

Балаковский инженерно-технологический институт - филиал
федерального государственного автономного образовательного
учреждения высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

Факультет атомной энергетики и технологий
Кафедра «Информационные системы и технологии»

РАБОЧАЯ ПРОГРАММА

по дисциплине «Современные технологии интернет-программирования»

Направления подготовки

«09.03.02 Информационные системы и технологии»

Основная профессиональная образовательная программа

«Информационные системы и технологии»

Квалификация выпускника

Бакалавр

Форма обучения

Очная

Балаково

Цель освоения дисциплины

Целями освоения учебной дисциплины в области обучения, воспитания, развития, соотнесенные с общими целями ООП ВО и требованиями профессиональных стандартов («Руководитель проектов в области информационных технологий» утвержден приказом Министерства труда и социальной защиты Российской Федерации от 18.11.2014 г. № 893н, Специалист по информационным системам утвержден приказом Министерства труда и социальной защиты Российской Федерации от 18.11.2014 N 896н): ознакомление с современными технологиями Интернет-программирования, моделями, методами и средствами решения функциональных задач и организации информационных процессов; понимание назначения веб-сайта, круга функций, которые он должен выполнять, понимание истинных задач, которые собираются решать с помощью сайта его владельца; рассмотрение перспектив использования Интернет-технологий в условиях перехода к информационному обществу; изучение особенностей технологии JSP, а также создания содержимого, которое имеет как статические, так и динамические компоненты.

Место учебной дисциплины в структуре ООП ВО

Необходимыми условиями для освоения дисциплины являются знания, умения и практические навыки по предшествующим дисциплинам и практикам в соответствии с требованиями освоения компетенций:

Технологии программирования

Инструментальные средства информационных систем

Современные технологии управления базами данных

Языки программирования

Современные среды визуального программирования

Объектно-ориентированное программирование

Интерактивное программирование web-приложений / Создание интерактивных приложений для интернет

Психология / Инженерная психология

Управление данными

Знания, умения и практические навыки, полученные при освоении дисциплины, необходимы при изучении следующих дисциплин и прохождения практик:

Архитектура информационных систем

Информационный менеджмент

Методы и средства проектирования информационных систем и технологий

Электронный документооборот / Системы электронного документооборота на предприятии

Производственная практика (преддипломная), а также государственная итоговая аттестация

При освоении данной дисциплины студент сможет частично продемонстрировать следующие трудовые функции: Сбор информации для

инициации проекта в соответствии с полученным заданием; организационное и технологическое обеспечение определения первоначальных требований заказчика к ИС и возможности их реализации в ИС

Компетенции обучающегося, формируемые в результате освоения дисциплины

В процессе освоения данной дисциплины у студента формируются следующие компетенции:
общепрофессиональные

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции
ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	<p>З-ОПК-6 Знать: методы алгоритмизации, языки и технологии программирования; библиотеки программных модулей; шаблоны, классы объектов, используемые при разработке программного обеспечения</p> <p>У-ОПК-6 Уметь: создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов; использовать выбранную среду программирования для написания программного кода</p> <p>В-ОПК-6 Владеть: языками и средами программирования для разработки алгоритмов и программ</p>

Профессиональные

Задачи профессиональной деятельности (ЗПД)	Объект или область знания	Код и наименование компетенции	Индикаторы достижения компетенции
выбор исходных данных для проектирования	информационные процессы, технологии, системы и сети, их инструментальное (программное, техническое, организационное) обеспечение, способы и методы проектирования, отладки, производства и эксплуатации информационных технологий и систем в различных областях и сферах	ПК-12 Способен проводить выбор исходных данных для проектирования с учетом требований заказчика	<p>-ПК-12 Знать: требования к разработке технического задания, его структуру и принципы составления</p> <p>У-ПК-12 Уметь: анализировать исходную документацию заказчика</p> <p>В-ПК-12 Владеть: методикой составления технического задания</p>
		ПК-13 Способен на основе взаимодействия с заказчиком осуществлять контроль содержания и	<p>З-ПК-13 Знать: принципы формирования пакета исходной документации для проектирования информационных систем</p> <p>У-ПК-13 Уметь:</p>

	деятельности.	качества исходной информации для проектирования ИС	осуществлять контроль содержания и качества исходной информации для проектирования информационных систем В-ПК-13 Владеть: инструментами формирования пакета исходной документации
--	---------------	--	---

Задачи воспитания, реализуемые в рамках освоения дисциплины

Направление/цели	Создание условий, обеспечивающих	Использование воспитательного потенциала учебной дисциплины	Вовлечение в разноплановую внеучебную деятельность
Профессиональное и трудовое воспитание	формирование культуры исследовательской и инженерной деятельности (В16)	Использование воспитательного потенциала дисциплин профессионального модуля для формирования инженерного мышления и инженерной культуры за счёт практических студенческих исследований современных производственных систем; проектной деятельности студентов по разработке и оптимизации технологических систем, связанной с решением реальных производственных задач; прохождения через разнообразные игровые, творческие, фестивальные формы, требующие анализа сложного объекта, постановки относительно него преобразовательных задач для их оптимального решения.	1. Организация научно-практических конференций и встреч с ведущими специалистами предприятий города и ветеранами атомной отрасли. 2. Организация и проведение предметных олимпиад и участие в конкурсах профессионального мастерства.

