МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИИСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И. Носова»

УТВЕРЖДАЮ:
Директор института энергетики и
автоматизированных систем
_____ С.И. Лукьянов
«27» сентября 2017г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Интернет программирование

Направление подготовки
09.03.03 Прикладная информатика

Профиль
Информационные системы и технологии в управлении ИТ-проектами

Уровень высшего образования – бакалавриат

Программа подготовки - академический бакалавриат

Форма обучения
Очная

Институт	Энергетики и автоматизированных систем
Кафедра	Бизнес-информатики и информационных технологий
Курс	3
Семестр	6

Магнитогорск
2017 г.

Рабочая программа составлена на основе ФГОС ВО по направлению 09.03.03 Прикладная информатика, утвержденного 12.03.2015 г. № 207 для профиля «Информационные системы и технологии в управлении ИТ-проектами».

Рабочая программа рассмотрена и одобрена на заседании кафедры Бизнес-информатики и информационных технологий «21» сентября 2017 г., протокол № 2.

Зав. кафедрой _____ Г.Н. Чусавитина

Рабочая программа одобрена методической комиссией института энергетики и автоматизированных систем «27» сентября 2017 г., протокол № 2.

Председатель _____ С.И. Лукьянов

Рабочая программа составлена: доцент, к.т.н.
(должность, ученая степень, ученое звание)

_____ / К.А. Рубан /
(подпись) (И.О. Фамилия)

Рецензент: начальник отдела программирования
ООО «Корпоративные системы Плюс»

/ Осипов Я. В./

Лист регистрации изменений и дополнений

№ п/п	Раздел программы	Краткое содержание изменения/дополнения	Дата. № протокола заседания кафедры	Подпись зав. кафедрой
1	3,4,7,8,9	Корректировка РПД в соответствии с новым макетом (распоряжение № 10-39/75 от 21.09.2018 «О формировании и актуализации образовательных программ). Актуализация информационно-методического и информационного обеспечения дисциплины. Актуализация материально-технического обеспечения дисциплины	25.09.18, протокол № 2	
2	8,9	О формировании и актуализации образовательных программ. Актуализация информационно-методического и информационного обеспечения дисциплины. Актуализация материально-технического обеспечения дисциплины	02.09.19, протокол № 1	
3	8	Актуализация учебно-методического и информационного обеспечения дисциплины	31.08.20, протокол №1	

1 Цели освоения дисциплины

Целями освоения дисциплины «Интернет программирование» являются: изучение методологических и концептуальных теоретических сведений о интернет-технологиях и стандартах, формирование у студентов умения и навыков работы с интернет приложениями, а также подготовка специалистов, умеющих применять современные методики разработки и сопровождения интернет приложений (в т.ч. на языке PHP), используемых в дальнейшей профессиональной деятельности.

2 Место дисциплины в структуре образовательной программы подготовки бакалавра

Дисциплина «Интернет программирование» входит в вариативную часть блока 1 образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения: «Прикладное программирование», «Вычислительные системы, сети и телекоммуникации», «Технологии баз данных и СУБД», «Языки и среды разработки Интернет приложений».

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы при изучении: «Разработка Интернет-приложений», «Корпоративные информационные системы».

3 Компетенции обучающегося, формируемые в результате освоения дисциплины и планируемые результаты обучения

В результате освоения дисциплины «Интернет программирование» обучающийся должен обладать следующими компетенциями:

Структурный элемент компетенции	Планируемые результаты обучения
ПК-8 – способностью программировать приложения и создавать программные прототипы решения прикладных задач	
Знать	– Знать особенности взаимодействия WEB-сервера и клиента. – Знать основные технологии программирования в части применения веб-скриптов. – Знать особенности реализации ИТ-сервисов различных видов предприятий
Уметь:	− Применять базовые ИКТ-решения в сфере интернет программирования. − Применять специализированные ИКТ-решения в сфере интернет программирования. − Рационально использовать базовые и специальные ИКТ-решения в сфере интернет программирования.
Владеть:	− Основными и расширенными методами решения задач в области интернет программирования и управления контентом.

4 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 4 единицы 144 акад. часа:

- контактная работа – 67,1 акад. часа:
- аудиторная работа – 64 акад. часа;
- внеаудиторная работа – 3,1 акад. часа;
- самостоятельная работа – 41,2 акад. час.
- подготовка к экзамену – 35,7 акад. часа

Раздел/ тема дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)			Самостоятельная работа (в акад. часах)	Вид самостоятельной работы	Формы текущего и промежуточного контроля успеваемости	Код и структурный элемент компетенции
		лекции	лаборат. занятия	практич. занятия				
Раздел 1. Механизмы взаимодействия web-сервера и клиента								
1.1 Протокол передачи гипертекста HTTP	6	2	2		3	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – з*
1.2 Веб-сервер: установка и настройка	6	1	4/2И		5	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – з*
1.3 Общие принципы обработки скриптов на сервере	6	1	4/2И		4	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – з*
Итого по разделу		**4**	**10/4И**		**12**			
Раздел 2. Изучение препроцессора гипертекста PHP								
2.1 Синтаксис языка PHP	6	1	6/2И		3	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – зув*
2.2 Реализация управляющих конструкций на языке PHP	**6**	2	6/2И		3	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – зув*

