

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Аксенов Сергей Леонидович

Должность: Ректор

Дата подписания: 25.08.2021 09:15

Идентификатор ключа:

159e22ec4edaa8a694913d5c08c0b6671130587da9e1acf845343ffaf5ad101e

автономная некоммерческая образовательная организация  
высшего образования  
«Региональный финансово-экономический институт»

Кафедра экономики и управления



Утверждаю  
Декан экономического факультета  
Ю.И. Петренко  
«29» мая 2020 г.

## Рабочая программа дисциплины «РАЗРАБОТКА ТРЕБОВАНИЙ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ»

Направление подготовки **38.03.05 Бизнес-информатика**  
Профиль **Информационный бизнес**  
Квалификация (степень) **Бакалавр**

Факультет **экономический**  
Заочная форма обучения



Курск 2020

Рецензенты:

Бутова Вера Николаевна, к.п.н., доцент кафедры экономики и управления;

Мордовин Аркадий Владленович, кандидат экономических наук, доцент кафедры экономики и управления

Рабочая программа составлена в соответствии с Федеральным государственным образовательным стандартом высшего профессионального образования по направлению подготовки 38.03.05 Бизнес-информатика, утвержденного приказом Министерства образования и науки Российской Федерации от от 11 августа 2016 г. N 1002.

Рабочая программа предназначена для методического обеспечения дисциплины образовательной программы 38.03.05 Бизнес-информатика.

«29» мая 2020 г.

Составитель:



Смецкой А.С., ст.преподаватель  
кафедры экономики и управления

© Смецкой А.С., 2020

© Региональный финансово-экономический институт, 2020

**Лист согласования рабочей программы  
дисциплины «Разработка требований к  
программному обеспечению»**

Направление подготовки 38.03.05: **Бизнес-информатика**

Профиль: **Информационный бизнес**

Квалификация (степень): **Бакалавр**

Факультет экономический

Заочная форма обучения

2020/2021 учебный год

Рабочая программа утверждена на заседании кафедры экономики и управления, протокол № 8 от «29» мая 2020 г.

Зав. кафедрой \_\_\_\_\_ С.Л. Аксенов

Составитель: \_\_\_\_\_ А.С. Смецкой

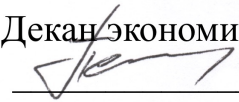
**Согласовано:**

Начальник УМУ \_\_\_\_\_ О.И. Петренко, «29» мая 2020 г.

Библиотекарь \_\_\_\_\_ Т.А. Котельникова, «29» мая 2020 г.


Председатель методической комиссии по профилю \_\_\_\_\_ В.Н. Бутова, «29» мая 2020 г.

**Изменения в рабочей программе  
дисциплины «Разработка требований к программному обеспечению»  
на 2021-2022 уч.год**

Утверждаю  
Декан экономического факультета  
  
Ю.И. Петренко  
«25» августа 2021 г.


В рабочую программу вносятся следующие изменения:  
1) внесены изменения в список основной литературы к экзамену.

Рабочая программа утверждена на заседании кафедры экономики и управления, протокол № 1 от «25» августа 2021 г.

Зав. кафедрой  С.Л. Аксенов

**Согласовано:**

Начальник УМУ  
  
О.И. Петренко, «25» августа 2021 г.

Председатель методической комиссии по профилю  
  
В.Н. Бутова, «25» августа 2021 г.

## СОДЕРЖАНИЕ

<b>ПОЯСНИТЕЛЬНАЯ ЗАПИСКА .....</b>	<b>5</b>
<b>1. Цель и задачи изучения дисциплины .....</b>	<b>5</b>
<b>2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы .....</b>	<b>5</b>
<b>3. Место дисциплины в структуре ООП .....</b>	<b>6</b>
<b>СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....</b>	<b>7</b>
<b>4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий.....</b>	<b>7</b>
<b>ПРАКТИЧЕСКИЕ ЗАНЯТИЯ.....</b>	<b>13</b>
<b>ЛАБОРАТОРНЫЕ ЗАНЯТИЯ .....</b>	<b>24</b>
<b>5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю) .....</b>	<b>29</b>
<b>6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю) .....</b>	<b>29</b>
<b>7. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля).....</b>	<b>37</b>
<b>8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), необходимых для освоения дисциплины (модуля).....</b>	<b>38</b>
<b>9. Методические указания для обучающихся по освоению дисциплины (модулю) .....</b>	<b>39</b>
<b>10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем.....</b>	<b>57</b>
<b>11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).....</b>	<b>58</b>
<b>ПРИЛОЖЕНИЯ .....</b>	<b>59</b>