Структура и содержание учебной дисциплины

Дисциплина преподаётся студентам в 5, 6-ом семестрах. Общая трудоемкость дисциплины составляет 6 зачетных единицы, 216 ак. часа.

Календарный план

№ Р а з д е л а	№ Т е м ы	Наименование раздела (темы) дисциплины	Виды учебной деятельности (в часах)					Аттеста ция раздела (форма *)	Макси маль- ный балл за раздел **
			Всего	Лекции	Лабораторные	Практические	СРС/КРС		
1	1-2	Обзор технологии сервлетов и их архитектура	18	4		4	10	Т	25
2	3-8	Основные понятия технологии Java Server Pages	54	12/6		12/6	30	Т	25
Вид промежуточной аттестации								3	50
Итого			72/6	16/6		16/6	40		100
3	9-12	Разработка корпоративных приложений на языке Java	48	12/6		24/8	12	Т	25
4	13- 14	Серверы приложений	40/20	4		24/8	12/20	Т	25
Вид промежуточной аттестации			36					Э	50
Итого			144	16/6		48/16	24/20		100
Всего			216	36	-	64	64/КСР20		

* - сокращенное наименование формы контроля

** - сумма максимальных баллов должна быть равна 100 за семестр, включая зачет и (или) экзамен

Сокращенное наименование форм текущего контроля и аттестации разделов:

Обозначение	Полное наименование
Т	Тестирование
З	Зачет
Э	Экзамен

Содержание лекционного курса

Тема лекции. Вопросы, отрабатываемые на лекции	Всего часов	Учебно- методическое обеспечение
Обзор технологии сервлетов и их архитектура. Интерфейс Servlet и жизненный цикл сервлета. Класс HttpServlet. Интерфейсы	2	1-3

HttpServletRequest и HttpServletResponse. Обработка get и post HTTP-запросов. Переадресация запросов.		
Отслеживание состояния сеанса. Cookies. Отслеживание сеанса с помощью интерфейса HttpSession. Многоуровневые приложения с использованием средств JDBC. Класс HttpUtils.	2	1-3
Обзор технологии JavaServer Pages. Неявные объекты. Сценарии. Стандартные действия: <jsp:include>, <jsp:forward>, <jsp:plugin>, <jsp:useBean>. Директивы page и include.	2	1-3
Библиотеки нестандартных тегов JSP. Простой нестандартный тег. Нестандартный тег с атрибутами. Обработка тела нестандартного тега.	2	1-3
Разработка приложений для беспроводной связи на базе Java и J2ME. Обзор сервлета WelcomeServlet. Обзор сервлета TipTestServlet. Java 2 Micro Edition.	2	1-3
Сеансовые компоненты EJB. Обзор технологии EJB: удаленный интерфейс, собственный интерфейс, реализация EJB, контейнер EJB. Сеансовые компоненты: сеансовые компоненты EJB с состоянием, развертывание сеансовых компонентов EJB, сеансовые компоненты EJB без состояния. EJB-транзакции.	2	1-3
Распределенные EJB-транзакции. Собственный и удаленный интерфейс EJB MoneyTransfer. Разграничение транзакций с управлением на стороне компонента. Разграничение транзакций с управлением на стороне контейнера. Клиентский EJB-компонент MoneyTransfer. Развертывание EJB-компонента MoneyTransfer.	2	1-3
Компоненты EJB с данными. Обзор EJB-компонентов с данными. Компонент-сущность EJB Employee, хранящий информацию о сотруднике. Собственный и удаленный интерфейсы EJB-компонента Employee. EJB-компонент Employee с персистентностью, управляемый контейнером. Клиент EJB-компонента Employee.	2	1-3
Обмен сообщениями с помощью Java Message Service. Обмен сообщениями «от точки к точке». Обмен сообщениями в модели «издатель/подписчик». Компоненты Enterprise JavaBeans, управляемые сообщениями.	2	1-3
Обзор архитектуры корпоративного приложения. Общая архитектуры системы. Компоненты Enterprise JavaBeans: EJB-сущности, сеансовые EJB-компоненты с состоянием. Логика управления, реализуемая сервлетами. Логика внешнего представления данных посредством XSLT.	2	1-3
Логика представления данных и логика управления корпоративным приложением. Базовый класс XMLServlet. Сервлеты, реализующие магазинную тележку. Сервлеты, обслуживающие каталог товаров. Сервлеты для обслуживания покупателей.	2	1-3
Бизнес-логика корпоративного приложения. Архитектура компонентов EJB. Реализация магазинной тележки. Реализация EJB-компонента Product. Реализация EJB-компонента Order. Реализация EJB-компонента OrderProduct. Реализация EJB-компонента Customer. Реализация EJB-компонента Address. Реализация EJB-компонента SequenceFactory. Развертывание приложения средствами J2EE.	2	1-3
Серверы приложений. Спецификация J2EE и ее преимущества. Коммерческие серверы приложений: BEA WebLogic, iPlanet Application Server, IBM WebSphere.	4	1-3
Введение в web-сервисы и SOAP. Простой протокол доступа к	4	1-3