Раздел/ тема дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)			Самостоятельная работа (в акад. часах)	Вид самостоятельной работы	Формы текущего и промежуточного контроля успеваемости	Код и структурный элемент компетенции
		лекции	лаборат. занятия	практич. занятия				
2.3 Создание пользовательских функций на языке PHP	**6**	2	6/2И		3	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – зув*
2.4 Методы работы с массивами и строками на языке PHP	**6**	2	7/2И		4	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – зув*
2.5 Методы работы с файловой системой на языке PHP	**6**	2	6/2И		4	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – зув*
Итого по разделу		**9**	**31/10И**		**17**			
Раздел 3. Системы управления базами данных в веб-программировании								
3.1Общие принципы применения СУБД	**6**	2	2/2И		4,2	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – зув*
3.2 Взаимодействие PHP и MySQL	**6**	1	5/4И		8	Подготовка к лекции и лабораторной работе	Проверка отчетности по лабораторной работе	*ПК-8 – зув*
Итого по разделу		**3**	**7/6И**		**11**			
Итого по дисциплине		**16**	**48/20И**		**41,2**			

5 Образовательные и информационные технологии

В ходе самостоятельной работы осуществляется разбор постановки задач и их решений по текущей тематике. При реализации учебных задач курса предусмотрено широкое использование современных средств работы с HTML, CSS, стандартами, средствами разработки на PHP с использованием СУБД MySQL.

В ходе проведения занятий предусматривается следующие образовательные технологии:

1. Лабораторная работа – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с аналоговыми моделями реальных объектов.
2. Технологии проблемного обучения – организация образовательного процесса, которая предполагает постановку проблемных вопросов, создание учебных проблемных ситуаций для стимулирования активной познавательной деятельности студентов.
3. Лабораторное занятие на основе кейс-метода – обучение в контексте моделируемой ситуации, воспроизводящей реальные условия научной, производственной, общественной деятельности. Обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы базируются на реальном фактическом материале или же приближены к реальной ситуации.
4. Информационно-коммуникационные образовательные технологии – организация образовательного процесса, основанная на применении специализированных программных сред и технических средств работы с информацией.

В рамках практических занятий предусматривается использование средств вычислительной техники при выполнении индивидуальных заданий. Используется существующий образовательный портал университета (newlms.magtu.ru) для размещения ЭУМК по дисциплине. Текущий, промежуточный и рубежный контроль проводится на образовательном портале университета.

Основной образовательной технологией данного практикума является метод проектов. Каждый студент имеет изначально заданную тематику проекта, который он разрабатывает по мере изучения новых тем курса.

6 Учебно-методическое обеспечение самостоятельной работы обучающихся

В ходе самостоятельного изучения дисциплины рекомендуется использовать следующие образовательные технологии:

- возможности образовательного портала ФГБОУ ВО «МГТУ» для предоставления студентам графика самостоятельной работы, расписания консультаций, заданий для самостоятельного выполнения и рекомендуемых тем для самостоятельного изучения;

- традиционные технологии обучения в виде лекционных занятий с использованием мультимедийных средств и лабораторных практикумов в компьютерных классах ФГБОУ ВО «МГТУ».

Раздел/тема дисциплины	Вид самостоятельной работы	Кол-во часов	Формы контроля
Раздел 1. Механизмы взаимодействия web-сервера и клиента			
1.1 Протокол передачи гипертекста HTTP	Подготовка к лекции и лабораторной работе	3	Контроль выполнения лабораторной работы
1.2 Веб-сервер: установка и настройка	Подготовка к лекции и лабораторной работе	5	Контроль выполнения лабораторной работы
1.3 Общие принципы обработки скриптов на сервере.	Подготовка к лекции и лабораторной работе	4	Контроль выполнения лабораторной работы
Итого по разделу		**12**	Проверка отчетности по лабораторной работе
Раздел 2. Изучение препроцессора гипертекста PHP			
2.1 Синтаксис языка PHP	Подготовка к лекции и лабораторной работе	3	Контроль выполнения лабораторной работы
2.2 Реализация управляющих конструкций на языке PHP	Подготовка к лекции и лабораторной работе	3	Контроль выполнения лабораторной работы
2.3 Создание пользовательских функций на языке PHP	Подготовка к лекции и лабораторной работе	3	Контроль выполнения лабораторной работы
2.4 Методы работы с массивами и строками на языке PHP	Подготовка к лекции и лабораторной работе	4	Контроль выполнения лабораторной работы
2.5 Методы работы с файловой системой на языке PHP	Подготовка к лекции и лабораторной работе	4	Контроль выполнения лабораторной работы

Раздел/тема дисциплины	Вид самостоятельной работы	Кол-во часов	Формы контроля
Итого по разделу	Выполнение лабораторной работы	**17**	Проверка отчетности по лабораторной работе
Раздел 3. Системы управления базами данных в веб-программировании			
3.1Общие принципы применения СУБД	Подготовка к лекции и лабораторной работе	4,2	Контроль выполнения лабораторной работы
3.2 Взаимодействие PHP и MySQL	Подготовка к лекции и лабораторной работе	8	Контроль выполнения лабораторной работы
Итого по разделу	Выполнение лабораторной работы	**11**	Проверка отчетности по лабораторной работе
Итого по дисциплине		**41,2**	**Экзамен**

Перечень тем для подготовки к лабораторным работам:

1. Механизмы взаимодействия web-сервера и клиента
2. Синтаксис языка PHP
3. Реализация управляющих конструкций на языке PHP
4. Создание пользовательских функций на языке PHP
5. Методы работы с массивами и строками на языке PHP
6. Методы работы с файловой системой на языке PHP
7. Взаимодействие PHP и MySQL.

