




МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Дальневосточный федеральный университет»
(ДФУ)

ИНЖЕНЕРНАЯ ШКОЛА

«СОГЛАСОВАНО»

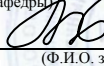
Руководитель ОП
«Прикладная механика»


(подпись) Озерова Г.П.
(Ф.И.О. рук.ОП)

«22» июня 2016г.

«УТВЕРЖДАЮ»

Заведующий кафедрой
Механики и математического моделирования
(название кафедры)


(подпись) Бочарова А.А.
(Ф.И.О. зав. каф.)

«22» июня 2016г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ (РПУД)
ИНЖЕНЕРНЫЕ WEB-ТЕХНОЛОГИИ

Направление подготовки: 15.03.03 Прикладная механика

Профиль подготовки:

«Математическое и компьютерное моделирование механических систем и процессов»

Форма подготовки очная

курс 3,4 семестр 6, 7
лекции 54 час.
практические занятия - час.
лабораторные работы 72 час.
в том числе с использованием МАО лек. 8час. /пр.- час. /лаб.72 час.
всего часов аудиторной нагрузки 126 час.
в том числе с использованием МАО 80 час.
самостоятельная работа 126 час.
в том числе на подготовку к экзамену 36 час.
контрольные работы -
курсовая работа / курсовой проект -
зачет 6 семестр
экзамен 7 семестр

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования Дальневосточного федерального университета, принятого решением Ученого совета ДВФУ, протокол от 25.02.2016 № 02-16, введенного в действие приказом ректора ДВФУ от 10.03.2016 № 12-13-391

Рабочая программа обсуждена на заседании кафедры Механики и математического моделирования, протокол № 9 от «21» июня 2016 г.

Заведующий кафедрой: к.ф.-м.н., проф. Бочарова А.А.
Составитель: к.т.н., доцент Озерова Г.П.

ABSTRACT

Bachelor's degree in 15.03.03 Applied Mechanics

Study profile “Mathematical and Computer Modeling Mechanical Systems and Processes”

Course title: Engineering Web Technologies

Variable part Block 1, 7 credits

Instructor: Ozerova Galina Pavlovna

At the beginning of the course a student should be able to:

- should know how to use different information sources: books, textbooks, references, internet;
- should have an algorithmic mind;
- can use computational machines;
- understanding basics of structural and procedural programming.

Learning outcomes:

PC 2 – the ability to apply physical and mathematical tools for theoretical, computational and experimental research, methods of mathematical and computer modeling in the professional activity.

PC 5 – the ability to create descriptions research and development projects, process and analyze the results, prepare data for reports and presentations, writing reports, articles and other scientific and technical documentation;

PC 6 – the ability to use software, computer graphics and visualization of the research activity results, preparing papers, reports and articles using modern information technology, text and graphics editors, print media.

Course description: The content of the course covers the following range of issues: technology design and creation of WEB-sites, Hypertext Markup Language (HTML), Cascading Style Sheets, client and server-side scripting languages, database. All stages of design WEB-sites discussed in detail in the course. Theoretical and practical knowledge of programming will allow students more confident to navigate the complex and diverse phenomena of the modern

development of the information society, as well as some practical skills in the use of computer technology in teaching, research and experimental activities.

Main course literature:

1. Bozhenyuk A.V. Intelligent web technologies: a textbook for high schools/ A.V. Bozhenyuk, E.M. Kotov. - Rostov-on-Don: Phoenix, 2009. - 382 p.
<http://lib.dvfu.ru:8080/lib/item?id=chamo:419137&theme=FEFU>

2. Burlakov N.N. Creating hypertext environment: a textbook for high schools. Part 1. Static coding / NN Burlakov. - Vladivostok: Publishing house of the Far Eastern Federal University, 2012. - 85 p.
<http://lib.dvfu.ru:8080/lib/item?id=chamo:681561&theme=FEFU>

3. Shustov L.I. Databases: the textbook / LI Shustov, OV Tarakanov. - M.: INFRA-M, 2016. - 336 p. <http://znanium.com/go.php?id=491069>

4. V.A. Dronov. HTML 5, CSS 3, and Web 2.0. Development of Web-sites today. - SPb.: BHV-Petersburg, 2011. - 414 p.
<http://znanium.com/bookread.php?book=351455>

5. Odinochkina S.V. PHP Web-programming [electronic resource]: a teaching aid. - Spb.: ITMO (St. Petersburg National Research University of Information Technologies, Mechanics and Optics), 2012. - 79 p.
http://e.lanbook.com/books/element.php?pl1_id=43562

6. Zudilova T.V. Web-programming HTML [electronic resource]: teaching manual / T.V. Zudilova, M.L. Burkov. - Spb.: ITMO (St. Petersburg National Research University of Information Technologies, Mechanics and Optics), 2012. - 70 p. - Access: http://e.lanbook.com/books/element.php?pl1_id=40724

Form of final knowledge control: pass, exam.

Аннотация дисциплины «Инженерные web-технологии»

Дисциплина «Инженерные web-технологии» разработана для студентов, обучающихся по направлению подготовки 15.03.03 «Прикладная механика», профиль «Математическое и компьютерное моделирование механических систем и процессов» и является дисциплиной выбора вариативной части Блока 1 «Дисциплины (модули)» учебного плана (Б1.В.ДВ.7.1).

Трудоемкость дисциплины составляет 252 часа (7 зачетных единиц). Учебным планом предусмотрены лекционные занятия (54 часа), практические, лабораторные работы (72 часа) и самостоятельная работа студентов (126 часов из них 36 часов на экзамен). Дисциплина реализуется на 3,4 курсе в 6,7 семестре. Форма промежуточной аттестации – зачет (6 семестр), экзамен (7 семестр).

Дисциплина «Инженерные web-технологии» логически связана с дисциплинами «Программирование в инженерных задачах», «Информационные технологии в прикладной механике», «Компьютерное моделирование механических систем и процессов».

Цель: является освоение современных интернет-технологий и сопутствующих областей знаний, методов и средств создания web-ресурсов, их продвижения и применения в различных видах деятельности.

Задачи:

1. Дать целостное представление о возможностях и структуре глобальной сети Internet.
2. Дать представление о развитии и применении Internet технологий в профессиональной деятельности.
3. Изучить методы и средства разработки web-приложений;
4. Изучить основы проектирования и доступа к базам данных;
5. Сформировать навыки практической работы по созданию сайтов.
6. Развить логическое и алгоритмическое мышление.

Для успешного изучения дисциплины «Инженерные web-технологии» у обучающихся должны быть сформированы следующие предварительные компетенции:

- владение навыками работы с различными источниками информации: книгами, учебниками, справочниками, Интернет;
- способность к алгоритмическому мышлению;
- знание основ технологии программирования;
- владение методами и средствами структурного, событийного и объектно-ориентированного программирования.

В результате изучения данной дисциплины у обучающихся формируются следующие общепрофессиональные и профессиональные компетенции.

Код и формулировка компетенции	Этапы формирования компетенции	
ПК-2 – способностью применять физико-математический аппарат, теоретические, расчетные и экспериментальные методы исследований, методы математического и компьютерного моделирования в процессе профессиональной деятельности	Знает	возможности и структуру сети Internet, способы поиска информации; основы теории баз данных
	Умеет	использовать информацию из сети Internet для решения стандартных задач профессиональной деятельности;
	Владеет	методикой применения баз данных и Интернет-технологии для решения задач профессиональной деятельности.
ПК-5 – способностью составлять описания выполненных научно-исследовательских работ и разрабатываемых проектов, обрабатывать и анализировать полученные результаты, готовить данные для составления отчетов и презентаций, написания докладов, статей и другой научно-технической документации	Знает	основные технологии разработки web-приложений; основы HTML и CSS, скриптовых языков программирования; основы разработки web-сайтов и приложений
	Умеет	создавать статические web-сайты для представления информации о разрабатываемых проектах, решаемых задачах
	Владеет	методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов
ПК-6 – способностью применять	Знает	основные средства компьютерной

Код и формулировка компетенции	Этапы формирования компетенции	
программные средства компьютерной графики и визуализации результатов научно-исследовательской деятельности, оформлять отчеты и презентации, готовить рефераты, доклады и статьи с помощью современных офисных информационных технологий, текстовых и графических редакторов, средств печати		графики, применяемые в web-приложениях
	Умеет	использовать стандартные библиотеки скриптовых языков для создания элементов web-страниц
	Владеет	технологией разработки web-сайтов для графического отображения и визуализации результатов научно-исследовательской деятельности

Для формирования вышеуказанных компетенций в рамках дисциплины «Инженерные web-технологии» применяются следующие методы активного/интерактивного обучения:

- лекция пресс-конференция;
- лекция «вдвоем»;
- игровое проектирование;
- групповая консультация.

I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

РАЗДЕЛ I. Инструменты создания сайтов (10 час.)

Тема 1. Архитектура интернет, развитие интернет-технологий (2 час.)

История Интернет. Возможности Интернет. Структура Интернет. Культура Интернет. Как работает Интернет. Основные компоненты. Протоколы. Адресация в сети Интернет. Схема поиска IP-адреса по доменному имени. Сервисы Интернет (основные службы). Утилиты.

Тема 2. Язык HTML (язык гипертекстовой разметки) (4час.)

Язык HTML. Основы, версии и стандарты языка HTML. Структура HTML документа. Основные разделы кода веб-страницы. Понятие тэгов HTML, их типы и оформление. Цвета HTML. Понятие атрибутов. Атрибуты тега BODY. Форматирование текста: заголовки, параграфы, разделители.

Обзор тэгов оформления и выделения текста, а также изменения его размеров и цвета. Понятие гиперссылки. Типы и области применения гиперссылок. Обзор поддерживаемых графических форматов, их различия, ограничения и использование. Тэги вставки изображений на web-страницы и их параметры. Обзор тэгов создания списков различных типов. Принципы вставки спецсимволов и тэги псевдографики. Понятие, использование и принципы построения таблиц. Обзор тэгов создания таблиц и их параметры. Фреймы

Тема 3. Каскадные таблицы стилей(4час.)

Понятие, область применения и принципы построения каскадных таблиц стилей (CSS).Использование CSS на веб-страницах, способы задания стилей и оформления отдельных элементов. Обзор инструкций CSS для оформления, изменения размеров и цвета текста. Обзор инструкций CSS для установки фоновых изображений и их параметры. Понятие и использование обрамления. Обзор инструкций CSS для создания обрамлений и их параметры (установка формы, цвета, ширины и позиционирования бордюров, а также отступов от них и т.д.).Понятие, инструкции и использование трёхмерного позиционирования объектов веб-страницы с помощью CSS. Примеры позиционирования отдельных элементов веб-страницы.Управление отображением элементов страницы с помощью CSS. Обзор инструкций управления отображением и их взаимодействие со скриптами. Примеры управления отображение объектов (создание исчезающих и меняющихся объектов).Понятие и использование фильтров CSS. Инструкции и примеры создания фильтров.