объектам (SOAP). Служба погоды, реализованная посредством SOAP.		
---	--	--

Перечень практических занятий

Тема практического занятия. Задания, вопросы, отрабатываемые на практическом занятии	Всего часов	Учебно-методическое обеспечение
Разработка web-приложения с использованием технологий JSP и Servlet.	4	1-3
Управление сеансом пользователя на JSP.	6	1-3
Управление данными cookies на JSP.	6	1-3
Основы построения пользовательских дескрипторов JSP.	8	1-3
Разработка системы авторизации на JSP.	8	1-3
Разработка системы поиска по сайту на JSP.	8	1-3
Разработка системы создания динамического контента на JSP.	8	1-3
Разработка системы тестирования на JSP.	8	1-3
Разработка RMI-приложений.	8	1-3

Перечень лабораторных работ - не предусмотрены учебным планом

Задания для самостоятельной работы студентов

Вопросы, для самостоятельного изучения (задания)	Всего часов	Учебно-методическое обеспечение
Навигация в сети Internet.	6	1-3
Принцип работы каталогов.	6	1-3
Принцип работы информационно-поисковых систем.	6	1-3
Полнота и релевантность информации.	6	1-3
Метод Google для измерения «важности» страницы PageRank.	6	1-3
Глубинный анализ текста Text Mining.	6	1-3
Поиск в динамической новостной среде.	6	1-3
Архитектура web-приложений.	10	1-3
Сервис-ориентированная архитектура SOA.	12	1-3

Расчетно-графическая работа

Расчетно-графическая работа не предусмотрена учебным планом

Курсовая работа

Курсовая работа выполняется по теме «Разработка информационного портала организации». Целью курсовой работы является систематизация и закрепление теоретических знаний, полученных за время обучения, а также приобретение и закрепление навыков самостоятельной работы.

В курсовой работе студент должен показать степень освоения теоретическим материалом, степень понимания назначения информационных порталов, умение производить обоснованный выбор программного

обеспечения для разработки портала. Выполнение курсового проекта включает классификацию веб-ресурсов, анализ целевой аудитории веб-ресурса, разработку технического задания, выбор языка и технологии программирования, выбор системы управления базами данных, выбор веб-сервера, структуру веб-ресурса, разработку дизайн-макета веб-ресурса, разработку модулей веб-ресурса.

Тематика курсовой работы определяется преподавателем. Студенту предоставляется право выбора одной из предложенных предметных областей для разработки информационного портала организации или предложения своей предметной области с обоснованием целесообразности разработки. Предметные области для курсовой работы:

1. Дизайн-студия.
2. Школа.
3. Больница.
4. Поликлиника.
5. Организация, оказывающая услуги связи.
6. Банк.
7. ВУЗ.
8. Библиотека.
9. Архив.
10. Отделение полиции.
11. Паспортный стол.
12. Строительная организация.
13. Компьютерные курсы.
14. Страховая компания.
15. Городская администрация.
16. ЖКХ.
17. Интернет-провайдеры.
18. Рекламное агентство.
19. Букмекерская контора.
20. Центр занятости населения.
21. Отделение СМИ.
22. Адвокатская контора.
23. Туристическое агентство.
24. Налоговая инспекция.
25. Регистрационная палата.
26. Центр статистики.
27. Пенсионный фонд.
28. Торговый центр.
29. Типография.
30. Консультативно-диагностический центр.

При выполнении курсового проекта рекомендуется использовать следующее учебно-методическое обеспечение: 1, 2, 3, 4.

Образовательные технологии

При реализации учебного материала курса используются различные образовательные технологии, способствующие созданию атмосферы свободной и творческой дискуссии как между преподавателем и студентами, так и в студенческой группе. Целью при этом является выработка у студентов навыков и компетенций, позволяющих самостоятельно вести исследовательскую и научно-педагогическую работу.

Практическая подготовка при реализации учебной дисциплины организуется путем проведения практических занятий, предусматривающих участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Практическая подготовка также включает в себя занятия лекционного типа, которые предусматривают передачу учебной информации обучающимся, необходимой для последующего выполнения работ, связанных с будущей профессиональной деятельностью.

Аудиторные занятия проводятся в виде лекций с использованием ПК и компьютерного проектора, практических занятий, с использованием ПК при проведении расчетов. Самостоятельная работа студентов проводится под руководством преподавателей, с оказанием консультаций и помощи при подготовке к практическим занятиям.

Фонд оценочных средств

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий текущего, рубежного и промежуточного контроля по дисциплине.