7 Оценочные средства для проведения промежуточной аттестации

а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации:

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
ПК-8 – способность программировать приложения и создавать программные прототипы решения прикладных задач		
Знать	Знать особенности взаимодействия WEB-сервера и клиента. Знать основные технологии программирования в части применения веб-скриптов. Знать особенности реализации ИТ-сервисов различных видов предприятий	***Примерный перечень вопросов к экзамену:*** 1. Серверный язык PHP. Синтаксис. Включение PHP-сценария в HTML-документ. 2. Управление интерпретатором PHP. 3. Управляющие структуры языка PHP. 1. Извлечение данных из полей форм. 2. Обработка форм. 3. Работа с массивами. Ассоциативные массивы. 4. Создание пользовательских функций, передача параметров функции и области видимости переменных. 5. Функции для работы со строками. Функции даты/времени и работы с календарем. 6. Работа с файлами. Операции с содержимым файлов. Работа с каталогами. 7. Хранение и использование данных пользователя. Способы хранения. Хранение данных в файлах. 8. Хранение данных в файлах. Открытие файла. Функция fopen(). Режимы файла. Чтение файла. Запись в файл. 9. Регулярные выражения, сопоставление и поиск с шаблоном. 10. Работа с динамическими изображениями. Создание и подключение модулей. 11. Отслеживание сеанса, управление сессиями и cookies. 12. Организация ветвлений. 13. Понятие класса. Основные компоненты класса. 14. Абстрагирование, инкапсуляция, модульность и иерархия. 15. Организация счетчика посещений.

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
Уметь	Применять базовые ИКТ-решения в сфере интернет программирования. Применять специализированные ИКТ-решения в сфере интернет программирования. Рационально использовать базовые и специальные ИКТ-решения в сфере интернет программирования.	***Примерные практические задания:*** 1. Установить и сконфигурировать веб-сервер Apache. 2. Установить и сконфигурировать интерпретатор PHP. 3. Установить и сконфигурировать СУБД MySQL. 4. Установить и сконфигурировать CMS по выбору. 5. Разработать и реализовать алгоритм по заданию преподавателя.
Владеть	− Основными и расширенными методами решения задач в области интернет программирования и управления контентом.	***Проектное задание:*** Разработать клиент-серверное приложение.

б) порядок проведения промежуточной аттестации, показатели и критерии оценивания:

Промежуточная аттестация по дисциплине включает теоретические вопросы, позволяющие оценить уровень усвоения обучающимися знаний, и практические задания, выявляющие степень сформированности умений и владений. Экзамен по данной дисциплине проводится в устной форме по экзаменационным билетам, каждый из которых включает 2 теоретических вопроса и одно практическое задание.

Показатели и критерии оценивания экзамена:

– на оценку **«отлично»** – обучающийся демонстрирует высокий уровень сформированности компетенций, всестороннее, систематическое и глубокое знание учебного материала, свободно выполняет практические задания, свободно оперирует знаниями, умениями, применяет их в ситуациях повышенной сложности.

– на оценку **«хорошо»** – обучающийся демонстрирует средний уровень сформированности компетенций: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.

– на оценку **«удовлетворительно»** – обучающийся демонстрирует пороговый уровень сформированности компетенций: в ходе контрольных мероприятий допускаются ошибки, проявляется отсутствие отдельных знаний, умений, навыков, обучающийся испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

– на оценку **«неудовлетворительно»** – обучающийся не может показать знания на уровне воспроизведения и объяснения информации, не может показать интеллектуальные навыки решения простых задач.

8 Учебно-методическое и информационное обеспечение дисциплины (модуля)

а) Основная литература:

1. Лисьев, Г.А. Программное обеспечение компьютерных сетей и web-серверов : учебное пособие / Г. А. Лисьев, П. Ю. Романов, Ю. И. Аскерко. — Москва : ИНФРА-М, 2020. — 145 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-013565-6. - Текст : электронный. - URL: https://znanium.com/read?id=350977

2. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2020. — 218 с. — (Высшее образование). — ISBN 978-5-534-00515-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/viewer/proektirovanie-i-razrabotka-web-prilozheniy-451207

б) Дополнительная литература:

1. Романова М. В. Разработка Web-страниц и презентаций [Электронный ресурс] : практикум / М. В. Романова, Е. В. Чернова. - Магнитогорск : МГТУ, 2017. - 70 с. : ил., табл. - Режим доступа: https://magtu.informsystema.ru/uploader/fileUpload?name=2704.pdf&show=dcatalogues/1/1131734/2704.pdf&view=true . - Макрообъект.

2. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 403 с. — (Высшее образование). — ISBN 978-5-534-12256-5. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/viewer/programmirovanie-na-sql-v-2-ch-chast-1-452357

3. Лавлинский, В. В. WEB-инжиниринг: Учебное пособие / Лавлинский В.В., Табаков Ю.Г. - Воронеж:ВГЛТУ им. Г.Ф. Морозова, 2013. - 268 с. - Текст : электронный. - URL: https://znanium.com/read?id=220865

4. Журнал «Программные продукты и системы»[Электронный ресурс]. Научно-исследовательский институт «Центрпрограммсистем» — Режим доступа: https://e.lanbook.com/journal/2276#journal_name

в) Методические указания:

Методические указания по выполнению лабораторных заданий представлены в приложении.

