

МИНОБРНАУКИ РОССИИ



Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный гуманитарный университет»
(ФГБОУ ВО «РГГУ»)

ФАКУЛЬТЕТ «ИНФОРМАЦИОННЫХ СИСТЕМ И БЕЗОПАСНОСТИ»
Кафедра «Информационных технологий и систем»

WEB-ПРОГРАММИРОВАНИЕ В ГУМАНИТАРНОЙ СФЕРЕ

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Направление подготовки 09.03.03 «Прикладная информатика»
Направленность «Прикладная информатика в гуманитарной сфере»

Уровень высшего образования: бакалавриат

Форма обучения: очная

РПД адаптирована для лиц
с ограниченными возможностями
здоровья и инвалидов

Москва 2022

Web-программирование в гуманитарной сфере

Рабочая программа дисциплины

Составитель:

к.х.н., с.н.с., доцент А.М. Подорожный

Ответственный редактор:

к.с-х.н., доцент, и.о. зав. кафедрой

Информационных технологий и систем Н.Ш. Шукенбаева

УТВЕРЖДЕНО

Протокол заседания кафедры

Информационных технологий и систем

№ 10 от 04.04.2022 г.

ОГЛАВЛЕНИЕ

1. Пояснительная записка.....	4
1.1 Цель и задачи дисциплины.....	4
1.2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций.....	4
1.3 Место дисциплины в структуре образовательной программы.....	4
2. Структура дисциплины.....	5
4. Содержание дисциплины.....	5
4. Образовательные технологии.....	6
5. Оценка планируемых результатов обучения.....	8
5.1 Система оценивания.....	8
5.2 Критерии выставления оценки по дисциплине.....	9
5.3 Оценочные средства (материалы) для текущего контроля успеваемости, промежуточной аттестации обучающихся по дисциплине.....	10
6 Учебно-методическое и информационное обеспечение дисциплины.....	13
6.1 Список литературы.....	13
6.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет»..	14
7 Материально-техническое обеспечение дисциплины.....	14
8 Обеспечение образовательного процесса для лиц с ограниченными возможностями здоровья и инвалидов.....	15
9 Методические материалы.....	16
9.1 Планы практических заданий.....	16
9.2 Иные материалы. Планы самостоятельной работы.....	17
Приложение 1. Аннотация дисциплины.....	26

1. Пояснительная записка

1.1. Цель и задачи дисциплины

Цель дисциплины: «Web-программирование в гуманитарной сфере» имеет целью теоретическое и практическое освоение методов и технологий, используемых в программной инженерии современных Web-документов.

Задачи: выработка у студентов системного подхода к решению задач программной инженерии интернет ресурсов, приобретение практических навыков построения различных web-систем на основе выбора наименее трудоемкой и, вместе с тем, адекватной поставленной задаче методологии разработки.

1.2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Компетенция	Индикаторы компетенций	Результаты обучения
ПК-2. Способен разрабатывать, тестировать и адаптировать прикладное программное обеспечение.	ПК-2.1. Знает основные среды для разработки программного обеспечения, методы тестирования и адаптации прикладного программного обеспечения.	Знать: принципы работы и основные протоколы сети интернет, языки программирования и разметки, подходы, используемые при создании сайтов, методы тестирования и адаптации программных компонентов интернет-ресурсов, технологии поисковой оптимизации и продвижения сайтов.
	ПК-2.2. Умеет разрабатывать, тестировать и адаптировать прикладное программное обеспечение.	Уметь: определять методы решения задач предметной области, с средствами сети интернет, разрабатывать структуру базы данных, адаптированную к использованию в интернет проектах, тестировать компоненты программного обеспечения интернет-ресурсов, оценивать эффективность использования конкретного языка и технологии при решении поставленных задач.
	ПК-2.3. Владеет современными языками программирования и методиками разработки, тестирования и адаптации прикладного программного обеспечения.	Владеть: навыками работы с инструментальными средствами разработки в сети интернет (HTML, CSS), языками PHP, MySQL и JavaScript, методами оптимизации, построения и продвижения сайтов. практическими навыками разработки и тестирования программных компонентов интернет-ресурсов.

1.3. Место дисциплины в структуре образовательной программы.

Дисциплина «Web-программирование в гуманитарной сфере» является элективной дисциплиной из части, формируемой участниками образовательных отношений учебного плана по направлению подготовки 09.03.03 Прикладная информатика, профиль: Прикладная информатика в гуманитарной сфере. Дисциплина реализуется на факультете Информационных систем и безопасности кафедрой Информационных технологий и

систем. Для освоения дисциплины необходимы знания, умения и владения, сформированные в ходе изучения следующих дисциплин: Архитектура вычислительных систем, Информационные системы, Программирование СИ++, Программирование Java, Операционные системы.

В результате освоения дисциплины формируются знания, умения и владения, необходимые для изучения следующих дисциплин: Мировые информационные ресурсы гуманитарной сферы, Управление информационными системами, Проектирование информационных систем.

2. Структура дисциплины

Общая трудоёмкость дисциплины составляет 4 з.е., 144 академических часа.

Структура дисциплины для очной формы обучения

Объем дисциплины в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Семестр	Тип учебных занятий	Количество часов
4	Лекции	24
4	Практические занятия	32
Всего:		56

Объем дисциплины в форме самостоятельной работы обучающихся составляет 88 академических часов.

3. Содержание дисциплины

№	Наименование раздела дисциплины	Содержание
1.	Основы работы сети интернет	<p>Интернет: история, современное состояние, органы управления, интернет вещей.</p> <p>Технические основы построения сетей: протоколы передачи данных, клиент-серверная технология, распределённая архитектура и маршрутизация, пакетная передача данных, уровни сетевого соединения модель OSI, URL и IP адреса, коммуникации интернета (проводные, беспроводные, оптоволокно), прием и обработка сигналов.</p>
2.	World Wide Web и другие прикладные сервисы	<p>Ресурс WWW: основные понятия, история развития. Другие ресурсы: почта web-mail и e-mail, FTP, терминальное управление, DNS, торрент, мессенджеры. Строение адреса URL: протоколы доступа к ресурсу, доменное имя, адреса порта и выводимого файла, вывод данных по запросу.</p> <p>Средства создания Web-страниц. Программные: HTML, CSS, JavaScript, MySQL, PHP, DOM, DHTML, пакет XAMPP. Конструкторы сайтов: технологии, виды, рейтинг, WordPress, Bitrix, упрощенные CMS и др. Виртуальные Web-редакторы их применение. Размещения сайтов в Сети: бесплатный домен, платный домен, собственный</p>