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

### 1. Цель и задачи изучения дисциплины

Целью освоения дисциплины «Разработка web-приложений» является знакомство студентов с базовыми концепциями и приемами Web-программирования, получение представления о современных Web-технологиях, о подходах к проектированию, разработке, отладке, оптимизации и развертыванию web-приложений с динамичным контентом.

В ходе достижения цели необходимо решить следующий круг задач:

1. знакомство с историей и тенденциями развития web-технологий; приобретение теоретических знаний и практических навыков в работе с современными продуктами Web-программирования: PHP, JavaScript, MySQL.

### 2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих общекультурных и профессиональных компетенций:

1. культурой мышления, обобщения и анализа, искусством определения целей исследуемой системы и выбора эффективного пути их достижения (ОК-1);
2. способностью анализировать социально-значимые проблемы и процессы, происходящие в обществе, и прогнозировать возможное их развитие в будущем (ОК-4);
3. умением находить организационно-управленческие решения и готовностью нести за них ответственность (ОК-8);
4. осознанием сущности и значения информации в развитии современного общества, владением основными методами, способами и средствами ее получения, хранения и переработки (ОК-12);
5. выбирать рациональные ИС и ИКТ-решения для управления бизнесом (ПК-3);
6. проводить анализ инноваций в экономике, управлении и ИКТ (ПК-4);
7. использовать современные стандарты и методики, разрабатывать регламенты деятельности предприятия (ПК-8);
8. позиционировать электронное предприятие на глобальном рынке; формировать потребительскую аудиторию и осуществлять взаимодействие с потребителями, организовывать продажи в среде Интернет (ПК-11);
9. проектировать и внедрять компоненты ИТ-инфраструктуры предприятия, обеспечивающие достижение стратегических целей и поддержку бизнес-процессов (ПК-15);

- 10.разрабатывать контент и ИТ-сервисы для предприятия и Интернет-ресурсов (ПК-18);
- 11.консультировать заказчиков по рациональному выбору ИС и ИКТ управления бизнесом (ПК-24).

В результате изучения дисциплины «Разработка WEB-приложений» студент должен:

Знать:

- 12.основные принципы функционирования Internet (З-1);
- 13.современные сервисы и службы глобальной сети Internet (З-2);
- 14.основные технологические этапы проектирования сайтов (З-3);
- 15.базовые концепции и приемы Web-программирования (З-4);
- 16.методы создания Web-страниц и их оптимизации (З-5);
- 17.инструментальные средствами создания Web-сайтов. (З-6).

Уметь:

- 18.находить необходимую информацию в глобальной сети Internet, подготавливать информацию для её дальнейшего размещения в Internet (У-1);
- 19.программировать Web-сайт с использованием технологий PHP, JavaScript, MySQL, Ajax (У-2);
- 20.проектировать и реализовывать структуру Web-сайта. (У-3).

Владеть:

навыками использования изученных технологий (PHP, JavaScript, MySQL, Ajax) и возможного их объединения при работе над конкретным Web-приложением (В-1).

Соотнесение результатов обучения по дисциплине с планируемыми результатами освоения образовательной программы представлено в таблице, Приложение 1.

### **3. Место дисциплины в структуре ООП**

Дисциплина включена в вариативную часть профессионального цикла ООП и является обязательной дисциплиной.

«Разработка web-приложений», относятся знания, умения и навыки, сформированные в ходе изучения предметов профессионального цикла: «Вычислительные системы, сети, телекоммуникации», «Программирование», «WEB-публицистика», «Современные операционные системы».