г) Программное обеспечение и Интернет-ресурсы:

Наименование ПО	№ договора	Срок действия лицензии
MS Windows 7	Д-1227 от 08.10.2018	11.10.2021
MS Office 2007	№ 135 от 17.09.2007	бессрочно
7Zip	свободно распространяемое	бессрочно
SCO OpenServer	свободно распространяемое	бессрочно
Google Chrome	свободно распространяемое ПО	бессрочно

Профессиональные базы данных и информационные справочные системы:

1. Официальный Интернет-ресурс Федерального агентства по техническому регулированиюи метрологии . – http://www.gost.ru

2. Справочник по ГОСТам и стандартам. Информационные технологии. [Электронный ресурс]. Информационное агентство MetalTorg.Ru. — Режим доступа: http://gostbank.metaltorg.ru/oks/629/

Интернет-ресурсы:

1. On-line учебник HTML5BOOK — Режим доступа: https://html5book.ru/
2. PHP: справочник языка - Режим доступа: http://php.net/manual/ru/langref.php

9 Материально-техническое обеспечение дисциплины

Материально-техническое обеспечение дисциплины включает:

Тип и название аудитории	Оснащение аудитории
Учебные аудитории для проведения занятий лекционного типа	Специализированная (учебная) мебель (столы, стулья, доска аудиторная), мультимедийное оборудование (проектор, компьютер, экран) для презентации учебного материала по дисциплине;
Учебные аудитории для проведения лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Специализированная (учебная) мебель (столы, стулья, доска аудиторная), персональные компьютеры объединенные в локальные сети с выходом в Интернет и с доступом в электронную информационно-образовательную среду университета, оснащенные современными программно-методическими комплексами
Аудитории для самостоятельной работы (компьютерные классы; читальные залы библиотеки)	Специализированная (учебная) мебель (столы, стулья, доска аудиторная), персональные компьютеры объединенные в локальные сети с выходом в Интернет и с доступом в электронную информационно-образовательную среду университета, оснащенные современными программно-методическими комплексами
Помещение для хранения и профилактического обслуживания учебного оборудования	Мебель (столы, стулья, стеллажи для хранения учебно-наглядных пособий и учебно-методической документации), персональные компьютеры.

ПРИЛОЖЕНИЕ

Приложение А

Методические рекомендации по выполнению лабораторных работ

Лабораторная работа №1. Арифметические операторы

Цель: изучить правила работы с PHP. Изучить основные операторы PHP.

Пример:

Даны два числа 5 и 7. Найти их сумму и произведение.

Решение:

```
<?php
$a = 5;
$b = 7;
$c = $a + $b;
$d = $a * $b;
echo $c, $d;
?>
```

Задание: Напишите скрипт PHP. Ответ представьте в виде файла PHP.

Вариант №1. Даны два числа 4 и 6. Найдите сумму их квадратов.

Вариант №2. Даны три числа 3, 5, 8. Найдите их среднее арифметическое.

Вариант №3. Даны три числа x = 2,y = 6 и z = 9. Найдите (x+1)4−2(z−2x^2+y^2)

Вариант №4. Даны три ненулевых числа $a = 4, $b = 8, $c = 3. Найдите всевозможные результаты деления суммы двух из них на оставшееся третье число

Вариант №5. Дано два числа 17 и 54. Найдите сумму 40% от первого числа и 84% от второго числа.

Вариант №6. Дано трехзначное число 567. Найдите сумму его цифр.

Вариант №7. Даны два числа 10 и 6. Найдите остаток от деления.

Вариант №8. Дано число 345. Найти сумму цифр данного числа.

Вариант №9. Даны два числа 66 и 55. Увеличить каждое число на 1 (инкремент) и вывести разность полученных значений.

Вариант №10. Дано число 3424. Вывести сумму первой и последней цифры числа.

Лабораторная работа №2. Математические функции

Цель: Изучить основные математические стандартные функции PHP (http://php.net/manual/ru/ref.math.php)

Задание: Написать скрипт на PHP, используя стандартные математические функции PHP. Ответ представить в виде PHP файла.

Вариант №1. Дано число в двоичной системе счисления (000110011), преобразовать число в десятичную систему счисления.

Вариант №2. Дано число в десятичной системе счисления (323), преобразовать число в двоичную систему счисления.

Вариант №3. Даны три числа: 34, 5, 223. Найти максимальное значение и вывести на экран.

Вариант №4. Дано число в десятичной системе счисления (5643), преобразовать число в восьмеричную систему счисления.

Вариант №5. Даны два числа: 45 и 55. Найти минимальное значение и вывести на экран.

Вариант №6. Вывести на экран результат возведения числа 5 в степень 6.

Вариант №7. Выводит сгенерированные случайным образом три числа в диапазоне от 0 до 10000.

Вариант №8. Дано число в десятичной системе счисления (2319), преобразовать число в шестнадцатеричную систему счисления.

Вариант №9. Выводит квадратный корень чисел: 49, 100, 225.

Вариант №10. Вывести округленное число 4.5555.

Лабораторная работа №3. Условный оператор

Цель: изучить правила построения условных операторов. Работа с формой.

Задание:

Вариант №1. Создать форму для ввода одного числа. Если оно больше 10, то увеличьте его на 100, иначе уменьшите на 30. Выведите результат.

Вариант №2. Создать форму для ввода одного числа. Если оно четное, то уменьшите его в 2 раза, иначе увеличьте в 3 раза. Выведите результат.

Вариант №3. Создать форму для ввода одного числа. Если оно не меньше 50, то выведите квадрат этого числа, если же это число больше 10 и меньше 30, то выведите ноль, в остальных случаях выведите слово "Ошибка"

Вариант №4. Создать форму для ввода двух чисел. Вывести наибольшее из них.