		домен, хостинг.
3.	Юзабилити сайта	<p>Понятие «Юзабилити». Постановка задачи на создание сайта, эволюция прототипов, этапы создания. Изучение аудитории, профилирование. Разработка семантического ядра. Учет психофизических характеристик людей при проектировании сайта. Структурирование информации, меры по повышению доступности сайта. Дизайн сайта: эстетическая и функциональная стороны, цвета, текст и фон, стили, элементы навигации, анимация, шрифты. Топологическая и файловая структура сайтов. Статичные и интерактивные сайты, средства, придающие интерактивность. Оптимизация крупных сайтов.</p>
4.	Верстка сайтов средствами HTML, CSS, JavaScript.	<p>Развитие, характеристика HTML 5. Правила записи тегов, контейнеры, автономные теги, атрибуты и пр. Структура документа HTML. Раздел Head: теги Meta, Link, Script, Style и др. Теги физического и логического форматирования текста, создание заголовков, списков, таблиц, вставка иллюстраций, блоки DIV, семантические теги, глобальные атрибуты.</p> <p>Назначение и синтаксис записи стилей CSS, способы помещения на веб-страницу. Наследование, каскадирование стилей. Стили форматирования документа, текста, шрифтов, графики, цвета и др.</p> <p>Основы JavaScript: способы подключения, помещения на странице. Возможности языка JavaScript: типы данных, операции, операторы, функции, объекты, массивы. Объектная модель документа DOM. Примеры использования JavaScript для верстки Web-страниц.</p>
5.	Клиент-серверное программирование с использованием PHP, MySQL	<p>Средства создания динамических ресурсов на стороне сервера и на стороне клиента. Выполняемые сервером функции, разделение на сервер базы данных и сервер приложений, примеры серверов. Конфигурация веб-сервера Apache в рамках сборки XAMPP: базовые команды настроек веб-сервера и обработчика приложений. Структура сайта на основании модели MVC.</p> <p>СУБД MySQL: пользователи и их права, типы данных, индексы и ключи, порядок проектирования базы данных, операции в базе данных СУБД MySQL.</p> <p>Язык PHP: переменные, типы данных, операции и операторы, базовые функции. Средства подключения файлов, работа с базой данных, использование форм, суперглобальных переменных.</p>
5.	Поисковые системы и поисковая оптимизация сайтов	<p>Появление и развитие поисковых систем, поисковые каталоги. Ведущие поисковые системы в России и в мире. Технология работы поисковых машин. Релевантность, пертинентность.</p> <p>Поисковая оптимизация сайта. Внутренние факторы ранжирования. Семантическое ядро ключевых слов, роль тегов Title, Description. Требования к структуре сайта и содержанию страниц. Ошибки, обман поисковика. Внешние факторы ранжирования. Google PageRank: определение, процедура подсчета, диапазоны. ТИЦ и ВИЦ Яндекса. Анализ конкуренции поисковых запросов, оптимизация высокочастотных и низкочастотных ключевых слов. Маркетинговые меры повышения популярности сайта.</p>

4. Образовательные технологии

Занятия лекционного типа.

Лекционные занятия включают изложение, обсуждение и разъяснение основных направлений и вопросов дисциплины «Web-программирование в гуманитарной сфере». На лекциях студенты получают необходимые знания по темам курса. Эти знания полезны в ходе реализации всех остальных видов занятий и в самостоятельной работе студентов.

Чтение лекций сопровождается показом на проекторе иллюстраций, презентаций, видеоматериалов; демонстрацией наглядных пособий: аппаратуры, различных предметов, печатных материалов и пр. Лекции имеют элементы интерактивности: допускается дискуссия, поощряются правильные ответы на вопросы.

Краткие записи лекций, конспектирование помогает усвоить материал. В ходе лекций не ставится задача дословной записи лекций под диктовку преподавателя. Но для подготовки к текущему контролю успеваемости и к промежуточной аттестации на почту студенческой группы высылаются PDF файлы с изложением пройденного материала. Файлы защищены паролем и другими способами, их содержимое не подлежит копированию и редактированию.

Практические занятия в компьютерном классе.

Задачи практических занятий: освоение ПО в сфере Web-программирования, а затем выполнение самостоятельных работ, предлагаемых преподавателем. Для освоения ПО составлены подробные инструкции по выполнению заданий. Для самостоятельной работы (а иногда и для освоения) имеются образцы того, что должно получиться.

Как правило, новое приложение лучше осваивать в следующей последовательности: базовые настройки, загрузка исходных данных (в виде файла или вручную), обработка данных основными инструментами, применение других эффективных методов. То есть, надо не просто описывать весь доступный интерфейс, а знакомить с возможностями программы, в ходе выполнения конкретных (лучше идущих из практики) задач.

Развитое, конкурентоспособное программное приложение может содержать сотни, даже свыше тысячи команд. Запомнить и профессионально освоить их за время занятий невозможно, и не нужно. Главное: ознакомиться со спектром возможностей программы. Тогда если в будущем перед человеком встанет нестандартная задача, то он вспомнит, что ее в принципе можно решить. Самостоятельно, а также с привлечением литературы и специалистов, он сможет ее реализовать.

В ходе занятий каждый студент осваивает ПО в естественном для себя темпе, по распечатанной инструкции. Поощряется взаимопомощь, полезная не только не только тому, кого консультируют, но и тому, кто консультирует.

При выполнении практических заданий по любой тематике нужно стремиться к тому, чтобы часть исходных данных произвольно, или по выбору задавалась самим студентом. Тогда точное повторение результатов работы будет невозможным. Полностью одинаковые работы могут получиться только в результате копирования, а не самостоятельного выполнения.

Опрос-коллоквиум.

Проводится в устной форме. Устный опрос позволяет оценить знания и кругозор студента, умение логически построить ответ, владение монологической речью и иные коммуникативные навыки. Опрос обладает большими возможностями воспитательного воздействия преподавателя, т. к. при непосредственном контакте создаются условия для неформального общения студентом.

Заблаговременно на почту группы отправляются перечень вопросов и материалы для подготовки к коллоквиуму. На контрольном занятии каждому студенту по случайной выборке даётся один из вопросов, при ответе на который он может готовиться не лимитируемое время (но ответить можно только на текущем занятии). При подготовке разрешается пользоваться любыми источниками.

По готовности студент отвечает на выданный ему вопрос, обычно в ходе ответа становится ясен уровень его знаний. Затем преподаватель задает ему вопросы в пределах всей темы, после чего итоговая оценка становится ясна как преподавателю, так и

студенту. Такая методика стимулирует студента при подготовке еще раз тщательно повторить весь курс, а не только заданный ему вопрос. Разумеется, преподаватель старается быть предельно объективным и не принимать во внимание характеристики и особенности студента, при условии соблюдения норм корректного поведения.

Тестирование.

Преимущественно осуществляется в ходе промежуточной аттестации по дисциплине. Программа тестирования разработана преподавателями из нескольких вузов и опубликована на ресурсе tester.pp.ru. При тестировании каждый студент получает случайно отобранный блок вопросов, по всем разделам курса, причём номера ответов каждый раз меняются; правильный ответ может оказаться в любой позиции. Это исключает составление «шпаргалки» по тесту. Имеются также развитые средства контроля процесса тестирования и обработки результатов.

Предварительно студентам отправляется список вопросов теста, без ответов, (в том числе и по практической работе в программах). При тестировании студенту даётся 24 вопроса. Время прохождения теста – 45 минут. Для каждого из вопросов теста существует 4 ответа, только один из 4-х правильный. Проходной балл: 50% - удовлетворительно (12-17 правильных ответов), 75 % - хорошо (18-21 правильных ответов), более 90% - отлично (22-24 правильных ответа).

Самостоятельная работа.