РАЗДЕЛ II. Программирование на стороне клиента (16 час.)

Тема 1. Обзор технологий web-программирования(2час.)

Развитие HTML. Сценарий и обработка события. События в динамическом HTML. Связывание кода с событиями. Создание сценария. Внедрение сценария в HTML. Понятие, типы и назначение скриптов. Понятие и использование форм на веб-страницах. Принципы взаимодействия формы со скриптами. Обзор тэгов для создания элементов формы (текстовых

полей, выпадающих списков и т.д.). Понятие, типы и использование событий на веб-страницах. Примеры использования некоторых событий. Создание интерактивных объектов на веб-странице.

Тема 2. Введение в JavaScript (4час.)

JavaScript как основной язык сценариев для Web. Сферы использования JavaScript. Основные идеи JavaScript. Структура JavaScript программы. Типовые примеры использования JavaScript-сценариев. Базовые элементы языка. Основные объекты языка. Синтаксис JavaScript. Переменные. Операции. Управляющие структуры и организация циклов. Функции. Объектная модель JavaScript. Обработка событий. Встроенные объекты JavaScript: массивы, строки, даты, Math. Разработка программ.

Тема 3. Модель DOM (2час.)

Объектная модель браузера и документа. Иерархия объектов браузера. Объект window. Свойства, методы и события объекта window. Объект document. Свойства, методы и события объекта document. Объект screen. Свойства, методы и события объекта screen. Объект form. Свойства, методы и события объекта form. Объект image. Свойства, методы и события объекта image. Примеры использования DOM модели.

Тема 4. Библиотеки JavaScript для разработки Web- приложений (2 час.)

Библиотека JQuery. Назначение. Синтаксис и семантика операторов. Функция \$(). Начало работы скрипта. Управление элементами документов и их свойствами. Цепочки методов. Работа с набором элементов. Анимационные эффекты.

Тема 5. Основы HTML 5. Рисование. (4 час.)

Описание HTML5. Отличия HTML5 от HTML4. Элемент canvas. Параметры canvas. Получение canvas. Работа с context. Работа со шрифтом. Отрисовка фигур. Пути. Метрики. Пиксельные манипуляции. Работа с тенями. Трансформация изображения. Примеры реализации. Развитие технологии.

Тема 6. Реализации концепций ООП. Библиотека Mootools (2 час.)

Библиотека Mootools. Классы и объекты. Поля и методы. Инкапсуляция, наследование, полиморфизм. Примеры создания иерархии классов

РАЗДЕЛ III. Основы теории баз данных (16 час.)**Тема 1. Реляционная модель (4 час.)**

Достоинства и недостатки реляционной модели. Базовые понятия. Фундаментальные свойства отношений. Ограничения целостности. Особенности исчисления доменов. Манипуляционная часть: реляционная алгебра и реляционное исчисление. Понятие информационного объекта, отношения. Его структура. Необходимость нормализации. Построение нормализованных отношений.

Тема 2. Основы проектирование реляционных баз данных (6 час.)

Планирование, проектирование и администрирование базы данных. Методология концептуального проектирования баз данных. Методология логического проектирования реляционных баз данных. Построение информационно-логической схемы базы данных.

Тема 3. Основы языка SQL(6 час.)

Основные характеристики языка. Диалекты языка. Язык манипулирования данными (DML). Запросы к базам данных (оператор Select). Простые выборки Запросы на выборку данных: ограниченные выборки, выборки с исключением дубликатов, вычисляемые поля в запросах, выборки с упорядочением. Операция соединения реляционной алгебры. Оператор JOIN. Типы соединения: INNER, LEFT, RIGHT, FULL OUTER, CROSS. Добавление, удаление, изменение данных. Создание, удаление таблиц. Операторы CREATE, DELETE, DROP, UPDATE, INSERT. Синтаксис и семантика запросов действия.

РАЗДЕЛ IV. Программирование на стороне сервера(12 час.)**Тема 1. Web-сервер (4 час.)**

Протокол IP и адресация. Протоколы TCP, UDP и адресация. Архитектура «клиент-сервер». Понятие сетевой службы. Стандартные

сетевые службы. Web-сервер. Протокол HTTP. Формат запроса и ответа HTTP. Основы интерфейса CGI; обзор альтернативных технологий.

Тема 2. Основы языка PHP (4 час.)

Вставка php кода. Определение констант и переменных. Типы данные, преобразование типов. Выражения. Оператор присваивания. Условный оператор. Оператор выбора. Операторы цикла. Примеры программ на PHP.

Метод передачи данных GET. Метод передачи данных POST. Пользовательские функции. Передача параметров по ссылке и значению. Параметры по умолчанию. Локальные и глобальные переменные. Примеры.

Одномерные массивы. Ассоциативные массивы. Многомерные массивы. Функции работы с массивами. Строки. Операции со строками. Функции работы со строками. Реализация наследования в PHP. Перекрытие. Многоуровневое наследование. Множественное наследование.

Тема 3. Организация и доступ к данным (4 час.)

Теория реляционных баз данных, язык запросов SQL. Установка и настройка MySQL. Сервер MySQL и работа с ним на PHP. Подключение к серверу, выполнение запросов. Построение эффективных запросов и повышение быстродействия работы с базой данных. Пример написания интерфейса к базе данных. Функции для работы с базами данных. Получение данных из базы данных. Сохранение данных в базе данных. Получение данных из HTML-форм. Валидация данных. Выбор информации из базы. Динамическое формирование HTML-страницы. Применение баз данных при работе с динамически формируемыми страницами.

II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА

Лабораторные работы (72 час.)

Лабораторная работа 1. Основы HTML.(4 час.)

Лабораторная работа 2. Использование DHTML для разработки сайта.(4 час.)

Лабораторная работа 3. Основы CSS.(4 час.)

Лабораторная работа 4. Создание адаптивного сайта на основе технологии CSS.(4 час.)

Лабораторная работа 5. Использование языка JavaScript для программирования приложений на стороне клиента.(4час.)

Лабораторная работа 6. Создание динамического сайта средствами JavaScript.(4 час.)

Лабораторная работа 7. Создание сайта с применением JavaScript и модели DOM.(4 час.)

Лабораторная работа 8. Создание сайта с использованием библиотеки JQUERY. (4 час.)

Лабораторная работа 9. Программное рисование, использование CANVAS HTML5. (4 час.)

Лабораторная работа 10. Программное рисование и библиотека Mootools (4 час.)

Лабораторная работа 11. Построение информационно-логической структуры базы данных, создание и наполнение таблиц для заданной предметной области (4 час.)

Лабораторная работа 12. Реализация запросов на выборку для различных предметных областей (4 час.)

Лабораторная работа 13. Корректировка данных средствами запросов для различных предметных областей. (4 час.)

Лабораторная работа 14. Основы PHP.(4 час.)

Лабораторная работа 15. Организация доступа к базам данных, PHP и MySQL.(4 час.)

Лабораторная работа 16. Создание динамической страницы на стороне сервера.(4 час.)

Лабораторная работа 17. Создания страницы для отображения информации из базы данных.(4 час.)

Лабораторная работа 18. Разработка клиент-серверного приложения.(4 час.)

III. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Инженерные web-технологии» представлено в Приложении 1 и включает в себя:

- план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;
- характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;
- требования к представлению и оформлению результатов самостоятельной работы;
- критерии оценки выполнения самостоятельной работы.

IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы / темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства		
			текущий контроль	промежуточная аттестация	
1	Инструменты создания сайтов	ПК-5	Знает основные технологии разработки web-приложений; основы HTML и CSS, скриптовых языков программирования; основы разработки web-сайтов и приложений	Тест (ПР-1)	Вопросы 1-24
			Умеет создавать статические web-сайты для представления информации о разрабатываемых проектах, решаемых задачах	Лабораторная работа (ПР-6)	Задание, тип 1
			Владеет методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов	Контрольная работа (ПР-2)	
2	Программир	ПК-6	Знает основные средства	Собеседование	Вопросы

№ п/п	Контролируемые разделы / темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства	
				текущий контроль	промежуточная аттестация
	ование на стороне клиента		компьютерной графики, применяемые в web-приложениях	(УО-1)	1-22
			Умеет использовать стандартные библиотеки скриптовых языков для создания элементов web-страниц	Лабораторная работа (ПР-6)	Задание, тип 2
			Владеет технологией разработки web-сайтов для графического отображения и визуализации результатов научно-исследовательской деятельности	Контрольная работа (ПР-2)	
3	Основы теории баз данных	ПК-5	Знает основные технологии разработки web-приложений; основы HTML и CSS, скриптовых языков программирования; основы разработки web-сайтов и приложений	Собеседование (УО-1)	Вопросы 1-23
			Умеет создавать статические web-сайты для представления информации о разрабатываемых проектах, решаемых задачах	Лабораторная работа (ПР-6)	Задание, тип 3
			Владеет методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов	Контрольная работа (ПР-2)	
4	Программирование на стороне сервера	ПК-2	Знает возможности и структуру сети Internet, способы поиска информации; основы теории баз данных	Собеседование (УО-1)	Вопросы 1-22
			Умеет использовать информацию из сети Internet для решения стандартных задач профессиональной	Лабораторная работа (ПР-6)	Задание, тип 4

№ п/п	Контролируемые разделы / темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства	
			текущий контроль	промежуточная аттестация
		деятельности;		
		Владеет методикой применения баз данных и Интернет-технологии для решения задач профессиональной деятельности.	Расчетно-графическая работа (ПР-12)	

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в Приложении 2.

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

1. Бурлакова, Н.Н. Создание гипертекстовой среды : учебное пособие для вузов ч. 1 . Статическое кодирование / Н. Н. Бурлакова ; Дальневосточный федеральный университет. -Владивосток : Изд-во Дальневосточного федерального университета, 2012, 85 с.

<http://lib.dvfu.ru:8080/lib/item?id=chamo:681561&theme=FEFU>

2. Волошина В. Н. Организация баз данных : учебное пособие для вузов ч. 2 / В. Н. Волошина, С. И. Гордеев ; Дальневосточный федеральный университет.- Владивосток : Изд-во Дальневосточного федерального университета, 2011, 503 с.