Вариант №5. Создать форму для ввода двух чисел. Вывести 'Да', если они отличаются на 100, иначе вывести 'Нет'.

Вариант №6. Создать форму для ввода двух чисел. Вывести 'Да', если они отличаются не более чем на 20, иначе вывести 'Нет'.

Вариант №7. Создать форму для ввода одного трехзначного числа. В данном трехзначном числе переставьте цифры так, чтобы новое число оказалось наибольшим из возможных.

Вариант №8. Пользователь вводит номер дня недели в форму. Вывести название дня недели.

Вариант №9. Пользователь вводит свой возраст в форму. Если он больше 80 лет, то вывести 'Здравствуйте, уважаемый', иначе 'Успехов!'.

Вариант №10. Создать форму для ввода одного числа. Проверить является ли число трехзначным, вывести ДА, либо НЕТ.

Лабораторная работа №4. Простые массивы в PHP

Цель: изучить правила работы с массивами в PHP. Циклы.

Задание:

Вариант №1. Дан массив с числами. Запишите в новый массив только те числа, в которых есть цифра 5.

Вариант №2. Дан массив с числами. Выведите массив отсортированный по возрастанию.

Вариант №3. Дан массив с числами. Выведите массив отсортированный по убыванию.

Вариант №4. Дан массив с числами. Найдите и выведите на экран максимальное значение в массиве.

Вариант №5. Дан массив с числами. Найдите и выведите на экран минимальное значение в массиве.

Вариант №6. Дан массив с числами. Проверьте есть ли массиве отрицательные значения. Если есть выведите ДА, иначе НЕТ.

Вариант №7. Дан массив с числами. Выведите массив сдвинув все элементы вправо на одну позицию, в начало массива запишите последний элемент.

Вариант №8. Напишите скрипт, который будет сливать два массива таким образом: из, к примеру, [1, 2, 3] и ['a', 'b', 'c'] она сделает [1, 'a', 2, 'b', 3, 'c'].

Вариант №9. Дан массив с числами. Создайте новый массив, элементы которого будут иметь противоположные знаки.

Вариант №10. Дан массив с числами. Посчитать сумму всех элементов массива

и вывести ее на экран.

Лабораторная работа №5. Ассоциативные, многомерные массивы в PHP

Цель: изучить правила работы с ассоциативными и многомерными массивами.

Задание:

Вариант №1. Необходимо написать программу, которая по фамилии некоторого человека из группы будет выдавать его имя и дату рождения. Фамилия вводится через форму. Изначально дан ассоциативный двумерный массив, в качестве ключей даны фамилии, значением элементов массива являются имена людей и даты их рождения.

Вариант №2. Необходимо написать программу, которая по названию животного будет выдавать его место обитания и изображение. Название животного вводится через форму. Изначально дан ассоциативный двумерный массив, в качестве ключей даны названия животных, значением элементов массива являются место обитания и URL к изображению.

Вариант №3. Необходимо написать программу, которая по названию сайта будет выдавать его рабочие гиперссылки. Название сайта вводится через форму. Изначально дан ассоциативный двумерный массив, в качестве ключей даны названия сайтов, значением элементов массива являются гиперссылки.

Вариант №4. Необходимо написать программу, которая по фамилии учащегося будет выдавать его успеваемость по нескольким дисциплинам. Фамилия вводится через форму. Изначально дан ассоциативный многомерный массив, в качестве ключей фамилии учащихся, названия дисциплин, значением элементов массива оценки по дисциплинам.

Вариант №5. Необходимо написать программу, которая по названию времени года будет выдавать названия месяцев сезона, для каждого месяца количество дней. Название времени года вводится через форму.

Вариант №6. Необходимо написать программу, которая по названию фамилии сотрудника будет выдавать его должность и заработную плату. Фамилия вводится через форму. Изначально дан ассоциативный двумерный массив, в качестве ключей даны фамилии сотрудников, значением элементов массива являются должности и заработная плата.

Вариант №7. Необходимо написать программу, которая по названию ФИО автора будет выдавать его книги. ФИО вводится через форму. Создать для хранения данных многомерный ассоциативный массив, содержащий данные по книгам в следующем формате: Название; Год издания; Объем. Обратите внимание количество книг у отдельных авторов может быть различным.

Вариант №8. Необходимо написать программу, которая по названию фамилии преподавателя будет выдавать читаемые им дисциплины. Фамилия вводится через форму.Изначально дан ассоциативный двумерный массив, в качестве ключей даны фамилии преподавателей, значением элементов массива являются читаемые дисциплины. Обратите внимание количество читаемых дисциплин у каждого преподавателя может быть различным.

Вариант №9. Необходимо написать программу, которая по названию фирмы будет выдавать виды выпускаемых ею изделий. Название фирмы вводится через форму. Изначально дан ассоциативный двумерный массив, в качестве ключей даны фирмы, значением элементов массива являются выпускаемые изделия.

Вариант №10. Необходимо написать программу, которая по названию планеты Солнечной системы будет выдавать ее изображение и краткое описание. Название планеты вводится через форму. Изначально дан ассоциативный двумерный массив, в качестве ключей даны названия планет Солнечной системы, значением элементов массива являются изображение планет и краткое описание.

Лабораторная работа №6. Функции. Строки в PHP

Цель: изучить функции PHP для обработки строк.

Задание:

Вариант №1. Дана строка. Переведите все первые буквы слов в верхний регистр, удалите лишние пробелы, подряд идущие точки и запятые.