Связь между формируемыми компетенциями и формами контроля их освоения представлена в следующей таблице:

№ п/п	Наименование контролируемых разделов (темы)	Код и наименование индикатора достижения компетенций	Наименование оценочного средства
Входной контроль			
1	Входной контроль		Вопросы входного контроля (устно)
Аттестация разделов, текущий контроль успеваемости			
2	Обзор технологии сервлетов и их архитектура	З-ОПК-6, У- ОПК-6, В- ОПК-6, З-ПК-12, У-ПК-12, В-ПК-12, З-ПК-13, У-ПК-13, В-ПК-13	Тестирование (письменно)
3	Основные понятия технологии Java Server Pages	З-ОПК-6, У- ОПК-6, В- ОПК-6, З-ПК-12, У-ПК-12, В-ПК-12, З-ПК-13, У-ПК-13, В-ПК-13	Тестирование (письменно)
Промежуточная аттестация			
4	Зачет	З-ОПК-6, У- ОПК-6, В- ОПК-6, З-ПК-12, У-ПК-12, В-ПК-12, З-ПК-13, У-ПК-13, В-	Вопросы к зачету (письменно)

		ПК-13	
5	Разработка корпоративных приложений на языке Java	З-ОПК-6, У- ОПК-6, В- ОПК-6, З-ПК-12, У-ПК-12, В-ПК-12, З-ПК-13, У-ПК-13, В-ПК-13	Тестирование (письменно)
6	Серверы приложений	З-ОПК-6, У- ОПК-6, В- ОПК-6, З-ПК-12, У-ПК-12, В-ПК-12, З-ПК-13, У-ПК-13, В-ПК-13	Тестирование (письменно)
Промежуточная аттестация			
7	Экзамен	З-ОПК-6, У- ОПК-6, В- ОПК-6, З-ПК-12, У-ПК-12, В-ПК-12, З-ПК-13, У-ПК-13, В-ПК-13	Вопросы к экзамену (письменно)

Входной контроль предназначен для выявления пробелов в знаниях студентов и готовности их к получению новых знаний. Оценочные средства для входного контроля представляют собой вопросы, которые задаются студентам в устной форме.

Перечень вопросов входного контроля

1. Дайте понятие перегрузки методов.
2. Опишите методы функционального и процедурного типа.
3. Опишите модификаторы доступа.
4. Опишите объявление классов.
5. Опишите объявление методов.
6. Опишите объявление полей.
7. Опишите понятие класса.
8. Опишите принцип работы виртуальной машины Java.
9. Опишите сигнатуру метода.
10. Опишите стандартные потоки ввода-вывода.
11. Опишите статические методы классов.
12. Охарактеризуйте инкапсуляцию.
13. Охарактеризуйте наследование.
14. Охарактеризуйте полиморфизм.
15. Перечислите типы отношений классов

Текущий контроль – это непрерывно осуществляемый мониторинг уровня усвоения знаний и формирования умений и навыков в течение семестра. Текущий контроль знаний, умений и навыков студентов осуществляется в ходе учебных (аудиторных) занятий, проводимых по расписанию. Формами текущего контроля выступают опросы на практических занятиях.

В качестве оценочного средства аттестации раздела используются тесты, итоговая контрольная работа. Тест содержит от 10 вопросов. На выполнение задания отводится 30 минут. Тест – это форма контроля, направленная на проверку уровня освоения контролируемого теоретического и практического материала.

Примерный перечень тестовых заданий:
Вариант 1

№	вопрос
1	<p>Какой метод сервлета FirstServlet будет вызван при активизации ссылки следующего HTML-документа?</p> <pre><html> <body> OK! </body> </html></pre> <p>Соответствующий сервлету тег <code><url-pattern></code> в файле <code>web.xml</code> имеет вид: <code><url-pattern>/FirstServlettest</url-pattern></code></p> <ol style="list-style-type: none"> 1) <code>doGET()</code> 2) <code>doGet()</code> 3) <code>doPOST()</code> 4) <code>doPost()</code>
2	<p>Укажите корректную последовательность методов в жизненном цикле <code>javax.servlet.Servlet</code>:</p> <ol style="list-style-type: none"> 1) <code>init</code>, <code>service</code>, <code>destroy</code> 2) <code>init</code>, <code>service</code>, <code>start</code>, <code>response</code>, <code>destroy</code> 3) <code>init</code>, <code>start</code>, <code>doPost</code>, <code>doGet</code>, <code>request</code>, <code>response</code>, <code>destroy</code> 4) <code>service</code>, <code>request</code>, <code>response</code>, <code>end</code>
3	<p>Какой метод за жизненный цикл сервлета может быть выполнен более одного раза?</p> <ol style="list-style-type: none"> 1) <code>init()</code> 2) <code>service()</code> 3) <code>destroy()</code> 4) конструктор сервлета
4	<p>Метод <code>destroy()</code> интерфейса <code>javax.servlet.Servlet</code> (несколько вариантов):</p> <ol style="list-style-type: none"> 1) вызывается автоматически сервлет-контейнером если сервлет больше не будет использоваться для обработки запросов от клиентов 2) позволяет программисту освободить все ресурсы, которые захватил сервлет 3) является последним методом в жизненном цикле сервлета 4) является предпоследним методом в жизненном цикле сервлета
5	<p>Каким образом в методе <code>init()</code> сервлета получить параметр инициализации сервлета с именем "PARAM"? (несколько вариантов):</p> <ol style="list-style-type: none"> 1) <code>this.getInitParameter("PARAM");</code> 2) <code>ServletConfig.getInitParameter("PARAM");</code> 3) <code>getServletConfig().getInitParameter("PARAM");</code> 4) <code>HttpServlet.getInitParameter("PARAM");</code>
6	<p>Какие типы протоколов поддерживает <code>HttpServlet</code>?</p> <ol style="list-style-type: none"> 1) FTP 2) HTTP 3) HTTPS 4) SSL
7	<p>Выберите верное утверждение относительно объекта класса <code>ServletConfig</code>:</p> <ol style="list-style-type: none"> 1) Объект <code>ServletConfig</code> представляет конфигурацию для всего веб-приложения в котором он задеплоен 2) Объект <code>ServletConfig</code> представляет конфигурацию для одного определенного сервлета 3) Все вышеприведенные выражения являются верными 4) Ни одно из вышеприведенных выражений не является верным
8	<p>Является ли необходимым наличие дескриптора приложения <code>web.xml</code> в Servlet 3.0?</p> <ol style="list-style-type: none"> 1) Нет, так как существуют аннотации для декларирования веб компонентов 2) Да, так как аннотации для декларирования веб компонентов не избавляют от