<http://lib.dvfu.ru:8080/lib/item?id=chamo:425983&theme=FEFU>

3. Богданова, С.В. Информационные технологии [Электронный ресурс] : учебное пособие / С.В. Богданова, А.Н. Ермакова. - Ставрополь: Сервисшкола, 2014. - 211 с. <http://znanium.com/catalog/product/514867>

4. Стасышин В.М. Проектирование информационных систем и баз данных/СтасышинВ.М. - Новоси́б.: НГТУ, 2012. - 100 с. <http://znanium.com/catalog/product/548234>

5. Сухов, К. HTML5 – путеводитель по технологии [Электронный ресурс] / К. Сухов. — Электрон. дан. — Москва : ДМК Пресс, 2013. — 352 с. <https://e.lanbook.com/book/40002>

6. Ушаков, Д.М. Введение в математические основы САПР: курс лекций [Электронный ресурс] : учебное пособие / Д.М. Ушаков. — Электрон. дан. — Москва : ДМК Пресс, 2011. — 208 с. <https://e.lanbook.com/book/1311>

Дополнительная литература

1. Зудилова, Т. В. Web-программирование HTML [Электронный ресурс] / Т. В. Зудилова, М. Л. Буркова. — Электрон. текстовые данные. — СПб. : Университет ИТМО, 2012. — 70 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/65748.html>

2. Сосновиков, Г. К. Основы реляционных баз данных [Электронный ресурс] : учебное пособие / Г. К. Сосновиков, В. Н. Шакин. — Электрон. текстовые данные. — М. : Московский технический университет связи и информатики, 2013. — 106 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/61516.html>

3. Кисленко, Н. П. Интернет-программирование на PHP [Электронный ресурс] : учебное пособие / Н. П. Кисленко. — Электрон. текстовые данные. — Новосибирск : Новосибирский государственный архитектурно-строительный университет (Сибстрин), ЭБС АСВ, 2015. — 177 с. — 978-5-7795-0745-5. — Режим доступа: <http://www.iprbookshop.ru/68769.html>

**Перечень ресурсов информационно-телекоммуникационной сети
«Интернет»**

1. Справочник Notepad++ - свободно распространяемый текстовый редактор с открытым исходным кодом – Режим доступа: <https://notepad-plus-plus.org/>;
2. Справочник по HTML. – Режим доступа: <https://htmlbook.ru/html/>;
3. CSS справочник. – Режим доступа: <https://css.manual.ru/>;
4. Справочник с примерами по языку SQL. – Режим доступа: sql.itsoft.ru
5. Справочник Javascript. – Режим доступа: <https://Javascript.ru>.

Перечень информационных технологий и программного обеспечения

Необходимое программное обеспечение:

- операционная система Windows;
- пакет Microsoft Office;
- свободно распространяемый текстовый редактор с открытым исходным кодом Notepad++ (<https://notepad-plus-plus.org/>);
- свободно распространяемый web-браузер Google Chrome (<https://www.google.ru/chrome/browser/desktop/>).

Материалы курса размещены в LMS BlackBoard, идентификатор: FU50219-151600.62-IWT-01.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

1. Описание последовательности действий студента («сценарий изучения дисциплины»).

Для успешного изучения дисциплины «Инженерные web-технологии» студенту необходимо:

1. При подготовке к лекции изучить конспект предыдущей лекции, а также при необходимости теоретический материал, представленный в системе BlackBoard. Далее ознакомиться с материалом следующей лекции (также через систему BlackBoard), вникнуть в суть изучаемой проблемы, подготовить вопросы.

2. На лекционном занятии тщательно конспектировать теоретический материал, участвовать в обсуждении, задавать вопросы.

3. При подготовке к лабораторным занятиям на основе материалов, представленных в системе BlackBoard, сначала понять задание лабораторной работы, найти теоретический материал, необходимый для работы, изучить алгоритм реализации задания, сформулировать вопросы преподавателю.

4. На лабораторном занятии сначала задать вопросы преподавателю по методике выполнения работы, затем выполнить задание «по образцу». Отправить преподавателю через систему BlackBoard выполненное задание на проверку, ответить на вопросы преподавателя. Прежде, чем приступать к выполнению самостоятельных заданий, обдумать алгоритм их реализации, задать вопросы преподавателю по сути заданий, спланировать их выполнение.

5. Самостоятельные задания лабораторной работы можно выполнять как на аудиторном занятии, так и самостоятельно во внеаудиторное время. При этом результат их реализации необходимо проверять через систему BlackBoard и только после этого отправлять преподавателю на проверку.

6. В течение недели выбрать время для работы со специальной литературой в библиотеке и для занятий на компьютере.

7. Самостоятельную работу организовывать в соответствии с графиком выполнения самостоятельной работы, приведенном в приложении 1.

2. Рекомендации по использованию материалов учебно-методического комплекса. Рекомендуется использовать методические указания и материалы по курсу «Инженерные web-технологии», размещенные в системе BlackBoard, идентификатор курса FU50219-151600.62-IWT-01.

3. Рекомендации по работе с литературой. Теоретический материал курса становится более понятным, когда дополнительно к прослушиванию лекций, использованию учебно-методического комплекса, представленного в системе BlackBoard, изучаются и книги из списка основной и

дополнительной литературы. Литературу по курсу можно изучать в библиотеке, брать книгу на дом или читать ее на компьютере (если это электронный ресурс). Полезно использовать несколько учебников, однако желательно придерживаться рекомендации преподавателя по выбору книг по каждому разделу. Не рекомендуется «заучивать» материал, желательно добиться понимания изучаемой темы дисциплины, а затем использовать изученный материал для реализации программ и/или сайтов. Кроме того, очень полезно выявить тенденции развития той или иной технологии разработки, выделить для себя направления дальнейшего изучения материала, для достижения более продвинутого уровня изучения дисциплины.

4. Рекомендации по подготовке к экзамену и зачету. Успешная подготовка к экзамену и зачету включает, с одной стороны, добросовестную работу в течение семестра, выполнение всех заданий преподавателя, а с другой – правильная организация процесса непосредственной подготовки. При подготовке к экзамену необходимо освоить теорию: разобрать определения всех понятий, повторить синтаксис и семантику изучаемых языков программирования, технологию разработки web-сайтов. Затем рассмотреть примеры и самостоятельно реализовать задания из каждой темы. При этом, если задания формулируются студентом самостоятельно, – достигается более продвинутый уровень изучения дисциплины.

VI. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Аудиторные занятия по дисциплине включают лекции и лабораторные работы. Для проведения лекционных занятий необходима аудитория со следующим оборудованием:

Акустическая система для потолочного монтажа с низким профилем,
Extron SI 3CT LP (пара)

Врезной интерфейс с системой автоматического втягивания кабелей TLS
TAM 201 Standart III

Документ-камера Avervision CP355AF

Комплект удлинителей DVI по витой паре (передатчик/приёмник),
Extron DVI 201 Tx/Rx

Матричный коммутатор DVI 4x4. Extron DXP 44 DVI PRO

Микрофонная петличная радиосистема УВЧ диапазона Sennheiser EW 122 G3 в составе речевого приёмника EM 100 G3, передатчика SK 100 G3, петличного микрофон ME 4 с ветрозащитой и антенн (2 шт.)

Мультимедийный проектор, Mitsubishi EW330U, 3000 ANSI Lumen, 1280x800

Расширение для контроллера управления Extron IPL T CR48

Сетевая видеочамера Multipix MP-HD718

Сетевой контроллер управления Extron IPL T S4

Усилитель мощности, Extron XPA 2001-100V

Цифровой аудиопроцессор, Extron DMP 44 LC

Шкаф настенный 19" 7U, Abacom VSP-W960SG60

Экран проекционный ScreenLine Trim White Ice, 50 см черная кайма сверху, размер рабочей области 236x147 см

Лабораторные работы проводятся в компьютерном классе, в котором должно быть установлено:

- Моноблок Lenovo C306G-i34164G500UDK (20 шт),
- Акустическая система для потолочного монтажа с низким профилем, Extron SI 3CT LP (пара)
- Акустическая система для потолочного монтажа с низким профилем, Extron SI 3CT LP (пара)
- Акустическая система для потолочного монтажа с низким профилем, Extron SI 3CT LP (пара)
- Врезной интерфейс с системой автоматического втягивания кабелей TLS TAM 201 Standart III
- Документ-камера Avervision CP355AF

- ЖК-панель 47", Full HD, LG M4716CCBA
- Комплект удлинителей DVI по витой паре (передатчик/приёмник), Extron DVI 201 Tx/Rx
- Матричный коммутатор DVI 4x4. Extron DXP 44 DVI PRO
- Микрофонная петличная радиосистема УВЧ диапазона Sennheiser EW 122 G3 в составе рэкового приёмника EM 100 G3, передатчика SK 100 G3, петличного микрофон ME 4 с ветрозащитой и антенн (2 шт.)
- Мультимедийный проектор, Mitsubishi EW330U, 3000 ANSI Lumen, 1280x800
- Расширение для контроллера управления Extron IPL T CR48
- Сетевая видеочамера Multipix MP-HD718
- Сетевой контроллер управления Extron IPL T S4
- Стойка металлическая для ЖК-дисплея У SMS Flatscreen FH T1450
- Усилитель мощности, Extron XPA 2001-100V
- Цифровой аудиопроцессор, Extron DMP 44 LC
- Шкаф настенный 19" 7U, Abacom VSP-W960SG60
- Экран проекционный ScreenLine Trim White Ice, 50 см черная кайма сверху, размер рабочей области 236x147 см



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДФУ)

ИНЖЕНЕРНАЯ ШКОЛА

**УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ
РАБОТЫ ОБУЧАЮЩИХСЯ**

по дисциплине «ИНЖЕНЕРНЫЕ WEB-ТЕХНОЛОГИИ»

Направление подготовки: 15.03.03 Прикладная механика

Профиль подготовки:

**«Математическое и компьютерное моделирование механических систем и
процессов»**

Форма подготовки очная

Владивосток

2016

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
1	1 - 2 недели семестра	Лабораторная работа «Основы HTML», самостоятельные задания	5 час.	ПР-6
2	3-4 недели семестра	Лабораторная работа «Использование DHTML для разработки сайта», самостоятельные задания	5 час.	ПР-6
3	5-6 недели семестра	Лабораторная работа «Основы CSS», самостоятельные задания	5 час.	ПР-6
4	7 -8 недели семестра	Подготовка к тесту по разделу «Инструменты создания сайтов» Лабораторная работа «Создание адаптивного сайта на основе технологии CSS», самостоятельные задания	5 час.	ПР-1 ПР-6
5	9 -10 недели семестра	Лабораторная работа «Использование языка JavaScript для программирования приложений на стороне клиента», самостоятельные задания	5 час.	ПР-6
6	11-12 недели семестра	Лабораторная работа «Создание динамического сайта средствами JavaScript», самостоятельные задания	5 час.	ПР-6
7	13 -14 недели семестра	Лабораторная работа «Создание сайта с применением JavaScript и модели DOM», самостоятельные задания	5 час.	ПР-6
8	15-16 недели семестра	Лабораторная работа «Создание сайта с использованием библиотеки JQUERY», самостоятельные задания	5 час.	ПР-6
9	17-18 недели семестра	Лабораторная работа «Программное рисование, использование CANVAS HTML5», самостоятельные задания Подготовка к зачету	5 час.	ПР-6 зачет
10	1 - 2 недели	Подготовка к опросу по	5 час.	УО-1