Вариант №2. Дана строка. Проверить, является ли строка палиндромом. Учитывая, что пробелы и прописные буквы не являются отличительным свойством.

Вариант №3. Дана строка. Проверить является ли строковое значение названием электронного почтового ящика в требованиях RFC2142. (Используйте регулярные выражения).

Вариант №4. Дан текст. Преобразовать исходный текст, сначала удалив из него все html теги, далее после каждой точки добавить перенос на следующую строку.

Вариант №5. Дана дата в формате 21.05.1999, либо 21/05/1999, либо 21-05-1999. Преобразовать в следующий вид: 21 мая 1999 г. Обратите внимание, дата может быть любой, необходимо учитывать склонение месяца.

Вариант №6. Дана строка. Преобразуйте строку по следующему правилу: заменить все буквы на их порядковый номер в латинском алфавите, все не буквенные символы на 0.

Вариант №7. Дан текст. Преобразовать исходный текст по следующему правилу: подряд идущие одинаковые символы заменяются на один символ, после каждой точки добавить (если таковой отсутствует) перенос на следующую строку.

Вариант №8. Дана строка. Заменить каждый символ строки на соответствующий код символа. К каждой цифре строки добавьте 1. Выведите зашифрованную строку. Напишите соответствующий скрип, который будет расшифровывать такие строки.

Вариант №9. Дана строка. Удвоить каждый символ строки. К каждой цифре строки добавьте 5. Выведите зашифрованную строку. Напишите соответствующий скрип, который будет расшифровывать такие строки.

Вариант №10. Дана строка в виде последовательности чисел разной длины, разделенные проблемами. Сформировать новую строку, содержащую только числа с цифрой 3, удалив все пробелы. Вывести полученное число в форматированном виде (например, 1 222 222 078).

Лабораторная работа №7. Файлы в PHP

Цель: изучить работу с файлами в PHP, форматы xml и json.

Задание:

Вариант №1. Создать html форму с полями: ФИО автора, название произведения, количество страниц, год издания. Написать скрипт, который сохраняет введенную информацию в файл на сервере в папке "Books" в формате xml. В случае, если файл с папкой не существует на сервере, то скрипт папку и файл создает. Если файл существует - скрипт добавляет в конец файла переданные данные с новой строки. На отдельной странице реализовать чтение файла в отформатированном виде.

Вариант №2. Создать html форму с полями: ФИО автора, название произведения, количество страниц, год издания. Написать скрипт, который сохраняет введенную информацию в файл на сервере в папке "Person/Lib" в формате json. В случае, если файл с папкой не существует на сервере, то скрипт папку и файл создает. Если файл существует - скрипт добавляет в конец файла переданные данные с новой строки. На отдельной странице реализовать чтение файла в отформатированном виде.

Вариант №3. Создать html форму с полями: дата новости, название новости, аннотация новости, автор новости, картинка для новости. Написать скрипт, который сохраняет введенную информацию в файл на сервере, файл с картинкой сохраняет в папке Images, всю информацию добавляет в формате json. На отдельной странице реализо-

вать чтение файла в отформатированном виде.

Вариант №4. Создать html форму с полями: дата новости, название новости, аннотация новости, автор новости, картинка для новости. Написать скрипт, который сохраняет введенную информацию в файл на сервере, файл с картинкой сохраняет в папке Images, всю информацию добавляет в формате xml. На отдельной странице реализовать чтение файла в отформатированном виде.

Вариант №5. Создать html форму с полями: имя пользователя, e-mail, тип сообщения (выпадающий список со значениями: жалоба, предложения, благодарность), текст сообщения. Написать скрипт, который сохраняет введенную информацию в файл на сервере в формате xml. На отдельной странице реализовать чтение файла с пагинацией (5 сообщений на странице).

Вариант №6. Создать html форму с полями: имя пользователя, e-mail, тип сообщения (выпадающий список со значениями: жалоба, предложения, благодарность), текст сообщения. Написать скрипт, который сохраняет введенную информацию в файл на сервере в формате json. На отдельной странице реализовать чтение файла с пагинацией (10 сообщений на странице).

Вариант №7. Создать html форму с полями: должность врача, фио врача, даты приема (на каждую должность может быть несколько записей о врачах). Написать скрипт, который сохраняет введенную информацию в файл на сервере,в формате json. На отдельной странице реализовать чтение файла в отформатированном виде.

Вариант №8. Создать html форму с полями: должность врача, фио врача, даты приема (на каждую должность может быть несколько записей о врачах). Написать скрипт, который сохраняет введенную информацию в файл на сервере,в формате xml. На отдельной странице реализовать чтение файла в отформатированном виде.

Вариант №9. Создать html форму с полями: название учебной группы, ФИО студента, адрес проживания студента, дата рождения (студентов в группе может быть до 25 человек). Написать скрипт, который сохраняет введенную информацию в файл на сервере в папке "Students" в формате xml. На отдельной странице реализовать чтение файла в отформатированном виде.

Вариант №10. Создать html форму с полями: название учебной группы, ФИО студента, адрес проживания студента, дата рождения (студентов в группе может быть до 25 человек). Написать скрипт, который сохраняет введенную информацию в файл на сервере в папке "Students" в формате json. На отдельной странице реализовать чтение файла в отформатированном виде.

Лабораторная работа №8. COOKIE Задание

Задание: изучить работу с куки (запись, получение, удаление).

Задание:

Вариант №1. Создать форму для авторизации пользователя: логин и пароль. Сохранять введенные данные в куки. При следующем запуске страницы проверять существую ли куки, если да, то отображать в качестве значений в форме. куки должны храниться в браузере неделю, далее- удаляться.