Изучение данной дисциплины является основой для последующего изучения дисциплин профессионального цикл: «Электронный бизнес», «Оптимизация и продвижение сайтов в Интернет», «Программирование расширений web-браузеров и web-программирование».

## СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий

Схема распределения учебного времени по видам учебной работы

Общая трудоемкость дисциплины при заочной форме обучения – 3 зачетные единицы (108 академических часов)

#### Схема распределения учебного времени по семестрам

Виды учебной работы	Трудоемкость, час	
	4 курс	Всего:
<b>Общая трудоемкость</b>	108	108
<b>Аудиторная работа</b>	12	12
в том числе:		
лекции	4	4
практические занятия	4	4
лабораторные работы	4	4
<b>Самостоятельная работа</b>	92	92
<b>Промежуточная аттестация</b>	зачет	4

### Тематический план

№ п/п	Разделы и темы дисциплины	Общая трудоемкость, час	В том числе аудиторных			Самостоятельная работа	Промежуточная аттестация	
			всего	из них:				
				лекц	лабор			практ
1	Динамическое содержание веб-страницы.	15	1		1	14		
2	PHP: основы, управлением процессом выполнения программ, функции и объекты, массивы.	16	3	1	1	1	13	
3	MySQL: основные характеристики, доступ к MySQL с использованием PHP.	15	2	1	1		13	
4	JavaScript: управлением процессом выполнения программ, функции и объекты, массивы.	15	2	1		1	13	
5	Проверка данных и обработка ошибок в JavaScript и PHP.	14	1		1		13	
6	Технология Ajax.	14	1	1			13	



7	Объединение технологий в разработке web-приложений.	15	2		1	1	13	
	Промежуточная аттестация (зачет)	4						+
	Итого	<b>108</b>	<b>12</b>	<b>4</b>	<b>4</b>	<b>4</b>	<b>92</b>	<b>4</b>

## Структура и содержание дисциплины

### Тема 1. Динамическое содержание веб-страницы.

Введение в динамическое содержимое веб-страницы. HTTP и HTML: основы, заложенные Бернерсом-Ли. Процедура «запрос — ответ». Преимущества использования PHP, MySQL и JavaScript. Использование PHP. Использование MySQL. Использование JavaScript. Веб-сервер Apache. Несколько слов о программах с открытым исходным кодом.

Литература:

Основная – 1; 2; 4.

Дополнительная – 1; 2.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ПК-3; ПК-4.

Образовательные результаты: З-1; З-2; В-1.

### Тема 2. PHP: основы, управлением процессом выполнения программ, функции и объекты, массивы.

Включение PHP в HTML. Вызов парсера PHP. Структура PHP. Использование комментариев. Основной синтаксис. Функции. Область видимости переменной. Локальные переменные. Глобальные переменные. Статические переменные. Суперглобальные переменные. Суперглобальные переменные и проблемы безопасности.

Выражения и управление процессом выполнения программы в PHP. Альтернативный синтаксис. Динамическое связывание в PHP.

Функции и объекты PHP. Совместимость версий PHP . Объявление класса. Создание объекта. Доступ к объектам. Клонирование объектов. Конструкторы. Деструкторы в PHP 5. Написание методов. Статические методы в PHP 5 . Наследование.

Массивы в PHP. Многомерные массивы. Использование функций для работы с массивами.

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Википедия [Электронный ресурс]: [свобод. Интернет-энцикл.] – Электрон. дан. и прогр. – Режим доступа: <http://ru.wikipedia.org>, свободный. – Русскояз. часть междунар. проекта «Википедия».
2. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ – <http://it.rfei.ru/~1r>.
3. Интернет ресурсы о PHP из списка: 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ПК-3; ПК-4; ПК 15.

Образовательные результаты: З-1; З-2; У-1; В-1.

### **Тема 3. MySQL: основные характеристики, доступ к MySQL с использованием PHP.**

Основные характеристики MySQL. Сводка понятий, используемых в базах данных. Команды MySQL. Создание базы данных. Организация доступа пользователей. Создание таблицы.

Типы данных: CHAR, BINARY, TEXT и VARCHAR, BLOB. Числовые типы данных: DATE и TIME, AUTO\_INCREMENT.