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
	семестра	разделу «Программирование на стороне клиента» Лабораторная работа «Программное рисование и библиотека Mootools», самостоятельные задания		ПР-6
11	3-4 недели семестра	Лабораторная работа «Построение информационно-логической структуры базы данных, создание и наполнение таблиц для заданной предметной области», самостоятельные задания	5 час.	
12	5-6 недели семестра	Лабораторная работа «Реализация запросов на выборку для различных предметных областей», самостоятельные задания	5 час.	ПР-12
13	7 -8 недели семестра	Подготовка к устному опросу по разделу «Основы теории баз данных» Лабораторная работа «Корректировка данных средствами запросов для различных предметных областей	5 час.	УО-1 ПР-12
14	9 -10 недели семестра	Лабораторная работа 9. «Основы PHP», самостоятельные задания	5 час.	ПР-12
15	11-12 недели семестра	Лабораторная работа «Организация доступа к базам данных, PHP и MySQL», самостоятельные задания	5 час.	ПР-12
16	13 -14 недели семестра	Лабораторная работа «Создание динамической страницы на стороне сервера», самостоятельные задания	5 час.	ПР-12
17	15-16 недели семестра	Подготовка к устному опросу по разделу «Программирование на стороне сервера» Лабораторная работа «Создания страницы для отображения информации из базы данных», самостоятельные задания	5 час.	УО-1 ПР-12

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
18	17-18 недели семестра	Расчетно-графическая работа «Разработка клиент-серверного приложения»	5 час.	ПР-12
19		Подготовка к экзамену	36 час.	Экзамен
Итого			126 час.	

Характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению

Расчетно-графическая работа

В расчетно-графической работе требуется создать web-приложение, используя стандартные инструменты разработки web-сайтов на стороне сервера и клиента. Приложение должно быть предназначено для организации работы с пользователем и решения некоторых задач выбранной предметной области. Результат работы – работающий на локальном сервере web-сайт и документация по его разработке.

Основные этапы разработки web-приложения

1. Разработка технического задания. Составление документации, детально описывающей структурную и поведенческую модели проекта. Простым языком: описываем коротко, на техническом языке, как будет выглядеть и как будет работать сайт или программа.

2. Создание дизайна и навигации. Детальная прорисовка и определение позиционирования, типовых структурных элементов проекта. Например: создание эскиза страницы сайта, разработка логотипа, выбор мест расположения элементов навигации, оформление шрифтов и заголовков и многое другое.

3. Вёрстка. Описание типовых структурных элементов проекта на языке разметки, в соответствии с заданной спецификацией. Проще говоря: то, что разработано на 2-м этапе, необходимо привести к рабочему виду, используя разметки HTML и CSS.

4. Программирование. Описание структурной и поведенческой моделей проекта на языках программирования.

5. Тестирование и отладка. Выявление синтаксических и логических ошибок проекта и последующее их устранение.

Устные опросы

Устные опросы проводятся преподавателем по завершению изучения каждого раздела. Вопросы и задания приведены в приложении 2. Для подготовки используется основная и дополнительная литература по дисциплине «Инженерные web-технологии», а также информация, размещенная в LMS BlackBoard.

Вопросы, возникающие в процессе подготовки, студент может задать преподавателю либо на консультациях, либо через специальное средство LMS BlackBoard.

Тестирование

Тестирование осуществляется на занятии через систему BlackBoard. Для подготовки тестов используются пробные тесты, размещенные в системе BlackBoard. Эти тесты не оцениваются преподавателем и служат элементом самоконтроля. Оба типа теста формируются на основе одной базы вопросов.

Лабораторные работы

Выполняются в соответствии с методическими указаниями к выполнению лабораторных работ, размещенных в системе BlackBoard. Выполненные работы отправляются преподавателю на проверку через LMS BlackBoard.

Требования к представлению и оформлению результатов самостоятельной работы

Результатом самостоятельной работы студентов являются:

1. Расчетно-графическая работа – web-сайт, реализованный стандартными средствами разработки web-приложений и оформленный в соответствии со стандартами. Созданные сайты обязательно отсылаются преподавателю через систему BlackBoard. Защита расчетно-графического

задания осуществляется только после того, как приложение проверено и оценено преподавателем в системе BlackBoard.

2. Тесты самоконтроля, выполненные в системе BlackBoard. Все попытки, выполненные пользователем, сохраняются. К основному тесту студент допускается только после прохождения на зачетном уровне данного вида тестов.

3. Самостоятельные задания к лабораторным работам, реализованные необходимыми средствами разработки web-приложений и оформленные в соответствии со стандартами. Реализованные задания обязательно отсылаются преподавателю через систему BlackBoard. Защита лабораторной работы осуществляется только после того, как задание проверено и оценено преподавателем в системе BlackBoard.

Правила оформления HTML/CSS кода

1. Протокол.

Не указывайте протокол при включении ресурсов на страницу. Опускайте название протокола (**http:**, **https:**) в ссылках на картинки или другие медиа-ресурсы, файлы стилей или скрипты, конечно, если эти файлы доступны по обоим протоколам. Отсутствие протокола делает ссылку относительной, что предотвращает смешивание ресурсов из разных протоколов и незначительно уменьшает размер файлов.

Не рекомендуется:

```
<script src="http://www.google.com/js/gweb/analytics/autotrack.js">
</script>
```

Рекомендуется:

```
<script src="//www.google.com/js/gweb/analytics/autotrack.js"></script>
```

2. Общее форматирование

2.1. Отступы

Всегда используйте для отступа два пробела. Не используйте табуляцию и не смешивайте табуляцию с пробелами.

Рекомендуется:

```
<ul>
  <li>
```

```

    <li>
</ul>

.example {
  color: blue;
}

```

2.2. Регистр.

Всегда пишите в нижнем регистре. Весь код должен быть написан в нижнем регистре: Это относится к названиям элементов, названиям атрибутов, значениям атрибутов (кроме текста/CDATA), селекторам, свойствам и их значениям (кроме текста).

Не рекомендуется:

```
<A HREF="/">Домой </A>
```

Рекомендуется:

```

```

2.3. Пробелы в конце строки

Убирайте пробелы в конце строки. Пробелы в конце строк не обязательны и усложняют использование diff.

3. Общие мета правила

3.1. Кодировка

Используйте UTF-8 (без BOM). Убедитесь, что ваш редактор использует кодировку UTF-8 без метки порядка байтов (BOM). Указывайте кодировку в HTML шаблонах и документах с помощью

```
<meta charset="utf-8">.
```

Опускайте кодировку для css-файлов: для них UTF-8 задана по умолчанию.

3.2. Комментарии

По возможности поясняйте свой код, где это необходимо. Используйте комментарии, чтобы пояснить свой код: что он делает, за что отвечает, и почему используется выбранное решение.

4. Правила оформления HTML

4.1. Тип документа

Используйте HTML5. HTML5 (HTML синтаксис) рекомендуется для всех html-документов: `<!DOCTYPE html>`.

4.2. Валидность HTML

По возможности используйте валидный HTML. Используйте валидный HTML код, кроме случаев, когда использование не позволяет достичь размера файла, необходимого для нужного уровня производительности. Используйте такие инструменты как [W3C HTML validator \(англ.\)](#) чтобы проверить валидность кода.

4.3. Семантика

Используйте HTML так, как это было задумано. Используйте элементы (Иногда неверно называемые “тегами”) по назначению: заголовки для заголовков, **p** для абзацев, **a** для ссылок и т.д. Это облегчает чтение, редактирование и поддержку кода.

Не рекомендуется:

```
<div onclick="goToRecommendations();">All recommendations </div>
```

Рекомендуется:

```
<a href="recommendations/">All recommendations </a>
```

4.4. Альтернатива для мультимедиа

Всегда указывайте альтернативное содержимое для мультимедиа. Постарайтесь указать альтернативное содержимое для мультимедиа: например для картинок, видео или анимаций, заданных с помощью **canvas**. Для картинок это осмысленный альтернативный текст (**alt**), а для видео и аудио расшифровка текста и подписи если это возможно.

Не рекомендуется:

```

```

Рекомендуется:

```

```

4.5. Разделение ответственности

Разделяйте структуру, оформление и поведение. Держите структуру (разметка), оформление (стили) и поведение (скрипты) отдельно и постарайтесь свести взаимодействие между ними к минимуму. Убедитесь,

что документы и шаблоны содержат только HTML, и что HTML служит только для задания структуры документа. Весь код, отвечающий за оформление, перенесите в файлы стилей, а код отвечающий за поведение — в скрипты. Старайтесь сократить их пересечения к минимуму, включая в шаблоны минимальное количество файлов стилей и скриптов.

Отделение структуры от представления и поведения помогает облегчить поддержку кода. Изменение шаблонов и HTML-документов всегда занимает больше времени чем изменение файлов стилей или скриптов.

4.6. Атрибут *type*

Не указывайте атрибут type при подключении стилей и скриптов в документ. Не используйте атрибут **type** при подключении стилей (кроме вариантов когда используется что-то кроме CSS) и скриптов (кроме вариантов когда это не JavaScript). Указывать атрибут **type** в данном случае не обязательно потому что HTML5 использует `text/css` и `text/javascript` по умолчанию. Это будет работать даже в старых браузерах.

Не рекомендуется:

```
<link rel="stylesheet" href="//www.google.com/css/maia.css"
type="text/css">
```

Рекомендуется:

```
<link rel="stylesheet" href="//www.google.com/css/maia.css">
```

5. Правила форматирования HTML

5.1. Форматирование

Выделяйте новую строку для каждого блочного, табличного или списочного элемента и ставьте отступы для каждого дочернего элемента. Независимо от стилей заданных для элемента (CSS позволяет изменить поведение элемента с помощью свойства **display**), переносите каждый блочный или табличный элемент на новую строку.

Также ставьте отступы для всех элементов вложенных в блочный или табличный элемент.

6. Правила оформления CSS

6.1 Идентификаторы и названия классов

Используйте шаблонные или имеющие смысл имена классов и идентификаторы. Вместо использования шифров, или описания внешнего вида элемента, попробуйте в имени класса или идентификатора выразить смысл его создания, или дайте ему шаблонное имя...

Рекомендуется выбирать имена, отражающие сущность класса, потому что их проще понять и, скорее всего, не понадобится менять в будущем.

6.2. Названия идентификаторов и классов

Для идентификаторов и классов используйте настолько длинные имена, насколько нужно, но настолько короткие, насколько возможно. Попробуйте сформулировать, что именно должен делать данный элемент, при этом будьте кратки насколько возможно. Такое использование классов и идентификаторов вносит свой вклад в облегчение понимания и увеличение эффективности кода.