Целью самостоятельной (внеаудиторной) работы студентов является обучение навыкам работы с научно-теоретической, периодической, научно-технической литературой и технической документацией, необходимыми для углубленного изучения дисциплины «Web-программирование в гуманитарной сфере», а также развитие у них устойчивых способностей к самостоятельному изучению и изложению полученной информации. Кроме того, на самостоятельную работу выносятся практические задания, которые студент по тем или иным причинам не выполнил в компьютерном классе.

В период временного приостановления посещения обучающимися помещений и территории РГГУ для организации учебного процесса с применением электронного обучения и дистанционных образовательных технологий могут быть использованы следующие образовательные технологии:

- видео-лекции;
- онлайн-лекции в режиме реального времени;
- выполнение практических заданий и предоставление результатов преподавателю, он-лайн или по почте;
- электронные учебники, учебные пособия, научные издания в электронном виде и доступ к иным электронным образовательным ресурсам;
- системы для электронного тестирования;
- консультации с использованием телекоммуникационных средств.

5. Оценка планируемых результатов обучения

5.1. Система оценивания

Форма контроля	Макс. количество баллов	
	За одну работу	Всего
Текущий контроль:		
- опрос-коллоквиум	3 балла	18 баллов
- практические задания	4 балла	24 балла
- самостоятельные работы	2 балла	18 баллов
Зачет с оценкой по билетам		40 баллов
Итого за семестр		100 баллов

Полученный совокупный результат конвертируется в традиционную шкалу оценок и в шкалу оценок Европейской системы переноса и накопления кредитов (European Credit Transfer System; далее – ECTS) в соответствии с таблицей:

100-балльная шкала	Традиционная шкала		Шкала ECTS
95 – 100	Отлично	зачтено	A
83 – 94			B
68 – 82	Хорошо		C
56 – 67	Удовлетворительно		D
50 – 55			E
20 – 49	Неудовлетворительно	не зачтено	FX
0 – 19			F

5.2. Критерии выставления оценки по дисциплине

Баллы/ Шкала ECTS	Оценка по дисциплине	Критерии оценки результатов обучения по дисциплине
100-83/ A,B	«отлично»	<p>Выставляется обучающемуся, если он глубоко и прочно усвоил теоретический и практический материал, может продемонстрировать это на занятиях и в ходе промежуточной аттестации.</p> <p>Обучающийся исчерпывающе и логически стройно излагает учебный материал, умеет увязывать теорию с практикой, справляется с решением задач профессиональной направленности высокого уровня сложности, правильно обосновывает принятые решения.</p> <p>Свободно ориентируется в учебной и профессиональной литературе.</p> <p>Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.</p> <p>Компетенции, закреплённые за дисциплиной, сформированы на уровне – «высокий».</p>
82-68/ C	«хорошо»	<p>Выставляется обучающемуся, если он знает теоретический и практический материал, грамотно и по существу излагает его на занятиях и в ходе промежуточной аттестации, не допуская существенных неточностей.</p> <p>Обучающийся правильно применяет теоретические положения при решении практических задач профессиональной направленности разного уровня сложности, владеет необходимыми для этого навыками и приёмами.</p> <p>Достаточно хорошо ориентируется в учебной и профессиональной литературе.</p> <p>Оценка по дисциплине выставляется обучающемуся с</p>

Баллы/ Шкала ECTS	Оценка по дисциплине	Критерии оценки результатов обучения по дисциплине
		<p>учётom результатов текущей и промежуточной аттестации.</p> <p>Компетенции, закреплённые за дисциплиной, сформированы на уровне – «хороший».</p>
67-50/ D,E	«удовлетворительно»	<p>Выставляется обучающемуся, если он знает на базовом уровне теоретический и практический материал, допускает отдельные ошибки при его изложении на занятиях и в ходе промежуточной аттестации.</p> <p>Обучающийся испытывает определённые затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, владеет необходимыми для этого базовыми навыками и приёмами.</p> <p>Демонстрирует достаточный уровень знания учебной литературы по дисциплине.</p> <p>Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.</p> <p>Компетенции, закреплённые за дисциплиной, сформированы на уровне – «достаточный».</p>
49-0/ F,FX	«неудовлетворительно»	<p>Выставляется обучающемуся, если он не знает на базовом уровне теоретический и практический материал, допускает грубые ошибки при его изложении на занятиях и в ходе промежуточной аттестации.</p> <p>Обучающийся испытывает серьёзные затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, не владеет необходимыми для этого навыками и приёмами.</p> <p>Демонстрирует фрагментарные знания учебной литературы по дисциплине.</p> <p>Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.</p> <p>Компетенции на уровне «достаточный», закреплённые за дисциплиной, не сформированы.</p>

5.3. Оценочные средства (материалы) для текущего контроля успеваемости, промежуточной аттестации обучающихся по дисциплине

Вопросы к опросам-коллоквиумам

Тема 1.

1. История Интернета: сети, ARPANET, развитие интернета. (ПК-2.2)

2. Основные черты современного интернета, органы управления. (ПК-2.2)
3. Интернет вещей, его развитие. (ПК-2.1)
4. Понятия: протокол передачи данных, уровни сетевого соединения. (ПК-2.1)
5. Понятия: сервер и клиент, пакетная передача данных OSI. (ПК-2.1)
6. Распределенная архитектура и динамическая маршрутизация в интернете. (ПК-2.1)
7. Уровни прикладной, представления, сеансовый в модели OSI. (ПК-2.1)
8. Транспортный и канальный уровни в модели OSI. (ПК-2.1)
9. Сетевой уровень в модели OSI. (ПК-2.1)
10. Цифровые кабельные и беспроводные коммуникации интернета. (ПК-2.2)
11. Оптоволоконные коммуникации, прием и обработка сигналов в OSI. (ПК-2.2)

Тема 2.

1. World Wide Web как гипертекст. Основные определения. (ПК-2.2)
2. Возникновение и развитие World Wide Web, браузеров. (ПК-2.2)
3. Электронная почта e-mail и web-mail. Сравнение, применение. (ПК-2.2)
4. Протоколы FTP, удаленного доступа и другие прикладные сервисы. (ПК-2.2)
5. Структура адреса URL: протокол доступа, доменное имя, адрес порта, путь к файлу и параметры файла. (ПК-2.2)
6. Программные средства создания Web-страниц, пакет XAMPP. (ПК-2.2)
7. Конструкторы сайтов, их виды, возможности, визуальные Web-редакторы. (ПК-2.2)
8. Размещение в Интернете бесплатного и платного доменов. (ПК-2.2)
9. Создание собственного Web-домена. Хостинг. (ПК-2.2)

Тема 3.

1. Планирование Web-проекта: этапы, модели, приемы. (ПК-2.2)
2. Изучение целевой аудитории Web-сайта. (ПК-2.2)
3. Проектирование ядра ключевых слов. (ПК-2.2)
4. Психофизические аспекты создания Web-контента. (ПК-2.1)
5. Структурирование информации, повышение доступности сайта. (ПК-2.2)
6. Дизайн сайта: цветовые решения, текст, элементы навигации, анимация. (ПК-2.2)
7. Топологические и файловая структуры Web-сайтов. (ПК-2.2)
8. Интерактивные средства Web-сайтов. (ПК-2.2)
9. Личный кабинет, оптимизация крупных сайтов. (ПК-2.2)
10. Структура и средства создания интернет-магазина. (ПК-2.2)

Тема 4.