Работа с таблицами. Использование логических операторов. Функции MySQL. Работа с MySQL через phpMyAdmin . Использование phpMyAdmin. Доступ к MySQL с использованием PHP. Запросы к базе данных MySQL с помощью PHP. Использование свойства AUTO\_INCREMENT. Использование идентификаторов вставленных строк. Выполнение дополнительных запросов. Предотвращение внедрения SQL-кода. Использование указателей места заполнения. Предотвращение внедрения HTML-кода.

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
2. Интернет ресурсы о MySQL из списка: 18, 19, 20, 21, 22, 23, 24.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная

лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-14.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

#### **Тема 4. JavaScript: управлением процессом выполнения программ, функции и объекты, массивы.**

JavaScript и текст HTML. Использование сценариев в заголовке документа. Устаревшие и нестандартные браузеры. Включение файлов JavaScript. Отладка кода JavaScript.

Выражения и управление процессом выполнения сценариев в JavaScript.

Функции, объекты и массивы JavaScript.

Массивы в JavaScript. Числовые массивы. Ассоциативные массивы. Многомерные массивы. Использование методов массивов: concat, forEach: для браузеров не из семейства IE и кросс-браузерное решение, join, push и pop, reverse, sort.

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
2. Интернет ресурсы о JavaScript из списка: 25, 26, 27, 28, 29, 30, 31, 32.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

#### **Тема 5. Проверка данных и обработка ошибок в JavaScript и PHP.**

Проверка данных, введенных пользователем, средствами JavaScript . Документ validate.html Регулярные выражения. Соответствие, закладываемое в метасимволы. Нестрогое символьное соответствие. Группировка с помощью скобок. Символьный класс. Указание диапазона. Инвертирование. Ряд более сложных примеров. Сводная таблица метасимволов. Общие модификаторы. Использование регулярных выражений в JavaScript. Использование регулярных выражений в PHP. Повторное отображение формы после проверки данных PHP-программой .

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11; ПК-15; ПК-22

Образовательные результаты: З-1; З-2; З-3; У-1; В-1; В-2.

### **Тема 6. Технология Ajax.**

Что такое Ajax? Использование XMLHttpRequest. Свойство readyState. Серверная половина Ajax-процесса. Использование GET вместо POST. Отправка XML-запросов. Несколько слов о XML. Использование YUI для работы с технологией Ajax. YUI-метод asyncRequest. Другие применения YUI.

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
2. Интернет ресурсы о Ajax из списка: 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

### **Тема 7. Объединение технологий в разработке web-приложений.**

Объединение технологий. Проектирование сайта социальной сети. О дополнениях сторонних производителей. Информация на веб-сайте. Файл `rnfunctions.php`

Функции. Файл `rnheader.php`. Файл `rnsetup.php`. Файл `index.php`. Файл `rnsignup.php`. Проверка возможности использования желаемого имени пользователя. Файл `rnsignup.php` (YUI-версия). Файл `rncheckuser.php`. Файл `rnlogin.php`. Файл `rnprofile.php`. Добавление текста в поле About Me («Обо

мне»). Добавление изображения профиля. Обработка изображения. Отображение текущего профиля. Файл `rnmembers.php`. Файл `rnfriends.php`. Файл `rnmessages.php`. Файл `rnlogout.php`

#### Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-1

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

## **Практические занятия**

### **Практическое занятие №1 «Динамическое содержание веб-страницы»**

Цель: Ознакомить студента с основными тенденциями веб-программирования, формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Основы HTTP и HTML
2. Процедура «запрос — ответ»
3. Преимущества использования PHP, MySQL и JavaScript
4. Использование PHP
5. Использование MySQL
6. Использование JavaScript
7. Веб-сервер Apache
8. Программы с открытым исходным кодом.

Литература:

Основная – 1; 2; 4.

Дополнительная – 1; 2.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ПК-3; ПК-4.

Образовательные результаты: З-1; З-2; В-1.

Форма контроля: групповая дискуссия, опрос.