6.3. Селекторы типа

Избегайте использование имен классов или идентификаторов с селекторами типа (тега) элемента.

6.4. 0 и единицы измерения

Не указывайте единицы измерения для нулевых значений. Не указывайте единицы измерения для нулевых значений если на это нет причины.

6.5. 0 в целой части дроби

Не ставьте "0" в целой части дробных чисел. Не ставьте **0** в целой части в значениях между -1 и 1.

6.6. Кавычки в ссылках

Не используйте кавычки в ссылках. Не используйте кавычки (" ", ' ') с `url()`.

6.7. Шестнадцатеричные названия цветов

Используйте трехсимвольную шестнадцатеричную запись где это возможно. Трехсимвольная шестнадцатеричная запись для цветов короче и занимает меньше места.

Не рекомендуется:

```
color: #eebbcc;
```

Рекомендуется:

```
color:#ebc;
```

6.8. Разделители в классах и идентификаторах

Разделяйте слова в идентификаторах и именах классов с помощью дефиса или подчеркивания. Не используйте ничего, кроме дефиса или подчеркивания, для соединения слов и сокращений в селекторах, чтобы повысить удобство чтения и легкость понимания кода.

7. Правила форматирования CSS**7.1. Упорядочивание объявлений**

Сортируйте объявления по алфавиту. Задавайте объявления в алфавитном порядке, чтобы получить согласованный код, с которым легко работать. При сортировке игнорируйте браузерные префиксы. При этом, если для одного свойства используются несколько браузерных префиксов, они также должны быть отсортированы (например **-moz** должен быть перед **-webkit**)

7.2. Отступы в блоках.

Всегда ставьте отступы для содержимого блоков. Всегда ставьте отступы для любого блочного содержимого, Например для правил внутри правил или объявлений, чтобы отобразить иерархию и облегчить понимание кода.

7.3. После объявлений

Ставьте точку с запятой после каждого объявления. После каждого объявления ставьте точку с запятой для согласованности кода и облегчения добавления новых свойств.

7.4. После названий свойств

Используйте пробелы после двоеточий в объявлениях. Всегда используйте один пробел после двоеточия (но не до) в объявлениях, для порядка в коде.

7.5. Отделение селектора и объявления

Отделяйте селекторы и объявления переносом строки. Начинайте каждый селектор или объявление с новой строки.

Не рекомендуется:

```
a:focus, a:active {  
  position: relative; top: 1px;  
}
```

Рекомендуется:

```
h1,  
h2,  
h3 {  
  font-weight: normal;  
  line-height: 1.2;  
}
```

Критерии оценки выполнения самостоятельной работы

Самостоятельная работа студентов включает расчетно-графическую работу, подготовку к устным опросам, тестирование и лабораторные работы. Критерии оценки каждого вида работы приведены в приложении 2.



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДФУ)

ИНЖЕНЕРНАЯ ШКОЛА

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине **«ИНЖЕНЕРНЫЕ WEB-ТЕХНОЛОГИИ»**
Направление подготовки: 15.03.03 Прикладная механика
Профиль подготовки:
«Математическое и компьютерное моделирование механических систем и
процессов»
Форма подготовки очная

Владивосток
2016

Паспорт ФОС

Код и формулировка компетенции	Этапы формирования компетенции	
ПК-2 – способностью применять физико-математический аппарат, теоретические, расчетные и экспериментальные методы исследований, методы математического и компьютерного моделирования в процессе профессиональной деятельности	Знает	возможности и структуру сети Internet, способы поиска информации; основы теории баз данных
	Умеет	использовать информацию из сети Internet для решения стандартных задач профессиональной деятельности;
	Владеет	методикой применения баз данных и Интернет-технологии для решения задач профессиональной деятельности.
ПК-5 – способностью составлять описания выполненных научно-исследовательских работ и разрабатываемых проектов, обрабатывать и анализировать полученные результаты, готовить данные для составления отчетов и презентаций, написания докладов, статей и другой научно-технической документации	Знает	основные технологии разработки web-приложений; основы HTML и CSS, скриптовых языков программирования; основы разработки web-сайтов и приложений
	Умеет	создавать статические web-сайты для представления информации о разрабатываемых проектах, решаемых задачах
	Владеет	методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов
ПК-6 – способностью применять программные средства компьютерной графики и визуализации результатов научно-исследовательской деятельности, оформлять отчеты и презентации, готовить рефераты, доклады и статьи с помощью современных офисных информационных технологий, текстовых и графических редакторов, средств печати	Знает	основные средства компьютерной графики, применяемые в web-приложениях
	Умеет	использовать стандартные библиотеки скриптовых языков для создания элементов web-страниц
	Владеет	технологией разработки web-сайтов для графического отображения и визуализации результатов научно-исследовательской деятельности

№	Контролиру	Коды и этапы формирования	Оценочные средства
---	------------	---------------------------	--------------------

п/п	емые разделы / темы дисциплин ы	компетенций		текущий контроль	промеж уточная аттеста ция
1	Инструменты создания сайтов	ПК-5	Знает основные технологии разработки web-приложений; основы HTML и CSS, скриптовых языков программирования; основы разработки web-сайтов и приложений	Тест (ПР-1)	Вопросы 1-24
			Умеет создавать статические web-сайты для представления информации о разрабатываемых проектах, решаемых задачах	Лабораторная работа (ПР-6)	Задание, тип 1
			Владеет методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов	Контрольная работа (ПР-2)	
2	Программирование на стороне клиента	ПК-6	Знает основные средства компьютерной графики, применяемые в web-приложениях	Собеседование (УО-1)	Вопросы 1-22
			Умеет использовать стандартные библиотеки скриптовых языков для создания элементов web-страниц	Лабораторная работа (ПР-6)	Задание, тип 2
			Владеет технологией разработки web-сайтов для графического отображения и визуализации результатов научно-исследовательской деятельности	Контрольная работа (ПР-2)	
3	Основы теории баз данных	ПК-5	Знает основные технологии разработки web-приложений; основы HTML и CSS, скриптовых языков программирования; основы разработки web-сайтов и приложений	Собеседование (УО-1)	Вопросы 1-23
			Умеет создавать статические web-сайты для представления	Лабораторная работа (ПР-6)	Задание, тип 3

№ п/п	Контролируемые разделы / темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства		
			текущий контроль	промежуточная аттестация	
		информации о разрабатываемых проектах, решаемых задачах			
		Владеет методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов	Контрольная работа (ПР-2)		
4	Программирование на стороне сервера	ПК-2	Знает возможности и структуру сети Internet, способы поиска информации; основы теории баз данных	Собеседование (УО-1)	Вопросы 1-22
			Умеет использовать информацию из сети Internet для решения стандартных задач профессиональной деятельности;	Лабораторная работа (ПР-6)	Задание, тип 4
			Владеет методикой применения баз данных и Интернет-технологии для решения задач профессиональной деятельности.	Расчетно-графическая работа (ПР-12)	

Шкала оценивания уровня сформированности компетенций

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
ПК-2 – способностью применять физико-математический аппарат, теоретические, расчетные и экспериментальные методы исследований, методы математического и компьютерного моделирования в процессе профессиональной	Знает	возможности и структуру сети Internet, способы поиска информации; основы теории баз данных	- знание определений основных понятий сети Интернет, знание ее структуры и возможностей; - знание основных приемов поиска информации в сети Интернет; - знание определений основных понятий теории реляционных баз данных	- способность сформулировать определения основных понятий сети Интернет; - способность перечислить и подробно описать возможности сети Интернет, ее структуру; - способность сформулировать и подробно описать правила и приемы поиска информации в

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
деятельности				сети Интернет; - способность дать определения основных понятий теории реляционных баз данных
	Умеет	использовать информацию из сети Internet для решения стандартных задач профессиональной деятельности;	- умение сформулировать критерии поиска информации, используемой для решения стандартных задач профессиональной деятельности; - умение отобразить, проанализировать и применить найденную информацию при решении стандартных задач профессиональной деятельности	- способность сформулировать критерии поиска информации, используемой для решения стандартных задач профессиональной деятельности; - способность отобразить, проанализировать и применить найденную информацию при решении стандартных задач профессиональной деятельности
	Владеет	методикой применения баз данных и Интернет-технологии для решения задач профессиональной деятельности.	- владение принципами проектирования баз данных для задач решения профессиональной деятельности; - владение инструментальными средствами Интернет-технологий для решения задач профессиональной деятельности	- способность спроектировать базы данных для решения задач профессиональной деятельности; - способность использовать инструментальные средства Интернет-технологий для решения задач профессиональной деятельности
ПК-5 – способностью составлять описания выполненных научно-исследовательских работ и разрабатываемых проектов, обрабатывать и анализировать полученные результаты, готовить данные для составления отчетов и презентаций, написания докладов, статей и другой научно-технической документации	Знает	основные технологии разработки web-приложений; основы HTML и CSS, скриптовых языков программирования; основы разработки web-сайтов и приложений	- знание определений основных понятий технологии разработки web-приложений; - знание синтаксиса и семантики HTML и CSS, скриптовых языков программирования; - знание основных этапов разработки web-сайтов и приложений и их содержания	- способность дать определения основных понятий технологии разработки web-сайтов и приложений; - способность перечислить основные конструкции HTML и CSS, скриптовых языков программирования; - способность перечислить и раскрыть содержание основных этапов разработки web-сайтов и приложений
	Умеет	создавать статические web-сайты для представления информации о разрабатываемых проектах, решаемых задачах	умение создавать статические web-сайты (HTML и CSS) для представления информации о разрабатываемых проектах, решаемых задачах	способность создавать статические web-сайты (HTML и CSS) для представления информации о разрабатываемых проектах, решаемых задачах

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
	Владеет	методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов	владение методикой разработки web-сайтов по технологии клиент-сервер средствами стандартного набора web-инструментов	способность разрабатывать web-сайты по технологии клиент-сервер средствами стандартного набора web-инструментов
ПК-6 – способностью применять программные средства компьютерной графики и визуализации результатов научно-исследовательской деятельности, оформлять отчеты и презентации, готовить рефераты, доклады и статьи с помощью современных офисных информационных технологий, текстовых и графических редакторов, средств печати	Знает	основные средства компьютерной графики, применяемые в web-приложениях	знание основных понятий растровой компьютерной графики, применяемой в web-приложениях	- способность сформулировать и объяснить основные понятия растровой графики; - способность объяснить основные положения технологии растровой графики, применяемой в web-приложениях
	Умеет	использовать стандартные библиотеки скриптовых языков для создания элементов web-страниц	умение использовать стандартные библиотеки скриптовых языков для создания элементов web-страниц	способность использовать стандартные библиотеки скриптовых языков для создания элементов web-страниц
	Владеет	технологией разработки web-сайтов для графического отображения и визуализации результатов научно-исследовательской деятельности	владение технологией разработки web-сайтов для графического отображения и визуализации результатов научно-исследовательской деятельности	Способность разработать web-сайты для графического отображения и визуализации результатов научно-исследовательской деятельности

Зачетно-экзаменационные материалы

Перечень типовых вопросов к зачету, 6 семестр

1. Введение в Web- технологии, основные понятия и определения (URL, Inter- net, WWW, HTTP, FTP, ...)
2. Доменные имена, IP - адрес, DNS, Web- сайты.
3. Понятие технологии клиент- сервер, Web-сервер, сервер БД, почтовый сервер, файловый сервер.
4. Характеристика программного обеспечения, используемого при создании Web -страниц.