1. Программные средства создания Web-страниц: набор программного кода, Web-редакторы, Web-мастера, CMS-средства. (ПК-2.1)
2. Язык HTML, его версии. Правила написания тегов и их атрибутов. (ПК-2.3)
3. Назначение и синтаксис важнейших мета тегов. (ПК-2.3)

4. Теги HTML, управляющие разметкой Web-страницы, созданием комментариев. (ПК-2.3)
5. Теги HTML, управляющие форматированием текста на странице. (ПК-2.3)
6. Создание в HTML дополнительных элементов текста: списки, анимация текста, специальные символы. (ПК-2.3)
7. Подготовка иллюстраций для Web, вставка и оформление графики на Web-странице. (ПК-2.3)
8. Создание различных видов гиперссылок в HTML, работа с цветом и фоном. (ПК-2.3)
9. Роль таблиц в форматировании Web-страниц. Теги создания и форматирования таблиц. (ПК-2.3)
10. Стили CSS: назначение, возможности, способы помещения в код HTML. (ПК-2.3)
11. Наследование и классы стилей CSS. (ПК-2.3)
12. Типы переменных в JS. (ПК-2.3)
13. Условные конструкции в JS. (ПК-2.3)
14. Циклы в JS, их отличия. (ПК-2.3)

Тема 5.

1. Виды моделей взаимодействия клиент-серверных приложений. (ПК-2.1)
2. Модель MVC. Принцип, особенности, преимущества. (ПК-2.1)
3. Какие бывают базы данных. Что такое реляционная База данных? (ПК-2.1)
4. Типы данных полей в таблицах MySQL. (ПК-2.3)
5. Что такое первичный ключ, когда и для чего он может быть использован? (ПК-2.3)
6. Что такое индексы, для чего они используются. Отличия первичного ключа от уникального индекса. (ПК-2.3)
7. Какие операции можно выполнять в БД на примере таблиц в MySQL? (ПК-2.3)
8. Синтаксис операций Insert и Delete. (ПК-2.3)
9. Синтаксис операций Select и Update. (ПК-2.3)
10. Типы данных в PHP. (ПК-2.3)
11. Арифметические и логические операторы языка PHP. (ПК-2.3)
12. Условные операторы в PHP. (ПК-2.3)
13. Циклы в PHP. (ПК-2.3)
14. Структура функций в PHP. (ПК-2.3)
15. Подключение к базе данных средствами PHP. (ПК-2.3)
16. Функции для работы с базой данных в PHP. (ПК-2.3)

Тема 6.

1. Характеристика распространенных поисковых систем Интернета. (ПК-2.1)
2. Возникновение и развитие поисковых систем. Поисковые каталоги. (ПК-2.1)
3. Поисковые машины, этапы их работы, релевантность и пертинентность. (ПК-2.2)
4. Правила составления поисковых запросов. Команды языков поисковых запросов. (ПК-2.2)
5. Поисковая оптимизация. Внутренние и внешние факторы ранжирования Web-страниц. (ПК-2.2)

6. Приемы и правила поисковой оптимизации текста и структуры Web-страниц. Распространенные ошибки. (ПК-2.2)
7. Google PageRank: принципы расчета, использование. ТИЦ и ВИЦ Яндекса. (ПК-2.2)
8. Индексация сайта в поисковых системах, определение значимости ключевых слов. (ПК-2.2)
9. Маркетинговые методы повышения популярности сайта. (ПК-2.1)

Из вопросов коллоквиумов к зачету составляются билеты, по два вопроса в билете, обязательно из разных разделов.

Практический курс предусматривает проведение следующих работ:

1. Создание Web-страниц формата .html в текстовом редакторе Notepad++.
2. Создание коммерческого Web-сайта.
3. Установка и настройка сервера, создание структуры сайта.
4. Создание Web-сайта на локальном сервере XAMPP с помощью конструктора WordPress.
5. Работа с массивом информации в базе данных.
6. Анализ уровня конкуренции поисковых запросов.

Порядок проведения практических работ описан в разделе 9.1.

На самостоятельную работу выносятся 9 практических заданий, приведённых в разделе 9.2.

6. Учебно-методическое и информационное обеспечение дисциплины

6.1. Список литературы

Основная

1. Г.А. Лисьев, П.Ю. Романов, Ю.И. Аскерко. Программное обеспечение компьютерных сетей и web-серверов : учебное пособие / — Москва : ИНФРА-М, 2020. — 145 с.
Режим доступа: <https://new.znanium.com/catalog/product/1068576>.
2. Фаррелл, Б. Веб-компоненты в действии : практическое руководство / пер. с англ. Д. А. Беликов. - Москва : ДМК Пресс, 2020. - 462 с. - ISBN 978-5-97060-856-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1210671>
3. Немцова, Т. И. Компьютерная графика и web-дизайн : учебное пособие / Т.И. Немцова, Т.В. Казанкова, А.В. Шнякин ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2022. — 400 с. + Доп. материалы [Электронный ресурс]. — (Высшее образование). - ISBN 978-5-8199-0703-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1865592>
4. Цыгулин, А. А. Основы веб-программирования : учебное пособие / А. А. Цыгулин. - Новосибирск : Изд-во НГТУ, 2020. - 64 с. - ISBN 978-5-7782-4197-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1866934>

Дополнительная

1. Энж Э. SEO - искусство раскрутки сайтов: Самоучитель. - СПб:БХВ-Петербург, 2017. - 816 с.: ISBN 978-5-9775-3686-8 - Режим доступа: <https://new.znanium.com/catalog/product/978493>.
2. Локхарт, Д., Рагимов Р.Н. Современный PHP. Новые возможности и передовой опыт. - Москва : ДМК Пресс, 2016. - 304 с. - ISBN 978-5-97060-184-6. Электронный ресурс. Режим доступа: <https://new.znanium.com/catalog/product/1028044>.
3. Дронов В.А. PHP, MySQL, HTML5 и CSS 3. Разработка современных динамических Web-сайтов: Пособие / Дронов В.А. - СПб:БХВ-Петербург, 2016. - 688 с. ISBN 978-5-9775-3529-8 - Режим доступа: <http://znanium.com/catalog/product/944562>

6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Smashing Magazine – крупнейший ресурс, посвященный Веб-разработкам. Статьи, книги, сведения о вакансиях, заказах и пр. <https://www.smashingmagazine.com>.
2. Хабр – крупнейший русскоязычный ресурс, посвященный Веб-разработкам. Разработки, администрирование, дизайн, менеджмент, маркетинг и др. <https://habr.com/ru/>
3. <https://www.scopus.com> – Международная реферативная наукометрическая БД, Scopus, доступна по подписке.
4. <http://www.consultant.ru/> - КонсультантПлюс компьютерная справочная правовая система.

7. Материально-техническое обеспечение дисциплины

Для реализации программы дисциплины требуется:

- компьютеры, с предустановленным базовым программным обеспечением в составе ОС Windows 10 и MS Office;
- объединение компьютеров в локальную сеть компьютерного класса с высокоскоростным выходом в интернет;
- для лекционного курса – посадочные места по количеству обучающихся, доска, мультимедиа проектор с экраном;
- рабочее место преподавателя, оснащенное компьютером с выходом на проектор.