### **Практическое занятие №2 «PHP: основы, управление процессом выполнения программ, функции и объекты, массивы»**

Цель: Ознакомить студента с основами работы с PHP; формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Включение PHP в HTML
2. Вызов парсера PHP
3. Структура PHP
4. Использование комментариев
5. Основной синтаксис
6. Осмысление переменных
7. Строковые переменные

8. Числовые переменные
9. Массивы
10. Двумерные массивы
11. Правила присваивания имен переменным
12. Операторы
13. Арифметические операторы
14. Операторы присваивания
15. Операторы сравнения
16. Логические операторы
17. Присваивание значений переменным
18. Увеличение и уменьшение значения переменной на единицу  
Объединение строк
19. Типы строк
20. Изменение предназначения символов
21. Многострочные команды
22. Типы переменных
23. Константы
24. Предопределенные константы
25. Различие между командами echo и print
26. Функции
27. Область видимости переменной
28. Локальные переменные
29. Глобальные переменные
30. Статические переменные
31. Суперглобальные переменные
32. Суперглобальные переменные и проблемы безопасности
33. Выражения
34. Литералы и переменные
35. Операторы
36. Приоритетность операторов
37. Взаимосвязанность операторов
38. Операторы отношения
39. Равенство
40. Операторы сравнения
41. Логические операторы
42. Условия
43. Инструкция if
44. Инструкция else

45. Инструкция `elseif`
46. Инструкция `switch`
47. Прекращение работы инструкции `switch`
48. Действие по умолчанию
49. Альтернативный синтаксис
50. Оператор «?»
51. Организация циклов
52. Циклы `while`
53. Циклы `do...while`
54. Циклы `for`
55. Прекращение работы цикла
56. Инструкция `continue`
57. Неявное и явное преобразование типов
58. Динамическое связывание в PHP
59. Функции PHP
60. Определение функции
61. Возвращение значения
62. Возвращение массива
63. Передача по ссылке
64. Возвращение глобальных переменных
65. И еще раз об области видимости переменных
66. Включение и запрос файлов
67. Инструкция `include`
68. Использование инструкции `include_once`
69. Использование инструкций `require` и `require_once`
70. Совместимость версий PHP
71. Объекты PHP
72. Терминология
73. Объявление класса
74. Создание объекта
75. Доступ к объектам
76. Клонирование объектов
77. Конструкторы
78. Деструкторы в PHP 5
79. Написание методов
80. Статические методы в PHP 5
81. Объявление свойств
82. Объявление констант



- 83. Область видимости свойств и методов в PHP 5
- 84. Статические свойства и методы
- 85. Наследование
- 86. Инструкция parent
- 87. Конструкторы подкласса
- 88. Методы Final

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Википедия [Электронный ресурс]: [свобод. Интернет-энцикл.] – Электрон. дан. и прогр. – Режим доступа: <http://ru.wikipedia.org>, свободный. – Русскояз. часть междунар. проекта «Википедия».
2. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ – <http://it.rfei.ru/~1r>.
3. Интернет ресурсы о PHP из списка: 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ПК-3; ПК-4; ПК 15.

Образовательные результаты: З-1; З-2; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, сравнительный анализ.

### **Практическое задание №3 «MySQL: основные характеристики, доступ к MySQL с использованием PHP»**

Цель: Дать теоретические знания и практические навыки работы с MySQL; формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Основные характеристики MySQL
2. Сводка понятий, используемых в базах данных
3. Доступ к MySQL из командной строки
4. Начало работы с интерфейсом командной строки
5. Использование интерфейса командной строки
6. Команды MySQL
7. Создание базы данных
8. Организация доступа пользователей
9. Создание таблицы
10. Типы данных

