоков. Уметь: применять потоковую организацию системы ввода-вывода Java.	клиент-серверное приложение справочной системы с документированием в javadoc.
--	---	--	---

Примерный перечень вопросов для подготовки к зачету

1. Определение класса. Данные и методы класса.
2. Конструктор и деструктор класса. Определение. Виды конструкторов.
3. Указатели на компонентные функции.
4. Объекты класса.
5. Массив объектов класса.
6. Статические члены класса.
7. Наследование. Простое и множественное наследование.
8. Конструкторы производных классов. Особенности вызова.
9. Виртуальные функции. Назначение.
10. Абстрактные классы. Чистые виртуальные функции.
10. Абстрактные классы. Чистые виртуальные функции.
12. Перегрузка операций. Назначение.
12. Какие операции не перегружаются.
13. Перегрузка унарных операций в Java Script функцией-членом класса.
14. Перегрузка унарных операций в Java Script дружественной функцией.
15. Перегрузка бинарных операций в Java Script функцией-членом класса.
16. Перегрузка бинарных операций в Java Script дружественной функцией.
17. Какие виды преобразований можно реализовать в Java Script.
18. Преобразование из типа класса к скалярному типу.
19. Преобразование из скалярного типа к типу класса.
20. Преобразование из типа одного класса к типу другого класса.
21. Конструкторы преобразований. Назначение
22. Операции преобразований. Назначений.
23. Преимущества перед процедурным программированием.
24. Принципы разработки иерархии классов и ее использования.
25. Разрешение вопросов, связанных с неоднозначностью доступа.
26. Иерархия классов.

27. Объектно-ориентированное программирование как основа современного программирования.
28. Совместное использование функций и перегрузка операторов.
29. Использование динамической памяти.
30. Механизм исключительных ситуаций.
31. Поддержка модульности. Разбиение программы на единицы компиляции.
32. Шаблонные классы и стандартные контейнеры.
33. Организация пользовательского интерфейса.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Рекомендуемая литература

Основная

1. Гуськова, О. И. Объектно ориентированное программирование в Java : учеб. пособие / О. И. Гуськова. Москва : МПГУ, 2018. 240 с. ЭБС Znanium.com. URL: <https://znanium.com/catalog/product/1020593> (дата обращения: 12.01.2025). Текст : электронный.
2. Пруцков, А. В. Язык программирования Java. Введение в курс: операторы и типы данных : учеб. пособие / А. В. Пруцков. Рязань : РГРТУ, 2016. 72 с. ЭБС Лань. URL: <https://e.lanbook.com/book/168307> (дата обращения: 12.01.2025). Текст : электронный.

Дополнительная

5. Блох Джошуа Java. Эффективное программирование. 3-е изд. М.: Диалектика-Вильямс, 2018. 464 с.
6. Уоллис Крейг Spring в действии. 6-е изд. М.: ДМК Пресс, 2022. 544 с.
7. Шилдт Герберт Java. Полное руководство. 10-е изд. М.: Диалектика-Вильямс, 2020. 1488 с.
8. Лонг Джош, Бастани Кеннет Java в облаке. Spring Boot, Spring Cloud, Cloud Foundry. 2-е изд. СПб.: Питер, 2019. 624 с.

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Java документация. <https://docs.oracle.com/en/java/>
2. The Java™ Tutorials <https://docs.oracle.com/javase/tutorial/java/concepts/index.html>
3. Электронная библиотека Финансового университета (ЭБ) <http://elibrary.ru/>
4. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
5. Электронно-библиотечная система Znanium <http://www.znanium.com>

6. Электронно-библиотечная система Университетская библиотека online
<https://biblioclub.ru>

10. Методические указания для обучающихся по освоению дисциплины

Лекционные занятия проводятся в соответствии с тематическим планом, при изложении материала активно используются презентации в среде PowerPoint.

В ходе интерактивных занятий проводится разбор конкретных ситуаций. При изучении теоретического материала необходимо опираться на рабочую программу дисциплины и литературу из основного списка.

Кроме этого материала необходимо активно работать с Интернет-источниками и пособиями других авторов, помогающими усвоить материал отдельных разделов программы.

Семинарские занятия проводятся по следующей схеме:

Студенты в индивидуальном режиме выполняют задания в компьютерном классе. Существуют две основных формы заданий. В первом случае всем студентам предлагаются одинаковые исходные данные. После выполнения заданий устраивается обсуждение результатов в форме тематической дискуссии. Во втором случае каждому из студентов предлагается индивидуальное задание. В этом случае после выполнения задания организуется работа в группах: каждый из студентов анализирует и критически оценивает или же интерпретирует результаты одного или нескольких членов группы. Домашнее задание, или контрольная работа выполняется студентами индивидуально и самостоятельно во внеаудиторное время. Каждый из студентов выбирает область экономических знаний, для которой можно найти данные для анализа в свободных источниках, выдвигает предположения о наличии скрытых закономерностей в отобранных данных, определяет вид этих закономерностей, выбирает соответствующие инструментальные средства интеллектуального анализа данных, осуществляет поиск закономерностей и делает выводы по возможности применения методов интеллектуального анализа данных в избранной области. Результаты творческого задания оформляются. Критериями его оценки, помимо адекватности выбора и применения инструментов интеллектуального анализа данных и корректности выводов, служат творческий подход, новизна содержания и осмысления, оригинальность выводов.

Внедрение активных и интерактивных элементов в проведение занятий по дисциплине может осуществляться разными методами: проблемный семинар с групповым обсуждением, деловая игра, опрос, case-study и др.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

11.1. Комплект лицензионного программного обеспечения:

лицензионное отечественное:

1. Антивирусная защита Kaspersky Endpoint Security.
2. Astra Linux.
3. LibreOffice.

лицензионное импортное:

ПО для ТСО лиц с ограниченными возможностями здоровья
свободно распространяемое:

Архиватор KDE (Ark).

11.2. Современные профессиональные базы данных и информационные справочные системы

1. Информационно-правовая система «Гарант»
2. Информационно-правовая система «Консультант Плюс»
3. Информационно-образовательный портал Финансового университета:

<http://org.fa.ru>.

11.3. Сертифицированные программные и аппаратные средства защиты информации

Не предусмотрены.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Образовательный процесс по учебной дисциплине осуществляется в учебных аудиториях для проведения учебных занятий. При освоении дисциплины используются технические средства мультимедийной техники аудиторий.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения в сети «Интернет» и обеспечением доступа в электронно-образовательную среду Финансового университета.