Вариант №2. Создать форму контактов: имя пользователя и электронная почта. Сохранять введенные данные в куки. При следующем запуске страницы проверять существую ли куки, если да, то отображать в качестве значений в форме. куки должны храниться в браузере один день, далее- удаляться.

Вариант №3. Создать форму контактов: имя фирмы и номер телефона. Сохранять введенные данные в куки. При следующем запуске страницы проверять существую ли куки, если да, то отображать в качестве значений в форме. куки должны храниться в браузере один месяц, далее - удаляться.

Вариант №4. Создать форму поиска данных (одно поле). Сохранять введенные

данные в куки. куки должны храниться в браузере 2 часа, далее - удаляться.

Вариант №5. Создать форму для выбора любимого цвета пользователя из предложенных. Сохранять выбранный цвет в куки. При следующем запуске страницы проверять существую ли куки, если да, то отображать в качестве значений в форме. куки должны храниться в браузере неделю, далее- удаляться.

Вариант №6. Создать форму для выбора пола пользователя. Сохранять выбранный пол в куки. При следующем запуске страницы проверять существую ли куки, если да, то отображать в качестве значений в форме.

Вариант №7. Реализовать скрипт, сохраняющий в куки дату последнего посещения страницы. При открытии страницы выводится информация из куки о последнем посещении.

Вариант №8. Реализовать скрипт, сохраняющий введенные баллы пользователем. В отдельной странице выводится все баллы из куки в виде таблицы.

Вариант №9. Создать форму для ввода города местонахождения пользователя (при условии, если куки пустые). Сохранять введенные данные в куки. При следующем запуске страницы проверять существую ли куки, если да, то отображать сообщение: Вы из города (значение куки). куки должны храниться в браузере один месяц, далее - удаляться.

Вариант №10. Создать форму контактов: имя пользователя, электронная почта и номер телефона. Сохранять введенные данные в куки. При следующем запуске страницы проверять существую ли куки, если да, то отображать в качестве значений в форме. куки должны храниться в браузере одну неделю, далее - удаляться.

Лабораторная работа №9. SESSION Задание

Цель: изучить правила работы с сессиями в PHP.

Задание: создать форму для авторизации пользователя.

Вариант №1-3. Создать три страницы с различным контентом. При авторизации пользователя создавать сессию и при переходе от страницы к странице выводить имя пользователя из массива сессии.

Вариант №4-6. Создать две страницы: административная панель и новостная лента. При авторизации администратора реализовать доступ к обеим страницам, иначе только к новостной ленте. При авторизации сразу создается сессия. Также реализовать кнопку Выход (уничтожение сессии).

Вариант №7-10. Создать страницу, которая размещает "секретный" материал, доступный только для администратора сайта, все остальные пользователи видят стандартную новостную ленту. Функционал реализовать через сессии. Реализовать уничтожение сессии.

Лабораторная работа №10. MySQL

Цель: изучить правила работы с СУБД MySQL, построение SQL запросов, библиотеки MySQLi, PDO.

Задание:

Вариант №1. Создать две таблицы в БД: Лагерь (id, название лагеря, адрес местонахождения, URLкартинки), Заезды (id, id_лагеря, дата начала заезда, дата окончания заезда, название заезда (тематика), стоимость заезда). Реализовать сайт из 3-х страниц: главная, Добавление информации, Калькулятор. Главная станица содержит общую информацию о лагерях и заездах. Добавление информации содержит формы для добавления информации об лагерях и заездах с сохранением в БД (при добавление данных в БД реализуется проверка на наличие таких записей, также реализовать проверку на даты заездов в отдельном лагере (они не должны пересекаться). На странице калькулятор

реализовать подсчет стоимость заезда (пользователь выбирает лагерь, даты заезда, вводит количество детей).

Вариант №2. Создать две таблицы для БД отеля: Номера_отеля(id, номер_комнаты, тип, стоимость, URLкартинки), Заказ (id, id_комнаты, дата начала заезда, дата окончания заезда, количество посетителей, статус (забронировано, свободно, занято)). Реализовать сайт из 3-х страниц: главная, Добавление информации о заказах, Свободные номера. Главная станица содержит общую информацию об отели, типах номеров, внешнего вида и стоимости. Добавление информации содержит формы для добавления информации о бронировании, занятости и освобождении номеров в отеле (при бронировании номера проверяется занятость его на данный период). На странице свободные номера выводится "шахматка" свободных и занятых на текущий день номеров, а также реализована возможность выбора через календарь конкретной даты, а также номера отеля и проверки его статуса.

Вариант №3. Создать две таблицы в БД: Области (id, название, губернатор), города (id, id_области, название города, краткое описание, URLкартинки). Реализовать сайт из 3-х страниц: главная, Добавление информации, Просмотр сведений. Главная содержит общую информацию. Добавление информации содержит формы для добавления информации об областях и городах с сохранением в БД. Просмотр сведений содержит поисковую строку для ввода города и реализует вывод соответствующих данных их таблиц БД.

Вариант №4. Создать две таблицы в БД: Компании (id, название компании, страна), марка машины (id, id_компании, название марки, технические характеристики, год, URLкартинки). Реализовать сайт из 3-х страниц: главная, Добавление информации, Просмотр сведений. Главная содержит общую информацию. Добавление информации содержит формы для добавления информации об компаниях и марках автомобилей с сохранением в БД. Просмотр сведений содержит поисковую строку для ввода названия марки и реализует вывод соответствующих данных их таблиц БД.