Используемое программное обеспечение:

1. Windows 10
2. Microsoft Office 2013 Pro
3. Mozilla Firefox 52.8.1 ESR
4. Кроссплатформенная сборка Web-сервера XAMPP 7.3.0
5. Adobe CS4 Master Collection
6. Kaspersky Endpoint Security
7. Платформа ZOOM

8. Обеспечение образовательного процесса для лиц с ограниченными возможностями здоровья и инвалидов

В ходе реализации дисциплины используются следующие дополнительные методы обучения, текущего контроля успеваемости и промежуточной аттестации обучающихся в зависимости от их индивидуальных особенностей:

- для слепых и слабовидящих: лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением; письменные задания выполняются на компьютере со специализированным программным обеспечением или могут быть заменены устным ответом; обеспечивается индивидуальное равномерное освещение не менее 300 люкс; для выполнения задания при необходимости предоставляется увеличивающее устройство; возможно также использование собственных увеличивающих устройств; письменные задания оформляются увеличенным шрифтом; экзамен и зачёт проводятся в устной форме или выполняются в письменной форме на компьютере.

- для глухих и слабослышащих: лекции оформляются в виде электронного документа, либо предоставляется звукоусиливающая аппаратура индивидуального пользования; письменные задания выполняются на компьютере в письменной форме; экзамен и зачёт проводятся в письменной форме на компьютере; возможно проведение в форме тестирования.

- для лиц с нарушениями опорно-двигательного аппарата: лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением; письменные задания выполняются на компьютере со специализированным программным обеспечением; экзамен и зачёт проводятся в устной форме или выполняются в письменной форме на компьютере.

При необходимости предусматривается увеличение времени для подготовки ответа.

Процедура проведения промежуточной аттестации для обучающихся устанавливается с учётом их индивидуальных психофизических особенностей. Промежуточная аттестация может проводиться в несколько этапов.

При проведении процедуры оценивания результатов обучения предусматривается использование технических средств, необходимых в связи с индивидуальными особенностями обучающихся. Эти средства могут быть предоставлены университетом, или могут использоваться собственные технические средства.

Проведение процедуры оценивания результатов обучения допускается с использованием дистанционных образовательных технологий.

Обеспечивается доступ к информационным и библиографическим ресурсам в сети Интернет для каждого обучающегося в формах, адаптированных к ограничениям их здоровья и восприятия информации:

- для слепых и слабовидящих: в печатной форме увеличенным шрифтом, в форме электронного документа, в форме аудиофайла.
- для глухих и слабослышащих: в печатной форме, в форме электронного документа.
- для обучающихся с нарушениями опорно-двигательного аппарата: в печатной форме, в форме электронного документа, в форме аудиофайла.

Учебные аудитории для всех видов контактной и самостоятельной работы, научная библиотека и иные помещения для обучения оснащены специальным оборудованием и учебными местами с техническими средствами обучения:

- для слепых и слабовидящих: устройством для сканирования и чтения с камерой SARA CE; дисплеем Брайля PAC Mate 20; принтером Брайля EmBraille ViewPlus;
- для глухих и слабослышащих: автоматизированным рабочим местом для людей с нарушением слуха и слабослышащих; акустический усилитель и колонки;

• для обучающихся с нарушениями опорно-двигательного аппарата: передвижными, регулируемые эргономическими партами СИ-1; компьютерной техникой со специальным программным обеспечением.

9. Методические материалы

9.1. Планы практических заданий

Практическое занятие №1. Создание Web-страниц формата .html в текстовом редакторе Notepad++.

- Общие положения.
- Создание стандартной структуры страницы.
- Разметка Web-страницы с помощью таблиц.
- Форматирование страницы средствами HTML.
- Форматирование текста средствами CSS.
- Добавление графики.
- Создание фона.
- Специальные символы.
- Создание гиперссылок.

Практическое занятие №2. Создание коммерческого Web-сайта.

- Определение параметров Web-проекта;
- Создание пустых страниц;
- Импорт и правка текста;
- Работа с таблицами;
- Создание и применение стилей CSS;
- Добавление иллюстраций;
- Создание навигационной панели и гиперссылок;
- Создание интерактивных элементов;
- Финишные операции.

Практическое занятие №3. Создание Web-сайта на сервере wordpress.com.

- Регистрация на wordpress.com, установка темы;
- Создание главной страницы;
- Создание других страниц;
- Установка виджетов;
- Загрузка плагинов.

Практическое занятие №4. Создание Web-сайта на локальном сервере XAMPP с помощью конструктора WordPress.

- Установка WordPress на XAMPP.
- Начало работы.
- Создание шапки сайта.
- Создание главного меню.
- Создание подвала.
- Дизайн и заполнение главной страницы.
- Блоги новостей и комментариев.
- Страницы с таблицами.
- Статичные и смешанные страницы.

Практическое занятие №5. Работа с массивом информации в базе данных.

- Создание и удаление таблицы базы данных.
- Добавление информации в таблицу базы данных.
- Редактирование информации в таблице базы данных по заданным условиям.
- Удаление записи из БД по условию.
- Поиск информации по условию, сортировка и группировка выборки.

Практическое занятие №6. Анализ уровня конкуренции поисковых запросов.

- Определение семантического ядра для предложенного материала.
- Определение среднего Page Rank и ТИЦ для выбранных поисковых фраз в первой десятке выдачи.
 - Определение среднего числа внешних ссылок в выдаче по версиям Google и Яндекса.
- Корректировка низкочастотных запросов по результатам анализа.

9.2. Иные материалы. Планы самостоятельной работы.**Самостоятельная работа №1 Форматирование****Задание 1**

Создайте в текстовом редакторе Блокнот веб-страничку в соответствии с образцом и сохраните ее в папке, названной вашей фамилией, под именем Text.html

Это заголовок 1

Это заголовок 2

Это заголовок 3

Дни недели (рабочие):

- Понедельник
- Вторник
- Среда
- Четверг
 1. первый урок
 2. второй урок
 3. перемена
- Пятница

Arial Courier Comic Sans MS Tahoma

Все это шрифт размера 5

Мы умеем использовать ^{верхний} и _{нижний} регистр (шрифт Tahoma размер 3)

**Травка зеленеет,
Солнышко блестит,
Ласточка с весною
В сени к нам летит.**

В строке заголовка браузера при просмотре должна быть фамилия исполнителя данного задания. Стихотворение напечатано зеленым цветом, а слово "ласточка" должно остаться черным, строки стихотворения отделены символом разрыва строки

Стремитесь к наименьшему объему файла – за лишние теги и атрибуты оценка будет снижаться.

Задание 2

Создайте веб-страничку, в заголовке которой укажите "Размещение рисунков", и сохраните ее в вашей папке под именем Pictures.html. Используя подходящие рисунки (они также должны быть сохранены непосредственно в вашей папке либо в специально созданном в ней каталоге Image), создайте страничку, представленную на рисунке.

Рисунок в верхней части страницы состоит из трех одинаковых картинок с границей по 5 пкс, размещенных по центру. Заголовок выделен красным. Вторая картинка размещена два раза, но имеет при этом различные размеры и свободное пространство со всех сторон по 20 пкс. Основной текст разделяется тегом разрыва строки, применено выравнивание по ширине.