	<p>необходимости иметь дескриптор web.xml</p> <p>3) Нет, так как в Servlet 3.0 введен новый дескриптор приложения: application.xml</p> <p>4) Да, так как с версии Servlet 2.8 ничего не изменилось</p>
9	<p>Жизненный цикл сервлета управляется:</p> <p>1) контекстом сервлетов (Servlet context)</p> <p>2) контейнером сервлетов (Servlet container)</p> <p>3) контекстом приложения (Application context)</p> <p>4) дескриптором развертывания (Deployment descriptor)</p>
10	<p>Какой из следующих интерфейсов предоставляет метод getWriter для получения экземпляра класса PrintWriter?</p> <p>1) HttpServletRequest</p> <p>2) HttpServletResponse</p> <p>3) ServletContext</p> <p>4) ServletResponse</p>
11	<p>Синтаксис скрипта JSP имеет вид:</p> <p>1) <@ код @></p> <p>2) <%-- код --%></p> <p>3) <%@ код %></p> <p>4) <% код %></p>
12	<p>Контекст страницы (доступ на уровне страницы) JSP:</p> <p>1) pageScope</p> <p>2) requestScope</p> <p>3) sessionScope</p> <p>4) applicationScope</p>
13	<p>Контекст запроса JSP:</p> <p>1) pageScope</p> <p>2) requestScope</p> <p>3) sessionScope</p> <p>4) applicationScope</p>
14	<p>Небольшие фрагменты информации, которые размещаются на стороне клиента:</p> <p>1) скрипты</p> <p>2) бины</p> <p>3) закладки</p> <p>4) сеансы</p>
15	<p>Теги для работы с xml-документами:</p> <p>1) <% @ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/xml" %></p> <p>2) <% taglib prefix="c" uri="http://java.sun.com/jsp/jstl/xml" %></p> <p>3) <% @ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/sql" %></p> <p>4) <@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/sql" %></p>
16	<p>Хеш-таблица - это структура данных, реализующая и позволяющая хранить пары (ключ, значение) и выполнять операции (несколько вариантов):</p> <p>1) идентификации ID (ключ, значение)</p> <p>2) добавления ADD (ключ, значение)</p> <p>3) поиска FIND (ключ)</p> <p>4) удаления REMOVE (ключ)</p>
17	<p>Чтобы создать пользовательский тэг первым делом необходимо:</p> <p>1) создать java-класс обработчик (Handler) тэга</p> <p>2) создать TLD (tag library descriptor)-файл</p> <p>3) создать тэг-файл с использованием JSP-синтаксиса</p> <p>4) связать файл-обработчик с TLD-файлом</p>
18	<p>Синтаксис операции <jsp:forward> имеет вид:</p> <p>1) <@jsp:forward page="относительный URL"@></p> <p>2) <jsp:forward page=*относительный URL*/></p> <p>3) <jsp:forward page="относительный URL"/></p>

	4) <code><jsp:forward page="относительный URL"></code>
19	Синтаксис операции <code><jsp:include></code> имеет вид: 1) <code><jsp: flush="true" include page="относительный URL" /></code> 2) <code><jsp: flush="false" include page="относительный URL"/></code> 3) <code><jsp:include page="относительный URL" flush="true"/></code> 4) <code><jsp:include page="относительный URL" flush="false"/></code>
20	Какой подкласс используется для обработки тела тэга JSP: 1) TagSupport 2) SimpleTagSupport 3) BodyTagSupport 4) PageTagSupport
21	Файл исходного кода JSP может содержать (несколько вариантов): 1) встроенное изображение (Image) 2) директивы (Directives) 3) операции (Actions) 4) JavaScript (JS)
22	Синтаксис директивы JSP page имеет вид: 1) <code><% page att="значение" %></code> 2) <code><@ page att="значение" @></code> 3) <code><%@ page att="значение" %></code> 4) <code><%@ page="значение" %></code>