5. Введение в язык HTML, элементы языка HTML.
6. HTML, простейшее форматирование текста, управление выравниванием текста.
7. HTML, оформление абзацев, заголовков, горизонтальная линейка.
8. HTML, управление шрифтом.
9. HTML, управление цветом, дополнительные варианты оформления.
10. Локальные гиперссылки в рамках Web-страницы, формирование гиперссылок в пределах сайта, якоря.
11. HTML, маркированные списки, смешанные списки.
12. HTML, нумерованные списки, смешанные списки.
13. Гиперссылки в пределах сайта, организация переходов средствами гиперссылок.
14. Работа с таблицами в языке HTML.
15. Управление шириной столбцов, высотой строк, объединение ячеек в таблицах средствами тегов языка HTML.
16. Формирование фоновых изображений на Web-страницах.
17. Графика на Web-страницах, вставка изображений.
18. CSS. Встраивание таблиц стилей в документ. Свойства CSS для шрифтов и текста. Свойства CSS для полей, отступов и границ. Свойства CSS для фона и цвета.
19. CSS. Каскадные таблицы стилей (в отдельных файлах). Визуальные фильтры, производимый эффект и параметры.
20. CSS. Каскадные таблицы стилей: стили и селекторы.
21. CSS. Позиционирование: свободное, абсолютное, фиксированное и относительное размещение.
22. CSS. Селекторы: селекторы классов и идентификаторов, селекторы атрибутов и псевдоселекторы.
23. CSS. Форматирование страниц с помощью языка иерархических стилевых спецификаций, использование каскадных таблиц стилей.

24. CSS. Цвет. Фоновые изображения. Объединение свойств фона. JavaScript. доступ к значениям элементов форм. Объект frames и динамическое создание фрейма.

25. JavaScript. объект Array. Обобщенные объекты.

26. JavaScript. объект images и его свойства. Динамическое изменение изображений, создание простейших анимаций, подгрузка рисунков.

27. JavaScript. объект Math, его свойства и методы. Объект String, его свойства и методы.

28. JavaScript. свойства объекта document.body, метод document.write() и объект history. Объект window и параметры метода window.open().

29. Характеристика библиотеки JQuery

30. Ссылки на элементы интернет-страницы.

31. Установка свойств элементов и стилей.

Перечень типовых вопросов к зачету, 8 семестр

1. Характеристика библиотеки Mootools
2. Реализация основных концепций объектно-ориентированного программирования средствами Mootools
3. Элемент canvas. Его свойство и характеристики
4. Рисование 2d изображений.
5. Преобразование изображений.
6. Использование таймера.
7. Серверный язык PHP. Синтаксис. Включение PHP-сценария в HTML-документ.
8. Серверный язык PHP. Организация ветвлений.
9. Серверный язык PHP. Понятие класса. Основные компоненты класса.
10. Серверный язык PHP. Абстрагирование, инкапсуляция, модульность и иерархия

11. Серверный язык PHP. Хранение и использование данных пользователя. Способы хранения. Хранение данных в файлах.

12. Серверный язык PHP. Хранение данных в файлах. Открытие файла. Функция `fopen()`. Режимы файла. Чтение файла. Запись в файл.

13. Серверный язык PHP. Организация счетчика посещений.

14. Серверный язык PHP. Обработка форм.

15. Серверный язык PHP. Массивы, наследование.

16. Базы данных MySQL. Базовая архитектура баз данных для web. Транзакция базы данных для web.

17. Базы данных MySQL. Доступ к базе данных из web. Выбор базы данных.

18. Базы данных MySQL. Доступ к базе данных из web. Структура процедуры доступа. Установка соединения.

19. Базы данных MySQL. Доступ к базе данных из web. Фильтрация входных данных.

20. Базы данных MySQL. Доступ к базе данных из web. Выполнение запроса к базе данных.

21. Базы данных MySQL. Доступ к базе данных из web. Получение результатов запроса.

22. Базы данных MySQL. Доступ к базе данных из web. Организация поиска в базе данных.

Перечень типовых экзаменационных вопросов

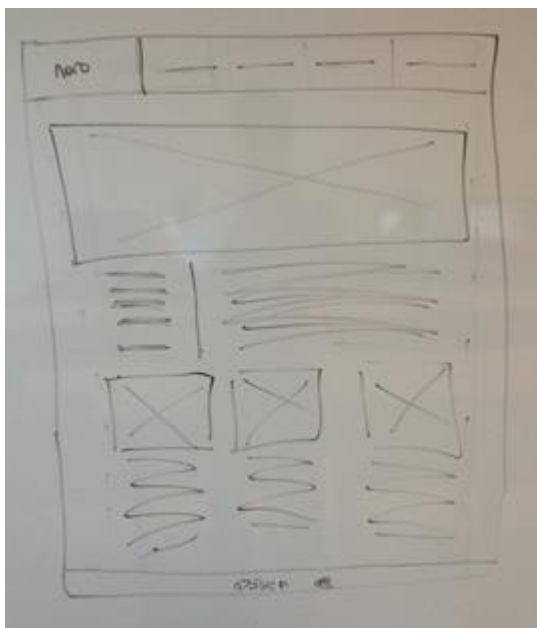
1. Определение и классификация информационных систем.
2. Требования, предъявляемые к базам данных
3. Принципы построения баз данных.
4. Логическая организация базы данных.
5. Модели данных.
6. Основные понятия реляционных баз данных: отношение, домен, кортеж, атрибут.
7. Понятия нормализованных отношений. Правила нормализации

8. Информационные объекты.
9. Способы выделения информационных объектов предметной области.
10. Типы связей информационных объектов.
11. Логическая структура реляционной базы данных.
12. Язык структурированных запросов для реляционной модели (SQL).
13. Бинарные операции над отношениями
14. Назначение и виды запросов.
15. Структура запроса
16. Запросы на выборку.
17. Запросы на выборку и групповые функции.
18. Вложенные запросы на выборку.
19. Корректировка данных средствами запросов.
20. Запросы на обновление.
21. Запросы на добавление
22. Запросы на создание таблицы
23. Запросы на удаление.

Перечень типовых экзаменационных задач

Задание тип 1, по разделу «Инструменты создания сайтов»

Создать сайт, используя HTML и CSS следующей структуры:



Задание тип 2, по разделу «Программирование на стороне клиента»

На языке JavaScript, используя DOM модель и библиотеку JQuery реализовать галерею, в которой реализована смена картинки через заданный интервал времени, выбор и загрузка картинки в отдельный блок.

Задание тип 3, по разделу «Основы теории баз данных»

Преобразовать таблицу СОТРУДНИК (разбить ее на несколько таблиц, включить новые столбцы) так, чтобы она отвечала правилам нормализации:

СОТРУДНИК

<i>Кафедра</i>	<i>Фамилия И.О.</i>	<i>Должность</i>	<i>Оклад</i>
Высшей математики	Борисов А.А.	ст. преп.	1000 р.
Информатики	Федоров Б.П.	доцент	1500 р.
Высшей математики	Сидоров К.Д.	ст. преп.	1000 р.
Прикладной математики	Назаров С.В.	ассистент	500 р.
Информатики	Петров В.А.	ассистент	500 р.

Задание тип 4, по разделу «Программирование на стороне сервера»

Разработать базу данных, используя MySQL, для предметной области «Библиотека»:

«В некоторой библиотеке хранятся книги. Каждая книга имеет название, относится к одному жанру, имеет одного или нескольких авторов, издана в определенном издательстве. В библиотеке может быть несколько экземпляров одной книги. Каждый экземпляр имеет уникальный шифр.

Пользователями библиотеки являются читатели, о которых сохраняется информация об их фамилии, адресе и телефоне. Каждый читатель может взять несколько экземпляров книг, в базе должны храниться сведения о взятой книге, дате получения и возврата».

Программно создать таблицы, занести данные и реализовать выборку информации по запросам пользователей.

Принцип составления экзаменационного билета заданий к зачету

Первый вопрос является теоретическим и предназначен для оценивания порогового уровня освоения дисциплины. Второй вопрос – представляет собой задание, которое необходимо выполнить на компьютере. Это может

быть создание небольшого сайта, решение некоторой задачи на языках JavaScript и PHP, проект структуры приложения пользователя.

Образец экзаменационного билета

1. Логическая структура реляционной базы данных.

2. На языке JavaScript, используя DOM модель и библиотеку JQuery реализовать галерею, в которой реализована смена картинки через заданный интервал времени, выбор и загрузка картинки в отдельный блок.

Критерии выставления оценки студенту на экзамене по дисциплине

«Инженерные web-технологии»

Баллы (рейтинговой оценки)	Оценка экзамена/заче та (стандартная)	Требования к сформированным компетенциям
86-100	«отлично»/ «зачтено»	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил материал по различным Интернет-технологиям, в том числе и технологиям разработки сайтов, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет проектировать web-приложения, используя инструменты web-разработки, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, свободно использует компьютер для реализации web-приложений, выбирает эффективную методологию при проектировании и реализации, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач, связанных с созданием приложений в области профессиональной деятельности.
76-85	«хорошо»/ «зачтено»	Оценка «хорошо» выставляется студенту, если он твердо знает материал по различным Интернет-технологиям в том числе и технологиям разработки сайтов, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет языки HTML, CSS, JavaScript, PHP при решении практических вопросов и задач, связанных с созданием web-приложений в области профессиональной деятельности, владеет необходимыми навыками и приемами использования инструментов разработки сайтов.
61-75	«удовлетворительно»/ «зачтено»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала по интернет-технологиям, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении синтаксических и семантических особенностей языков HTML, CSS, JavaScript, PHP, испытывает затруднения при проектировании и реализации web-приложения.