Добавьте на страничку еще 2-3 подходящих рисунка, разместив их по своему усмотрению.





История земли Вятской

Вятская земля имеет богатую историю. Она стала заселяться еще в глубокой древности, очевидно, уже в верхнепалеолитическое время (50-15 тыс. лет назад). На территории области известны археологические памятники эпохи мезолита, неолита, бронзового века. В VII в. до н.э. в бассейне Вятки начался железный век. Ранний железный век здесь представлен памятниками ананьинской культуры. Ананьинцы принадлежали к финно-угорской этнической группе. Есть предположение, что они назывались тиссагетами, о которых упоминает древнегреческий историк Геродот, помещавший их к северо-востоку от скифов и сарматов. В конце XII-нач.XIII вв. в бассейн Вятки стали проникать русские, они селились на свободных землях среди удмуртов и марийцев. Вятка впервые упоминается в летописях под 1374 годом в связи с походом новгородских ушкуйников на Волжскую Болгарию, входившую в то время в состав Золотой Орды. "В лето 6882 (1374) идоша на низ рекою Вяткою ушкунцы разбойницы, 90 ушкунцев, и пограбиша Вятку и шедше взяша Болгары". В 70-е гг. XIV в. Вятская земля входила в состав Нижегородской провинции, присоединено к Москве. Нижегородские князья после долгой борьбы получили в удел Вятскую землю.



Внимание! Все рисунки необходимо оптимизировать (например, в PhotoShop) до объема не более 20-30 килобайт!

Задание 3

1. Создайте в своей папке на сервере главную веб-страничку **Index.html** еще три дополнительных странички с именами

Page1.html, Page2.html, Page3.html, поместите их в специально созданную папку **Pages**.

2. Разместите на каждой дополнительной странице по одной картинке (любой) и текст «Возврат на главную страницу». Картинки при необходимости оптимизируйте и разместите в папке **Image**.

3. Превратите эти картинки и текст в гиперссылки на главную страницу – **Index.html**.

4. На главной странице расположите ссылки-картинки и текстовые ссылки на дополнительные странички.

5. Также на главной странице разместите карту на основе картинки M16.jpg (ее размер 214x300 пкс). Выделенные на рисунке области сделайте гиперссылками все на те же три новых странички.

HTML-код создаваемой карты может выглядеть примерно так:





```
<IMG SRC="Image/M16.jpg" WIDTH="214" HEIGHT="300"
BORDER=0 USEMAP="#mymap">
<MAP NAME="mymap">
<AREA SHAPE="circle" COORDS="150,75,20"
HREF="Pages/Page1.html">
<AREA SHAPE="poly" COORDS="150,150,200,280,120,280"
HREF="Pages/Page2.html">
<AREA SHAPE="rect" COORDS="20,30,100,200"
HREF="Pages/Page3.html">
</MAP>
```

6. Оформите страницы, используя изменение цвета фона, гиперссылок. Примените фоновые рисунки. Наполните страницы содержанием.

Задание 4

Эти таблицы создайте на новой страничке **Table.html**. Сделайте ссылку любого типа на эту страницу с **Index.html**. Обратную гиперссылку сделайте из картинки, вставленной во вторую табличку.

Таблица №1

<p>Ротонда (архитектор А.Л. Витберг)</p>		<p>Александр- Невский собор (архитектор А.Л. Витберг)</p>	
<p>Жемчужины старой Вятки</p>			
	<p>Свято- Серафимовская церковь</p>		<p>Успенский собор Трифонова монастыря</p>

Каждый архитектурный памятник расположен в ячейке индивидуального фонового цвета, тот же цвет имеет ячейка с названием памятника (цвета подберите по своему вкусу). Гарнитуру, цвет и начертание текста "Жемчужины старой Вятки" также подберите самостоятельно. Картинку во второй таблице вставьте как фоновый рисунок, чтобы разместить на его фоне текст.

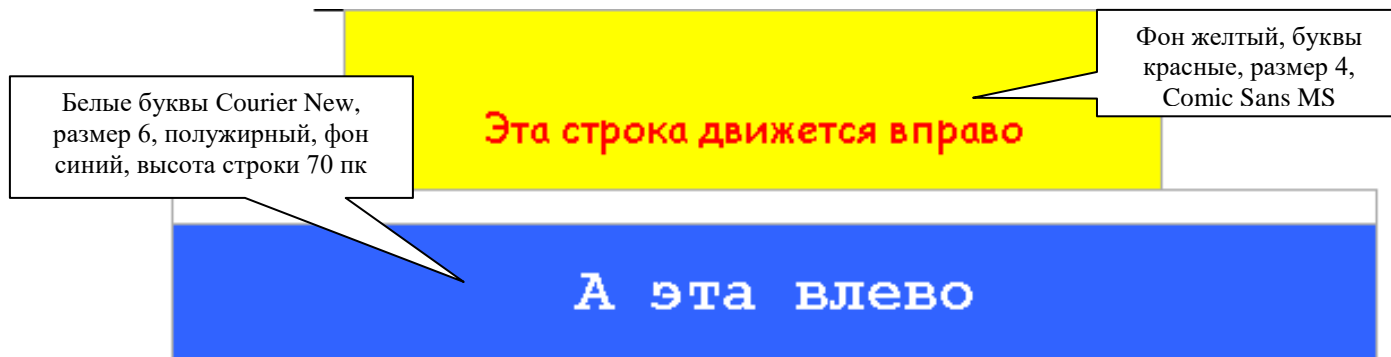
Таблица №2



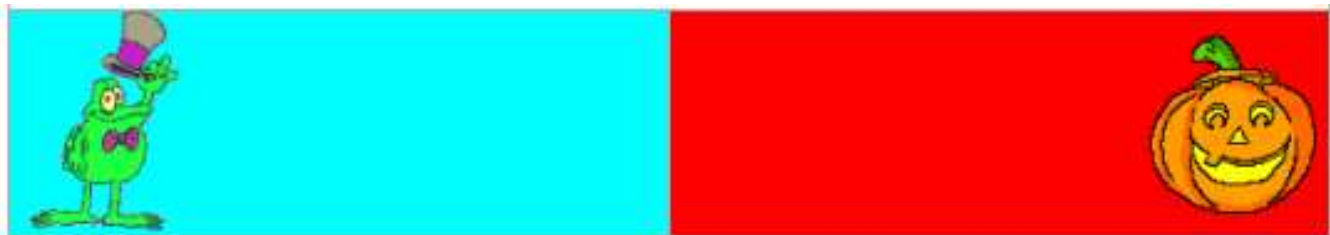
Задание 5

1. Создайте в папке, где хранятся Ваши предыдущие работы, веб-страничку Marquee.html.

2. Сделайте бегущие строки следующего содержания:



3. А эти бегущие строки, размещенные в ячейках невидимой таблицы, вместо текста содержат картинки, которые двигаются, «отражаясь» от краев окна и меняя направление движения, причем с разными скоростями. Лягушка двигается плавно, а тыква – скачками.



4. А эти буквы "живые" – они имеют размер 7, разный цвет, начертание и двигаются в разные стороны, не "уползая" за края ячеек.

		И				!
	Р		В		Т	
П				Е		

5. Дополните страничку несколькими горизонтальными линиями, употребив при этом различные атрибуты тега <HR>

Самостоятельная работа № 2 Рисунки и гиперссылки

Размещение рисунков.