Вариант №5. Создать две таблицы в БД: Жанр (id, название жанра),Книга (id, id_жанра, ФИО автора, Название книги, Количество страниц, Издательство, год издания). Реализовать сайт из 3-х страниц: главная, Добавление информации, Просмотр сведений. Главная содержит общую информацию. Добавление информации содержит формы для добавления информации об областях и городах с сохранением в БД. Просмотр сведений содержит поисковую строку для ввода жанра с автодобавлением (AJAX) и реализует вывод соответствующих данных их таблиц БД.

Вариант №6. Создать две таблицы в БД: Автор (id, ФИО автора), Книга (id, id_автора, Название книги, Количество страниц, Издательство, год издания). Реализовать сайт из 3-х страниц: главная, Добавление информации, Просмотр сведений. Главная содержит общую информацию. Добавление информации содержит формы для добавления информации об областях и городах с сохранением в БД. Просмотр сведений содержит поисковую строку для ввода названия книги с автодобавлением (AJAX) и реализует вывод соответствующих данных их таблиц БД.

Вариант №7. Создать две таблицы в БД: Автор (id, ФИО автора), Книга (id, id_автора, Название книги, Количество страниц, Издательство, год издания). Реализовать сайт из 3-х страниц: главная, Добавление информации, Просмотр сведений. Главная содержит общую информацию. Добавление информации содержит формы для добавления информации об областях и городах с сохранением в БД. Просмотр сведений содержит таблицу с данными из 2 таблиц с возможностью фильтрации по автору и году издания.

Вариант №8. Создать две таблицы для БД Интернет магазина: Справочник продукции(id, название продукции, единица измерения, количество, стоимость, URLкартинки), Заказы (id, id_продукции, ФИО заказчика, адрес доставки, количество продукции). Реализовать сайт из 3-х страниц: главная, Добавление информации о поступив-

шей продукции, Корзина. Главная станица содержит общую информацию о продукции Интернет-магазина (название, изображение, цена). Добавление информации содержит формы для добавления новой продукции, а также изменение количества поступивших товаров. На странице Корзина предоставляется возможность сделать заказ на отдельный товар, указать количество (осуществить проверку на наличие заданного количества товара) с подсчетом общей суммы заказа. При подтверждении заказа количество товаров в справочнике продукции должно уменьшиться.

Вариант №9. Создать две таблицы в БД: Жанр (id, название жанра), Журнал (id, id_жанра, Название журнала, Номер журнала, Количество страниц, Издательство, год издания). Реализовать сайт из 3-х страниц: главная, Добавление информации, Просмотр сведений. Главная содержит общую информацию. Добавление информации содержит формы для добавления информации о новых журналах и с выбором из выпадающего списка жанра и сохранением в БД. Просмотр сведений содержит поисковую строку для ввода названия журнала с автодобавлением (AJAX) и реализует вывод соответствующих данных их таблиц БД в форматированном виде.

Вариант №10. Создать две таблицы в БД: Аппаратное обеспечение (id, название компании, страна, название устройства, марка/версия устройства, цена, год выпуска), Отзыв (id, id_аппаратного обеспечения, имя пользователя, оценка (от 1 до 5), текст отзыва). Реализовать сайт из 3-х страниц: главная, Добавление отзыва, Просмотр отзывов. Главная содержит общую информацию об аппаратном обеспечении. Добавление отзыва содержит форму с выбором АО для которого пишется отзыв (с фильтрами: по компании, по названию, по марке), полями указания количества звезд/оценки, полем ввода имени пользователя и многострочным текстом для отзыва. Просмотр сведений содержит отзывы по АО с возможностью выбора страны, компании, названия и марки устройства, сортировки по цене (по возрастанию и убыванию), сортировки по оценке (звездам) (по возрастанию и убыванию).

Приложение Б

Проектное задание

Задание: Спроектировать и разработать Интернет-сервис по заданной тематике.

Основной функционал: наличие сессии, поиск информации, формы ввода и вывода данных, хранение данных в БД.

Дополнительный (не менее 3 дополнительных функций, разрешается внедрять альтернативный функционал в соответствии с предметной областью, учитывая требования полезности и UX/UI):

- отображение данных в отсортированном виде с наличием не менее 3 параметров сортировки;
- проверка на безопасность данных, передаваемых с форм на сервер;
- шифрование идентификационных данных на сервере;
- пагинация отображения данных;
- поиск с подсказками (автодополнение);
- сохранение кукки;
- панель администратора;
- др.

Вариант №1 Сервис по бронированию билетов в кинотеатр.

Вариант №2. Сервис записи на услугу в парикмахерскую.

Вариант №3. Сервис записи на услугу в автосервис.

Вариант №4. Сервис поиска репетитора и отправка репетитору через сервис сообщения.

Вариант №5. Сервис записи к врачу.

Вариант №6. Сервис покупки авиабилетов.

Вариант №7. Сервис поиска помещений для аренды (с возможностью размещения новых арендных мест).

Вариант №8. Сервис знакомств.

Вариант №9. Сервис для букинистов.

Вариант №10. Сервис для проведения лотереи.

Вариант №11. Мобильный сервис "Чистый город" (задание в рамках сотрудничества с Кванториумом)

Примечание. На первом этапе сформировать требования к сервису в текстовом файле: Функционал, структура данных, мокап страниц сервиса. Проект к ТЗ согласовать с преподавателем и выложить на портал. Второй этап - реализация сервиса. Третий этап - тестирование.