0-60	«неудовлетворительно»/ «незначительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала по основам интернет-технологий, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, связанные с созданием web-приложений. Оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине
------	---	--

.Оценочные средства для текущей аттестации

Вопросы для собеседований по дисциплине «Инженерные web-технологии»

Раздел «Программирование на стороне клиента»

1. JavaScript. Язык клиентских сценариев. Объект navigator, свойства браузера, объект document и свойства документа.
2. JavaScript. Арифметические, логические и операторы сравнения.
3. JavaScript. Ввод и вывод данных средствами JavaScript. Типы данных.
4. JavaScript. Динамический HTML: программирование окон и фреймов, свойства документа, работа с формами.
5. JavaScript. Операторы цикла и условного перехода.
6. JavaScript. Переменные и их область действия. Операторы языка JavaScript.
7. JavaScript. Символы-разделители и переводы строк. Комментарии. Литералы. Идентификаторы.
8. JavaScript. События мыши. Обработка нажатия и отпускания клавиши мыши.
9. JavaScript. События мыши. Обработка движения указателя мыши.
10. JavaScript. События клавиатуры. Обработка ввода символа.
11. JavaScript. События клавиатуры. Обработка нажатия/отпускания клавиши.

12. JavaScript. События, связанные с компонентами HTML-страницы.

13. JavaScript. Сценарии обработки событий. Объект EVENT и его атрибуты.

14. JavaScript. Числа и работа с числами. Логические значения в JavaScript.

15. JavaScript. гиперссылки (метки) и массив объектов links (anchors).
Объект Date и его методы.

16. JavaScript. Использование объектной модели документа.

17. JavaScript и библиотека JQuery.

18. JavaScript и библиотека Mootools.

19. Объекты в JavaScript. Иерархия объектов.

20. Новые элементы web-страниц в HTML5.

21. Структура страницы HTML 5.

22. Программное рисование

23. Особенности использования canvas

24. Трансформация изображений

25. Реализация движения

Раздел «Основы теории баз данных»

1. Основные требования к организации баз данных.

2. Назначение и основные компоненты системы баз данных.

3. Этапы проектирования баз данных.

4. Модели данных. Классификация моделей данных.

5. Модель «сущность-связь». Основные понятия. Область применения.

6. Иерархическая модель данных. Основные понятия. Область применения. Достоинства и недостатки.

7. Сетевая модель данных. Основные понятия. Область применения. Достоинства и недостатки.

8. Реляционная модель данных. Основные понятия. Область применения. Достоинства и недостатки.

9. Операции реляционной алгебры.

10. Структурированный язык запросов SQL. Категории SQL.
11. Структурированный язык запросов SQL. Описание данных. Таблицы. Типы данных. Целостность данных.
12. Структурированный язык запросов SQL. Операторы манипулирования данными. Курсор.
13. Структурированный язык запросов SQL. Типы связывания.
14. Структурированный язык запросов SQL. Многотабличные запросы.
15. Структурированный язык запросов SQL. Операции изменения и обновления базы данных.
16. Структурированный язык запросов SQL. Индексы.
17. Структурированный язык запросов SQL. Определение пользовательских представлений.
18. Структурированный язык запросов SQL. Использование UNION для объединения результатов инструкций SELECT.
19. Структурированный язык запросов SQL. Формирование запросов.

Раздел «Программирование на стороне сервера»

1. Web сервер и логика его работы.
2. Протокол HTTP. Структура запросов и ответов.
3. CGI. Способы передачи данных. Запоминание состояния.
4. HTML формы (описание и использование).
5. PHP. Особенности языка.
6. PHP. Основные конструкции программирования
7. PHP. Выражение
8. PHP. Простые типы данных
9. PHP. Описание и использование составных типов данных
10. PHP. Процедуры и функции.
11. PHP. Обработка входных данных.
12. PHP. Работа с GET и POST.
13. PHP. Доступ к базам данных.

14. СУБД MySQL. Характеристика и особенности.

15. Язык SQL. Создание и удаление баз данных и таблиц в MySQL.

16. Типы данных в MySQL.

17. Ключи, индексы, автоинкрементирование. Изменение структуры таблиц.

18. MySQL и PHP. Запросы на добавление, изменение и удаление данных.

19. MySQL и PHP. Запросы на извлечение данных. Объединения. Псевдонимы, группировка и упорядочивание.

Критерии оценки ответа на устных опросах:

✓ 100-85 баллов выставляется студенту, если его ответ показывает прочные знания основных положений изучаемого раздела дисциплины, отличается глубиной и полнотой раскрытия темы; студент показывает свободное владение терминологическим аппаратом; умение объяснять синтаксис и семантику языков HTML, CSS, JavaScript и PHP, делать выводы и обобщения, давать аргументированные ответы, приводить примеры интернет-технологий; обучающийся свободно владеет монологической речью, логичностью и последовательностью ответа; умеет алгоритмически описывать проблему и данные из выбранной предметной области.

✓ 85-76 баллов выставляется студенту, если его ответ, обнаруживает прочные знания основных положений изучаемого раздела дисциплины, отличается глубиной и полнотой раскрытия темы; обучающийся показывает владение терминологическим аппаратом; умение объяснять синтаксис и семантику языков HTML, CSS, JavaScript и PHP, делать выводы и обобщения, давать аргументированные ответы; студент демонстрирует свободное владение монологической речью, логичностью и последовательностью ответа; умение алгоритмически описывать проблему и данные из выбранной предметной области. Однако допускается одна - две неточности в ответе.

✓ 75-61 балл выставляется студенту, если его ответ, свидетельствует в основном о знании основных положений изучаемого раздела дисциплины, отличается недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками объяснения синтаксис и семантику языков HTML, CSS, JavaScript и PHP, недостаточным умением давать аргументированные ответы и применять интернет-технологии; студент недостаточно владеет монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; студент демонстрирует неумение алгоритмически описывать проблему данные из выбранной предметной области.

✓ 60-50 баллов выставляется студенту, если его ответ, обнаруживает незнание процессов основных положений изучаемого раздела дисциплины, отличается неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками использования языков HTML, CSS, JavaScript и PHP; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа; студент демонстрирует неумение алгоритмически описывать выбранную предметную область.

Тестовые задания по дисциплине «Инженерные web-технологии»

Каждому студенту формируется индивидуальный тест, в который входит 20 вопросов, выбранных случайным образом из списка заданий. В тесте реализованы следующие типы тестовых заданий:

- выбор одного варианта ответа;
- выбор нескольких вариантов ответа;
- ввод числового значения.

Типовые задания теста по разделу «Инструменты создания сайтов»

1. С помощью какого тега можно объявить версию языка HTML?
 - a. <head> и </head>
 - b. <html> и </html>

- c. `<!doctype>`
 d. `<body>` и `</body>`
2. Какие теги не имеют закрывающего?
- a. `<tr>`
 b. `<hr>`
 c. ``
 d. ``
 e. `<head>`
3. Какой размер в пунктах будет у текста
`<p>Пример текста</p>`,
 если на странице задан следующий стиль?
- ```
body { font-size: 24px; }
p { font-size: 50%; }
```

### **Критерии оценки теста, состоящего из 20 вопросов**

- ✓ 14-20 баллов – считается, что тест пройден.
- ✓ 0-13 баллов – тест не засчитывается

### **Комплекты заданий для выполнения расчетно-графической работы по дисциплине «Инженерные web-технологии»**

Расчетно-графическая работа предполагает реализацию web-приложения, используя стандартные инструменты web-разработки. Для создания приложения должна быть выбрана некоторая предметная область. При разработке необходимо выполнить следующие этапы: разработка технического задания, создание дизайна и навигации, верстка, программирование, отладка и тестирование.

#### **Типовое задание**

Разработать web-приложение «Интернет магазин», которое позволяет:

- просматривать товары с различной степенью детализации;
- выполнять поиск товаров по разным критериям;
- сравнивать между собой два или несколько товаров;

- наполнять «корзину» выбранными для покупки товарами;
- корректировать список товаров в корзине;
- оформлять покупку;
- выполнять авторизацию пользователя.

Разрабатываемое приложение должно быть реализовано на основе клиент-серверной технологии, включать несколько страниц, удобные средства навигации.

### **Критерии оценки расчетно-графической работы**

✓ 10-8 баллов выставляется студенту, если студент полностью выполнил расчётно-графическое задание. Фактических ошибок, связанных с пониманием проблемы, нет; выполнены все этапы создания web-приложения; семантических и синтаксических ошибок в программах нет; выдержаны правила оформления HTML/CSS кода. При защите студент отвечает на все вопросы преподавателя.

✓ 7-6 баллов – работа выполнена полностью; есть незначительные погрешности в реализации отдельных компонент приложения, программах или в организации интерфейса с пользователем; выдержаны правила оформления HTML/CSS кода. При защите студент отвечает на все вопросы преподавателя.

✓ 5-4 балла – работа выполнена полностью, есть погрешности в реализации отдельных компонент приложения, в программах или в организации интерфейса с пользователем, связанные с непониманием предметной области. При защите студент не отвечает на 1-2 вопроса преподавателя.

✓ 1-3 балла – работа выполнена не полностью. Допущены ошибки в реализации отдельных компонент приложения, в программах или в организации интерфейса с пользователем, связанные с непониманием предметной области или неумением использования инструментов web-

разработки. При защите студент не отвечает на более, чем на 2 вопроса преподавателя.

### **Лабораторные работы по дисциплине «Инженерные web-технологии»**

Структура каждой лабораторной работы следующая:

1. Реализация web-страницы или ее фрагмента в качестве образца рассматриваемой технологии.
2. Задания для самостоятельной работы.

Все лабораторные работы можно проверить через специальное средство, реализованное в системе BlackBoard. Выполненные лабораторные работы необходимо защитить, но предварительно они должны быть отправлены преподавателю на проверку через систему BlackBoard.

#### **Типовая лабораторная работа на тему «Основы HTML»**

Порядок выполнения:

1. Создать каталог **Lab1**, в нем создать вложенный каталог **Pict** для хранения рисунков
2. Создать файл **Index.html**, реализовав в нем фреймовую структуру вида, приведенного на рисунке 1.

Задания для самостоятельной работы:

3. Создать следующие web-страницы:

**tutorialHead.html** – для заголовка

**tutorialMenu.html** – для описания структуры справочника

**tutorial1.html** для описания основных понятий HTML;

**tutorial2.html** – структуры документа;

**tutorial3.html** – форматирование текста;

**tutorial4.html** – оформление документа;

**tutorial5.html** - организации переходов по страницам.

4. Реализовать вызов страниц **tutorial\*.html** и загрузку их в правый фрейм по щелчку на соответствующий элемент списка в левом фрейме.