Изображения, которые демонстрируются браузером при просмотре веб-страницы, хранятся в отдельных файлах формата gif, jpg (jpeg) или png, а в коде страницы делается ссылка на нужный файл. Для этого используется непарный тег , имеющий один обязательный и ряд необязательных атрибутов..

Обязательный атрибут:

SRC = url	Адрес графического файла (относительный или абсолютный)
------------------	---

Основные необязательные атрибуты:

ALT = текст	Альтернативный текст, выводимый в режиме браузера без загрузки изображений (обязательно заключается в кавычки)
BORDER = значение	Толщина обрамляющей рамки в пикселях, 0 означает отсутствие рамки (по умолчанию)
HEIGHT = значение	Высота изображения в пикселях (по умолчанию оригинальная) либо в процентах от высоты окна браузера
WIDTH = значение	Ширина изображения в пикселях (по умолчанию оригинальная) либо в процентах от ширины окна браузера
HSPACE = значение	Свободное пространство слева и справа от изображения в пикселях
VSPACE = значение	Свободное пространство сверху и снизу от изображения в пикселях
ALIGN = значение	Выравнивание изображения по горизонтали. Если заданы значения LEFT либо RIGHT, изображение соответствующим образом будет выровнено по горизонтали, задание этих значений обеспечивает обтекание изображения текстом

Задание 2

Создайте веб-страничку, в заголовке которой укажите "Размещение рисунков", и сохраните ее в вашей папке под именем Pictures.html. Используя подходящие рисунки (они также должны быть сохранены непосредственно в вашей папке либо в специально созданном в ней каталоге Image), создайте страничку следующего содержания (текст может быть другим):



История земли Вятской

Вятская земля имеет богатую историю. Она стала заселяться еще в глубокой древности, очевидно, уже в верхнепалеолитическое время (50-15 тыс. лет назад). На территории области известны археологические памятники эпохи мезолита, неолита, бронзового века. В VII в. до н.э. в бассейне Вятки начался железный век. Ранний железный век здесь представлен памятниками ананьинской культуры. Ананьинцы принадлежали к финно-угорской этнической группе. Есть

предположение, что они назывались тиссагетами, о которых упоминает древнегреческий историк Геродот, помещавший их к северо-востоку от скифов и сарматов.

В конце XII-нач.XIII вв. в бассейн Вятки стали проникать русские, они селились на свободных землях среди удмуртов и марийцев.

Вятка впервые упоминается в летописях под 1374 годом в связи с походом новгородских ушкуйников на Волжскую Болгарию, входившую в то время в состав Золотой Орды. "В лето 6882 (1374) идоша на низ рекою Вяткою ушкунцы разбойницы, 90 ушкунцев, и пограбиша Вятку и шедше взяша Болгары".

В 70-е гг. XIV в. Вятская земля входила в состав Нижегородского княжества. В 1393 г. это княжество было присоединено к Москве. Нижегородские князья после долгой борьбы вынуждены были покориться и получили в удел Вятскую землю.



Рисунок в верхней части страницы состоит из трех одинаковых картинок с границей по 5 пкс, размещенных по центру. Заголовок выделен красным. Вторая картинка размещена два раза, но имеет при этом различные размеры и свободное пространство со всех сторон по 20 пкс. Основной текст разделяется тегом разрыва строки, применено выравнивание по ширине.

Добавьте на страничку еще 2-3 подходящих рисунка, разместив их по своему усмотрению.

Внимание! Все рисунки необходимо оптимизировать (например, в PhotoShop) до объема не более 20-30 килобайт!

Гиперссылки

Гиперссылкой называют объект (текст, изображение, фрагмент изображения), при щелчке мышью по которому происходит переход к новому документу или фрагменту документа. Именно гиперссылки позволяют организовать переходы между любыми размещенными в сети Интернет документами.

Текстовые гиперссылки

Связь между HTML-документами и фрагментами документов организуется с помощью тега `<A> ... ` (от английского *anchor* - якорь).

Тег `<A>` употребляется как для создания ссылки на другой документ, так и для ссылки на фрагмент документа.

Обязательный атрибут:

<code>HREF = url</code>	Адрес целевого документа (может быть представлен в абсолютной и относительной форме)
-------------------------	--

Основные необязательные атрибуты:

TITLE="текст"	Выводит всплывающую подсказку при наведении указателя мыши на гиперссылку
---------------	---

Текст и изображения, размещенные между тегами, становятся активной зоной документа, чувствительной к щелчку мыши, который вызывает загрузку целевого документа. Текст гиперссылки выделяется подчеркиванием и цветами, указанными как значение атрибутов LINK, ALINK, VLINK тега <BODY> (либо цветом по умолчанию).

Гиперссылки-картинки

Чтобы сделать гиперссылкой целое изображение, также применяется тег <A>, только вместо текста (или вместе с текстом) между <A> и располагается тег <IMG...> со всеми соответствующими атрибутами.

Например, .

Карта гиперссылок

Тег <A> позволяет сделать активной зоной текстовый фрагмент или изображение целиком; для того же, чтобы **разные фрагменты** одного изображения ссылались на разные целевые документы, придется использовать тег <MAP>, реализующий изображение-карту.

У тега-контейнера <MAP> единственным обязательным атрибутом является **NAME**, значением которого будет имя (например, **NAME="mymap"**, которое должно быть использовано при описании атрибута **USEMAP** тега IMG, описывающего изображение, призванного служить картой (при этом к имени карты приписывается слева # – **USEMAP="#mymap"**)

Внутри контейнера <MAP> каждой чувствительной к перемещению мыши зоне изображения должен соответствовать тег <AREA> с атрибутами:

Синтаксис атрибута	Назначение
COORDS = <i>список</i>	Список через запятую координат активной зоны (зависит от типа заданной формы зоны)
HREF = <i>url</i>	Адрес целевого документа гиперссылки, связанного с указанной зоной
SHAPE = <i>форма</i>	Определяет форму активной зоны. Возможные значения этого атрибута: <i>circle</i> (окружность – задается координатами центра и радиусом в пикселях); <i>rect</i> (прямоугольник - задается координатами левого верхнего и правого нижнего угла); <i>poly</i> (многоугольник - задается координатами своих вершин) Координаты во всех случаях отсчитываются от верхнего левого угла изображения в пикселях, ось X направлена вправо, ось Y – вниз.
NOREF	иногда бывает нужно указать, что данная зона (заданная атрибутом SHAPE и координатами COORDS) не является активной, реакции на щелчок мыши нет

Задание 3

1. Создайте в своей папке на сервере главную веб-страничку **Index.html** еще одну дополнительную страничку с именем **Page1.html**, поместите ее в специально созданную папку **Pages**, туда же переместите ранее созданные странички **Text.html** и **Pictures.html**.

2. Разместите на каждой дополнительной странице по одной картинке (любой) и текст «Возврат на главную страницу». Картинки при необходимости оптимизируйте и разместите в папке **Image**.

3. Превратите эти картинки и текст в гиперссылки на главную страницу – **Index.html**.