Вариант 2

№	вопрос
1	Что нужно выполнить чтобы при доступе к сервлету заработала стандартная авторизация tomcat на основе аутентификации BASIC (несколько вариантов)? 1) Определить сервлет в web.xml 2) В web.xml добавить элемент url-pattern, соответствующий сервлету в элемент web-resource-collection, в элементе auth-constraint прописать роли, имеющие доступ к ресурсу 3) В web-xml в элементе login-config указать метод аутентификации, в элементе security-role указать роли для аутентификации 4) Сконфигурировать роли и пользователей в tomcat-users
2	Куда лучше всего поместить данные для того, чтобы они были доступны в последующих запросах одного и того же пользователя? 1) request 2) session 3) servlet context 4) page
3	Servlet mapping определяет: 1) Связь между url-шаблоном и сервлетом 2) Связь между url-шаблоном и страницей запроса 3) Связь между url-шаблоном и страницей ответа 4) Связь между url-шаблоном и контекстом сервлета (Servlet context)
4	Какой метод имеет класс javax.servlet.http.HttpSessionEvent? 1) getSession 2) getSessionId 3) getSessionEvent 4) getSessionAttribute
5	Какой класс/интерфейс непосредственно расширяет/реализует класс HttpServlet? 1) Servlet 2) Http 3) GenericServlet

	4) HttpServletImpl
6	<p>Контейнер вызывает метод init() экземпляра сервлета... (несколько вариантов):</p> <ol style="list-style-type: none"> 1) при каждом запросе к сервлету, при котором создается новый поток 2) только один раз за жизненный цикл экземпляра 3) когда сервлет создается впервые 4) если время жизни сессии пользователя, от которого пришел запрос, истекло
7	<p>Синтаксис выражения JSP имеет вид:</p> <ol style="list-style-type: none"> 1) <%@= выражение %> 2) <%= выражение %> 3) <%! = выражение !%> 4) <%!= выражение %>
8	<p>Сеанс инициализируется однократно, когда пользователь входит на сайт. Сеанс существует до тех пор, пока не будут выполнены следующие действия (несколько вариантов):</p> <ol style="list-style-type: none"> 1) пользователь закрыл браузер 2) пользователь свернул браузер 3) в течение определенного времени (обычно 30 мин) пользователь не выполнил никаких действий 4) пользователь выполнил процедуру выхода
9	<p>Для создания пользовательского тэга необходимо создать Java-класс поддержки тэга, реализующий интерфейс:</p> <ol style="list-style-type: none"> 1) javax.servlet.jsp.jspexception 2) javax.servlet.jsp.pagecontext 3) javax.servlet.jsp.tagext.Tag 4) javax.servlet.jsp.tagsupport
10	<p>Константы интерфейса Tag, при которой тело тэга передается без изменений:</p> <ol style="list-style-type: none"> 1) SKIP_BODY 2) EVAL_BODY_INCLUDE 3) SKIP_PAGE 4) EVAL_PAGE
11	<p>Тег TLD- файла, который определяет расположение библиотеки:</p> <ol style="list-style-type: none"> 1) <short-name> 2) <display-name> 3) <uri> 4) <description>
12	<p>Теги для работы с базами данных:</p> <ol style="list-style-type: none"> 1) < taglib prefix="c" uri="http://java.sun.com/jsp/jstl/sql" > 2) <% taglib prefix="c" uri="http://java.sun.com/jsp/jstl/sql" %/> 3) <% taglib prefix="c" uri="http://java.sun.com/jsp/jstl/sql" %> 4) <% @ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/sql" %>
13	<p>Контекст приложения JSP:</p> <ol style="list-style-type: none"> 1) pageScope 2) requestScope 3) sessionScope 4) applicationScope
14	<p>В Интернете протокол, использующий службу с установлением логического соединения, имеет название:</p> <ol style="list-style-type: none"> 1) TCP 2) IP 3) UDP 4) HTTP
15	<p>Подход к построению пользовательских интерфейсов веб-приложений, при котором веб-страница, не перезагружаясь, асинхронно загружает нужные пользователю данные:</p>

	1) ASP 2) AJAX 3) JSP 4) XML
16	Сеанс (сессия) может быть определен как: 1) взаимодействие, которое происходит в течение некоторого интервала времени между клиентом и сервером 2) последовательность связанных между собой взаимодействий, которые происходят в течение некоторого интервала времени между двумя клиентами 3) последовательность связанных между собой взаимодействий, которые происходят в течение некоторого интервала времени между клиентом и сервером 4) последовательность связанных между собой запросов на авторизацию, которые происходят в течение некоторого интервала времени между клиентом и сервером
17	Основные теги позволяющие реализовывать циклы, условия, выводить информацию на экран: 1) <code><%= taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %/></code> 2) <code><% @ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %></code> 3) <code><% taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %></code> 4) <code><% @= taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %/></code>
18	Для наблюдения за сеансом (session tracking) можно использовать (несколько вариантов): 1) информацию HTTP-протокола, скрытые поля HTML-форм 2) информацию application 3) закладки (Cookie) 4) расширенную информацию о пути и перезаписывание URL
19	Синтаксис директивы JSP include имеет вид: 1) <code><% include file="url" %></code> 2) <code><% @ include ="url" %></code> 3) <code><@ include ="url" %></code> 4) <code><% @ include file="url" %></code>
20	Для тэга верхнего уровня родительский тэг определен как: 1) null 2) parent 3) first 4) main
21	Синтаксис операции <code><jsp:setProperty></code> имеет вид: 1) <code><jsp:setProperty att=значение*/></code> 2) <code><jsp:setProperty att=*значение*/></code> 3) <code><jsp:setProperty att=значение/></code> 4) <code><jsp:setProperty att=/*значение*/></code>
22	Метод, который должен освободить ресурсы, которые использовались классом поддержки тэга JSP: 1) doStartTag 2) release 3) doEndTag 4) reset