5. Реализовать перемещение по тексту страниц **tutorial\*.html** с помощью «закладок», то есть при щелчке на элемент вложенного списка, например *Графика*, необходимо переместится к соответствующему разделу текста в файле **tutorial4.html**.



|  <b>Справочник HTML</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• <i>Основные понятия HTML</i></li> <li>• <i>Структура документа</i></li> <li>• <i>Форматирование текста</i> <ul style="list-style-type: none"> <li>○ <i>Характеристики шрифта</i></li> <li>○ <i>Параграфы</i></li> <li>○ <i>Заголовки</i></li> <li>○ <i>Списки</i></li> </ul> </li> <li>• <i>Оформление документа</i> <ul style="list-style-type: none"> <li>○ <i>Цвета в HTML</i></li> <li>○ <i>Разделители</i></li> <li>○ <i>Графика</i></li> <li>○ <i>Специальные символы</i></li> </ul> </li> <li>• <i>Организация перехода</i> <ul style="list-style-type: none"> <li>○ <i>Гиперссылки</i></li> <li>○ <i>Закладки</i></li> </ul> </li> </ul> | <p>Документ HTML требует наличия определенных тегов в определенных позициях. Структура документа состоит из двух частей: заголовка и тела. Заголовок заключается в парный тег &lt;HEAD&gt;, а &lt;BODY&gt; служит для описания тела документа. Например:</p> <pre>&lt;HTML&gt;   &lt;HEAD&gt;     &lt;TITLE&gt; &lt;!-- Здесь идет заголовок документа в браузере --&gt; &lt;/TITLE&gt;   &lt;/HEAD&gt;   &lt;BODY&gt;     &lt;!-- Здесь идет тело документа --&gt;   &lt;/BODY&gt; &lt;/HTML&gt;</pre> <p>Документ на языке разметки определяется парным тегом &lt;HTML&gt;. Он открывается в самом начале документа, а закрывается в самом конце.</p> <p>Заголовок документа определяется парным тегом &lt;HEAD&gt;, после которого идут теги, задающие информацию о документе. В частности, заголовок документа в браузере (программа, предназначенная для просмотров документа Интернета) пишется между тегами &lt;TITLE&gt; и &lt;/TITLE&gt;.</p> <p>После тега &lt;BODY&gt; следует тело документа. Под "телом" понимается весь набор элементов на Web-странице. Это может быть текст, графика, ссылки и таблицы.</p> |

Рисунок 1. Вид страницы HTML

### Критерии оценки лабораторной работы

✓ 10-8 баллов выставляется студенту, если он выполнил все задания лабораторной работы, в том числе и самостоятельные. Фактических ошибок, связанных с пониманием проблемы, нет; семантических и синтаксических ошибок в программах нет; все инструменты web-разработки используются верно. При защите студент отвечает на все вопросы преподавателя.

✓ 7-6 баллов – работа выполнена полностью; студент выполнил все предложенные в лабораторной работе задания, одно самостоятельное задание реализовано не для всех исходных данных или есть погрешности в коде; все инструменты web-разработки используются верно. При защите студент отвечает на все вопросы преподавателя.

✓ 5-4 балла – работа выполнена полностью. Два самостоятельных задания реализованы не для всех входных данных или есть значительные погрешности в коде; часть инструментов web-разработки используются неверно. При защите студент не отвечает на 1-2 вопроса преподавателя.

✓ 1-3 балла – работа выполнена не полностью. Самостоятельные задания не выполнены или студент демонстрирует слабое владение инструментами web-разработки. При защите студент не отвечает более, чем на 2 вопроса преподавателя.

## **Комплекты заданий для контрольных работы по дисциплине**

### **«Инженерные web-технологии»**

#### **Раздел «Инструменты создания сайтов»**

##### **Типовой вариант**

Используя HTML и CSS создать страницы *Главная* и *Знания*.  
Необходимые рисунки и тексты скачать из системы BlackBoard.

**ТЕСТЫ**

ГЛАВНАЯ ИНТУИЦИЯ ЛОГИКА ИНТЕЛЛЕКТ ЗНАНИЯ

**ИНТЕЛЛЕКТ**

Многие считают, что интеллект – это то же самое, что ум. Показали практику интеллект – шаре. Это не только способность к логическому мышлению, но и умение правильно оценивать ситуацию и находить верные решения.

Интеллект – это также умение приспосабливаться к жизни и не теряться, когда она подкидывает сюрпризы. Что же касается логики и интуиции, это верные помощники интеллекта в решении различных проблем. Именно интеллект устанавливает, чему отдать предпочтение. С помощью тестов можно проверить, насколько хорошо логика и интуиция помогают решать мыслительные задачи.

**ЛОГИКА**

Каждый день мы сталкиваемся с множеством задач, решение которых требует от нас способности к логическому мышлению. Логика как умение думать и рассуждать последовательно и непротиворечно, требуется нам во многих жизненных ситуациях, начиная с решения сложных технических и бизнес-задач, заканчивая убеждением собеседников и совершенном покупке в магазине.

Но несмотря на высокую потребность в этом умении мы часто совершаем логические ошибки, сами того не подозревая. Ведь среди многих людей будет много, кто правильно мыслить можно на основе жизненного опыта и так называемого здравого смысла, но пользоваться законами и специальными приемами «формальной логики». Для совершения простых логических операций, высказывания элементарных суждений и логических выводов может пригодиться здравый смысл, а если нужно познать или объяснить что-то более сложное, то здравый смысл нередко приводит нас к заблуждениям.

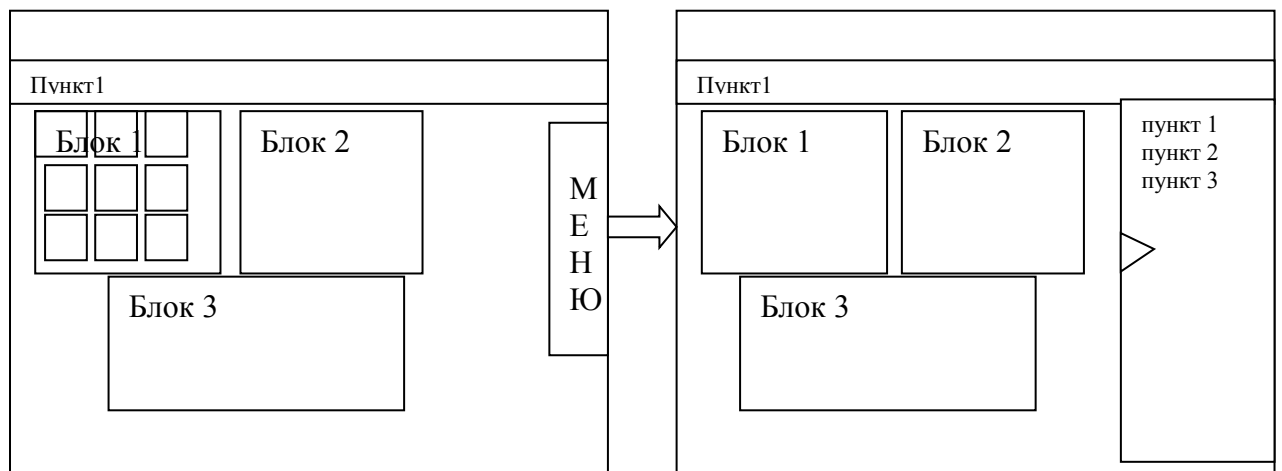
© ВЫСШЕГО СТУДЕНТА \_\_\_\_\_

## Раздел «Программирование на стороне клиента»

### Типовой вариант

Реализовать следующую страницу, используя HTML, CSS, JavaScript

библиотеку JQuery :



На которой

- **справа** – по щелчку на блок МЕНЮ раскрывается меню с пунктами, по щелчку по кнопке - меню сворачивается.

- **Блок 1**: Картинки расположены в 3 строки и три столбца, по щелчку на любую – она выводится в увеличенном размере поверх всех блоков страницы. Закрытие картинки осуществляется по щелчку по ней.



- **Блок 2:** Используется для размещения формы, которая включает поле для ввода, многострочное текстовое поле для вывода, кнопки очистить и запустить. Реализовать следующую задачу:

*Ввести  $n$ - размерности двумерного массива. Заполнить массив случайным образом. Вывести в поле вывода все седловые точки массива (Седловая точка – элемент, который является максимальным в строке и минимальным в столбце и наоборот).*

- **Блок 3:** Текстовый блок, в котором реализовать смену цвета последовательную смену цвета фона через некоторое количество миллисекунд (количество цветов – не менее 6).

## **Раздел «Основы теории баз данных»**

### **Типовой вариант**

*Задание 1.* Реализовать совокупность таблиц и информационно-логическую схему базы данных для заданной предметной области:

«Фирма выполняет ремонт компьютеров. Требуется разработать базу данных для хранения информации о выполнении ремонтных работ сотрудниками фирмы. Фирма имеет прайс лист, в котором описываются выполняемые работы и их стоимость. Также для некоторых работ требуются запасные части, стоимость которых тоже определена. В фирме имеется некоторое количество каждой запчасти. В фирме работают мастера, каждый из которых имеет категорию, с каждой категорией связывается фиксированный процент от выполнения работ.

При оформлении заказа фиксируется дата выполнения заказа, вид выполненной работы, используемые запчасти, исполнитель работы, а также информация о заказчике. В конце месяца каждому сотруднику начисляется зарплата в зависимости от его категории и выполненных работ».

*Задание 2.* Реализовать следующие запросы на выборку:

1. Вывести всех работников, имеющих заданную категорию
2. Вывести все работы, для которых требуется заданная запасная часть.

3. Вывести запасную часть, которая чаще всех используется в ремонтных работах.
4. Вывести работы, в которых требуются более 3-х запасных частей.
5. Начислить зарплату сотрудникам за определенный месяц
6. Вывести самого высокооплачиваемого сотрудника
7. Посчитать доход фирмы за месяц (сумма всех работ минус сумма, выплаченная сотрудникам)
8. Посчитать ежедневное количество заказов, выполненных сотрудниками, в заданный период.
9. Вывести количество сотрудников фирмы.

### **Критерии оценки контрольной работы**

✓ 10-8 баллов выставляется студенту, если студент выполнил контрольную работу полностью. Фактических ошибок, связанных с пониманием заданий, нет; семантических и синтаксических ошибок в программах нет; все элементы на странице отформатированы верно; все инструменты web-разработки используются верно. При защите студент отвечает на все вопросы преподавателя.

✓ 7-6 баллов – работа выполнена полностью; студент выполнил все задания контрольной, но есть погрешности в коде или в форматировании элементов; все инструменты web-разработки используются верно. При защите студент отвечает на все вопросы преподавателя.

✓ 5-4 балла – работа выполнена полностью; однако есть существенные погрешности в коде или в форматировании элементов; часть инструментов web-разработки используются неверно. При защите студент не отвечает на 1-2 вопроса преподавателя.

✓ 1-3 балла – работа выполнена не полностью. Большинство заданий выполнены неверно или студент демонстрирует слабое владение инструментами web-разработки. При защите студент не отвечает более, чем на 2 вопроса преподавателя.