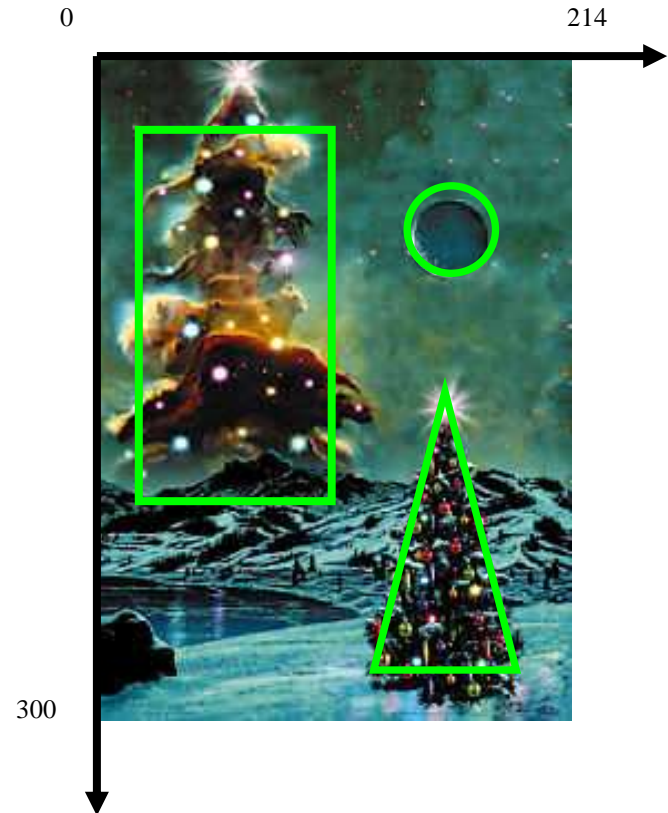
4. На главной странице расположите ссылки-картинки и текстовые ссылки на странички, расположенные в папке **Pages**.

5. Также на главной странице разместите карту на основе картинки **M16.jpg** (ее размер 214x300 пкс). Выделенные на рисунке области сделайте гиперссылками все на те же три новых странички.

HTML-код создаваемой карты может выглядеть примерно так:

```
<IMG SRC="Image/M16.jpg" WIDTH="214" HEIGHT="300"
BORDER=0 USEMAP="#mymap">
<MAP NAME="mymap">
<AREA SHAPE="circle" COORDS="150,75,20"
  HREF="Pages/Page1.html">
<AREA SHAPE="poly" COORDS="150,150,200,280,120,280"
  HREF="Pages/Page2.html">
<AREA SHAPE="rect" COORDS="20,30,100,200"
  HREF="Pages/Page3.html">
</MAP>
```

6. Оформите страницы, используя изменение цвета фона, гиперссылок. Примените фоновые рисунки. Наполните страницы содержанием.



Самостоятельная работа № 3 Пользовательские формы

Создайте страничку, предназначенную для сбора сведений о посетителях вашего сайта. Сведения должны пересылаться по электронной почте на ваш адрес и никак не кодироваться.

Используйте следующие элементы пользовательских форм:

- ✓ текстовая строка;
- ✓ поле для ввода пароля;
- ✓ поле выбора локального файла для загрузки на Web-сервер;
- ✓ опция выбора нескольких вариантов данных;
- ✓ опция переключения между несколькими вариантами данных;
- ✓ текстовое поле;

- ✓ ниспадающее меню;
- ✓ список наименований;
- ✓ кнопка отправления данных;
- ✓ кнопка сброса данных.

Ваша страничка может выглядеть, например, так:

The screenshot shows a web browser window with the title "Итоговый пример по организации пользовательских форм - Microsoft Internet Explorer". The browser's address bar and menu bar are visible. The main content area displays a survey form titled "ПОЛЬЗОВАТЕЛЬСКАЯ АНКЕТА" in red text. The form is organized into four quadrants:

- Анкетные данные (Survey Data):** Contains three text input fields labeled "Ваше имя:", "Ваш пароль:", and "Ваша фотография:". The "Ваша фотография:" field includes a small "Обзор..." button.
- Увлечения (Hobbies):** A list of five hobbies with checkboxes: "Компьютеры и Интернет" (checked), "Литература и искусство", "Путешествия и туризм", "Автомобили", and "Спорт и активный отдых".
- Место жительства (Place of Residence):** A dropdown menu for "Выберите город" with options: "Москва", "Санкт-Петербург", "Киров", and "Рязань". Below it is another dropdown for "Выберите страну".
- Комментарии (Comments):** A large text area for comments and two radio buttons: "Я впервые на этом сайте" and "А тут не в первый раз" (selected).

At the bottom of the form, there are two buttons: "Отправить" (Submit) and "Очистить" (Reset). The browser's status bar at the bottom shows "Готово" and "Мой компьютер".

Самостоятельна работа №4. Реализовать игру Пятнашки средствами javascript.

Самостоятельна работа №5. Реализовать форум на PHP.

Самостоятельна работа №7. Создание аналога wikipedia на движке Web2Py.

Самостоятельна работа №8. Реализовать экспорт данных в yaml и XML.

Самостоятельна работа №9. Добавить функции записи турниров с их воспроизведением.

АННОТАЦИЯ ДИСЦИПЛИНЫ

Дисциплина «Web-программирование в гуманитарной сфере» реализуется на факультете информационных систем и безопасности ИИНТБ РГГУ, кафедрой информационных технологий и систем.

Цель дисциплины: Web-программирование в гуманитарной сфере имеет целью теоретическое и практическое освоение методов и технологий, используемых в программной инженерии современных Web-документов.

Задачи: выработка у студентов системного подхода к решению задач программной инженерии интернет ресурсов, приобретение практических навыков построения различных web-систем на основе выбора наименее трудоемкой и, вместе с тем, адекватной поставленной задаче методологии разработки.

Дисциплина направлена на формирование следующих компетенций:

ПК-2 Способен разрабатывать, тестировать и адаптировать прикладное программное обеспечение.

- ПК-2.1. Знает основные среды для разработки программного обеспечения, методы тестирования и адаптации прикладного программного обеспечения.
- ПК-2.2. Умеет разрабатывать, тестировать и адаптировать прикладное программное обеспечение.
- ПК-2.3. Владеет современными языками программирования и методиками разработки, тестирования и адаптации прикладного программного обеспечения.

В результате освоения дисциплины обучающийся должен:

Знать: принципы работы и основные протоколы сети интернет, языки программирования и разметки, подходы, используемые при создании сайтов, методы тестирования и адаптации программных компонентов интернет-ресурсов, технологии поисковой оптимизации и продвижения сайтов.

Уметь: определять методы решения задач предметной области, с средствами сети интернет, разрабатывать структуру базы данных, адаптированную к использованию в интернет проектах, тестировать компоненты программного обеспечения интернет-ресурсов, оценивать эффективность использования конкретного языка и технологии при решении поставленных задач.

Владеть: навыками работы с инструментальными средствами разработки в сети интернет (HTML, CSS), языками PHP, MySQL и JavaScript, методами оптимизации, построения и продвижения сайтов. практическими навыками разработки и тестирования программных компонентов интернет-ресурсов.

По дисциплине предусмотрена промежуточная аттестация, в форме зачета с оценок.

Общая трудоемкость освоения дисциплины составляет 4 зачетных единицы.