Критерии оценки тестовых заданий:

1. Полнота знаний теоретического контролируемого материала.
2. Количество правильных ответов.

Промежуточная аттестация осуществляется в форме зачета и экзамена

Для промежуточной аттестации предусмотрены теоретические и экзаменационные вопросы.

По итогам обучения выставляются зачет и экзамен.

Примерный перечень вопросов к зачету

1. Понятие сервлета.
2. Преимущества сервлетов.
3. Интерфейс ServletContext.
4. Интерфейс ServletConfig.
5. Интерфейсы ServletRequest и HttpServletRequest.
6. Интерфейсы ServletResponse и HttpServletResponse.
7. Паттерн проектирования MVC.
8. Паттерн проектирования MVP.
9. Паттерн проектирования MVVM.

Примерный перечень вопросов к экзамену

1. Директивы JSP.
2. Выражения JSP.
3. Скриптлеты JSP.
4. Декларации JSP.
5. Операции JSP.
6. JSTL.
7. Пользовательские тэги JSP.
8. Сеанс (Session) JSP.
9. Закладки (Cookie) JSP.

Шкалы оценки образовательных достижений

Баллы (итоговой рейтингово й оценки)	Оценка (балл за ответ на зачете)	Требования к знаниям
100-60	<i>«зачтено» 30-50 баллов</i>	– Оценка «зачтено» если он имеет знания основного материала, если он прочно усвоил программный материал, последовательно, четко и логически стройно его излагает его на зачете, владеет необходимыми навыками и приемами их выполнения, умеет тесно увязывать теорию с практикой – Учебные достижения в семестровый период и результатами рубежного контроля демонстрируют достаточную степень овладения программным материалом.
59-0	<i>«не зачтено» 29 - 0 баллов</i>	– Оценка «не зачтено» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки. Как правило, оценка «не зачтено» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

		– Учебные достижения в семестровый период и результатами рубежного контроля демонстрировали не высокую степень овладения программным материалом по минимальной планке.
--	--	--

Баллы (итоговой рейтинговой оценки)	Оценка (балл за ответ на экзамене)	Требования к знаниям
90-100	Отлично	Ответы на поставленные вопросы полные, четкие и развернутые. Решения задач логичны, доказательны и демонстрируют аналитические и творческие способности студента.
70-89	Хорошо	Даются полные ответы на поставленные вопросы. Показано умение выделять причинно-следственные связи. При решении задач допущены незначительные ошибки, исправленные с помощью «наводящих» вопросов преподавателя.
60-69	Удовлетворительно	Ответы на вопросы и решения поставленных задач недостаточно полные. Логика и последовательность в решении задач имеют нарушения. В ответах отсутствуют выводы.

Учебно-методическое и информационное обеспечение учебной дисциплины

Основная литература

1. Свердлов, С. З. Языки программирования и методы трансляции: учебное пособие / С. З. Свердлов. — 2-е изд., испр. — Санкт-Петербург : Лань, 2019. — 564 с. <https://e.lanbook.com/book/173116>

Дополнительная литература

2. Беляев, С. А. Разработка игр на языке JavaScript : учебное пособие / С. А. Беляев. — 3-е изд., стер. — Санкт-Петербург: Лань, 2020. — 128 с. <https://e.lanbook.com/reader/book/138172/#2>

3. Ли, П. Архитектура интернета вещей / П. Ли ; перевод с английского М. А. Райтман. — Москва: ДМК Пресс, 2019. — 454 с. <https://e.lanbook.com/reader/book/112923/#6>

Программное обеспечение и Интернет-ресурсы

1. Электронно-библиотечная система «Лань».
2. Электронно-библиотечная система «Юрайт»
3. Российская национальная библиотека - <http://www.nlr.ru/>
4. Поисковая система - <http://www.rambler.ru>.
5. Поисковая система - <http://www.yandex.ru>.
6. Гарант - <http://base.garant.ru/>.
7. Интернет-Университет Информационных Технологий -

<http://www.intuit.ru>

Для проведения практических занятий и выполнения самостоятельной работы используются учебные компьютерные классы с выходом в Интернет и лицензионным программным обеспечением.

Материально-техническое обеспечение учебной дисциплины

Лекционные занятия проводятся в аудиториях: учебная мебель, учебная доска, комплект мультимедийного оборудования

Практические занятия проводятся в компьютерных классах: учебная мебель, учебная доска, комплект мультимедийного оборудования, персональные компьютеры.

Для самостоятельной работы обучающихся имеется: читальный зал с выходом в сеть Интернет: Учебная мебель, комплект мультимедийного оборудования, персональные компьютеры, МФУ.

Учебно-методические рекомендации для студентов

1. Указания для прослушивания лекций

Перед началом занятий внимательно ознакомиться с учебным планом проведения лекций и списком рекомендованной литературы.

Перед посещением очередной лекции освежить в памяти основные концепции пройденного ранее материала. Подготовить при необходимости вопросы преподавателю. Не надо опасаться, что вопросы могут быть простыми.

На лекции основное внимание следует уделять не формулам и математическим выкладкам, а содержанию изучаемых вопросов, определениям и постановкам задач.

В процессе изучения лекционного курса необходимо по возможности часто возвращаться к основным понятиям и методам решения задач (здесь возможен выборочный контроль знаний студентов).

Желательно использовать конспекты лекций, в которых используется принятая преподавателем система обозначений.

Для более подробного изучения курса следует работать с рекомендованными литературными источниками и вновь появляющимися источниками.

2. Указания для участия в практических занятиях

Перед посещением уяснить тему практического занятия и самостоятельно изучить теоретические вопросы.

В конце практического занятия при необходимости выяснить у преподавателя неясные вопросы.

Основные результаты выполнения работы необходимо распечатать.

3. Указания для выполнения самостоятельной работы

Получить у преподавателя задание и список рекомендованной литературы. Изучение теоретических вопросов следует проводить по

возможности самостоятельно, но при затруднениях обращаться к преподавателю.

Подготовить письменный отчет о проделанной работе.

При выполнении заданий по усмотрению преподавателя работа может быть оценена без письменного отчета на основе ответов на контрольные вопросы, при условии активной самостоятельной работы.

Подготовить ответы на контрольные вопросы.

Методические рекомендации для преподавателей

1. Указания для проведения лекций

На первой вводной лекции сделать общий обзор содержания курса и отметить новые методы и подходы к решению задач, рассматриваемых в курсе, довести до студентов требования кафедры, ответить на вопросы.

При подготовке к лекционным занятиям необходимо продумать план его проведения, содержание вступительной, основной и заключительной части лекции, ознакомиться с новинками учебной и методической литературы, публикациями периодической печати по теме лекционного занятия. Уточнить план проведения семинарского занятия по теме лекции. Перед изложением текущего лекционного материала напомнить об основных итогах, достигнутых на предыдущих лекциях. С этой целью задать несколько вопросов аудитории и осуществить выборочный контроль знания студентов.

В ходе лекционного занятия преподаватель должен назвать тему, учебные вопросы, ознакомить студентов с перечнем основной и дополнительной литературы по теме занятия. Раскрывая содержание учебных вопросов, акцентировать внимание студентов на основных категориях, явлениях и процессах, особенностях их протекания. Раскрывать сущность и содержание различных точек зрения и научных подходов к объяснению тех или иных явлений и процессов.

Следует аргументировано обосновать собственную позицию по спорным теоретическим вопросам. Приводить примеры. Задавать по ходу изложения лекционного материала риторические вопросы и самому давать на них ответ. Это способствует активизации мыслительной деятельности студентов, повышению их внимания и интереса к материалу лекции, ее содержанию. Преподаватель должен руководить работой студентов по конспектированию лекционного материала, подчеркивать необходимость отражения в конспектах основных положений изучаемой темы, особо выделяя, категориальный аппарат. В заключительной части лекции необходимо сформулировать общие выводы по теме, раскрывающие содержание всех вопросов, поставленных в лекции. Объявить план очередного практического занятия, дать краткие рекомендации по подготовке студентов к семинару. Определить место и время консультации студентам, пожелавшим выступить на занятии с докладами.

На последней лекции уделить время для обзора наиболее важных положений, рассмотренных в курсе.

2. Указания для проведения практических занятий

Четко обозначить тему занятия.

Обсудить основные понятия, связанные с темой.

В процессе решения задач вести дискуссию со студентами о правильности применения теоретических знаний.

Отмечать студентов, наиболее активно участвующих в решении задач и дискуссиях.

В конце занятия задать аудитории несколько контрольных вопросов.

3. Указания по контролю самостоятельной работы студентов

По усмотрению преподавателя задание на самостоятельную работу может быть индивидуальным или фронтальным.

При использовании индивидуальных заданий требовать от студента письменный отчет о проделанной работе.

При применении фронтальных заданий вести коллективные обсуждения со студентами основных теоретических положений.

С целью контроля качества выполнения самостоятельной работы требовать индивидуальные отчеты (допустимо вместо письменного отчета применять индивидуальные контрольные вопросы).

Программа составлена в соответствии с требованиями ОС НИЯУ МИФИ по направлению 09.03.02 Информационные системы и технологии

Рабочую программу составил

Ст. преподаватель кафедры



А.Г. Мотков

Рецензент: доцент



И.А. Штырова

Программа одобрена на заседании УМКН «Информационные системы и технологии» от 15.11.2021 года, протокол №2.

Председатель учебно-методической комиссии



О.В. Виштак