11. Тип данных CHAR
12. Тип данных BINARY
13. Типы данных TEXT и VARCHAR
14. Тип данных BLOB
15. Числовые типы данных
16. Типы данных DATE и TIME
17. Тип данных AUTO\_INCREMENT
18. Добавление данных к таблице
19. Переименование таблиц
20. Изменение типа данных столбца
21. Добавление нового столбца
22. Переименование столбца
23. Удаление столбца
24. Удаление таблицы
25. Индексы
26. Создание индекса
27. Использование команды CREATE INDEX
28. Добавление индексов при создании таблиц
29. Первичные ключи
30. Создание индекса FULLTEXT
31. Создание запросов к базе данных MySQL
32. SELECT
33. SELECT COUNT
34. SELECT DISTINCT
35. DELETE
36. WHERE
37. LIMIT
38. MATCH...AGAINST
39. MATCH...AGAINST... IN BOOLEAN MODE
40. UPDATE...SET
41. ORDER BY
42. GROUP BY
43. Объединение таблиц
44. NATURAL JOIN
45. JOIN...ON
46. Использование ключевого слова AS
47. Использование логических операторов
48. Функции MySQL
49. Работа с MySQL через phpMyAdmin
50. Использование phpMyAdmin
51. Проектирование базы данных
52. Первичные ключи: ключи к реляционным базам данных
53. Нормализация

54. Первая нормальная форма
55. Вторая нормальная форма
56. Третья нормальная форма
57. Когда не следует проводить нормализацию
58. Отношения
59. «Один к одному»
60. «Один ко многим»
61. «Многие ко многим»
62. Базы данных и анонимность
63. Транзакции
64. Ядра (механизмы хранения) транзакций
65. Использование команды BEGIN
66. Использование команды COMMIT
67. Использование команды ROLLBACK
68. Использование команды EXPLAIN
69. Резервное копирование и восстановление данных
70. Использование команды mysqldump
71. Создание файла резервной копии
72. Восстановление данных из файла резервной копии
73. Выгрузка данных в файлы формата CSV
74. Планирование резервного копирования

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

- Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
- Интернет ресурсы о MySQL из списка: 18, 19, 20, 21, 22, 23, 24.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-14.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос.

#### **Практическое занятие №4 «JavaScript: управлением процессом выполнения программ, функции и объекты, массивы»**

Цель: предоставление теоретических знаний и практических навыков студенту по работе с технологией JavaScript, формирование общекультурных и профессиональных компетенций.

**Вопросы для обсуждения:**

1. JavaScript и текст HTML
2. Использование сценариев в заголовке документа
3. Устаревшие и нестандартные браузеры
4. Включение файлов JavaScript
5. Отладка кода JavaScript
6. Использование комментариев
7. Точка с запятой
8. Переменные
9. Строковые переменные
10. Числовые переменные
11. Массивы
12. Операторы
13. Арифметические операторы
14. Операторы присваивания
15. Операторы сравнения
16. Логические операторы
17. Инкремент и декремент переменной
18. Объединение строк
19. Управляющие символы
20. Типизация переменных
21. Функции
22. Глобальные переменные
23. Локальные переменные
24. Объектная модель документа
25. Несовместимость браузеров
26. Использование DOM
27. Выражения и управление процессом выполнения сценариев в JavaScript
28. Функции JavaScript
29. Определение функции
30. Массив аргументов
31. Возвращение значения
32. Возвращение массива
33. Объекты JavaScript
34. Объявление класса
35. Создание объекта
36. Доступ к объектам
37. Ключевое слово prototype
38. Статические методы и свойства
39. Расширение объектов JavaScript
40. Массивы в JavaScript
41. Числовые массивы
42. Ассоциативные массивы

- 43. Многомерные массивы
- 44. Использование методов массивов
- 45. Метод concat
- 46. Метод forEach: для браузеров не из семейства IE
- 47. Метод forEach: кросс-браузерное решение
- 48. Метод join
- 49. Методы push и pop
- 50. Использование метода reverse
- 51. Метод sort

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;

Интернет-ресурс:

- Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
- Интернет ресурсы о JavaScript из списка: 25, 26, 27, 28, 29, 30, 31, 32.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, сравнительный анализ.

### **Практическое занятие №5 «Проверка данных и обработка ошибок в JavaScript и PHP»**

Цель: предоставление теоретических знаний и практических навыков студенту по проверке данных и обработке ошибок в JavaScript и PHP, формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Проверка данных, введенных пользователем, средствами JavaScript
2. Документ validate.html Регулярные выражения
3. Соответствие, закладываемое в метасимволы
4. Нестрогое символьное соответствие
5. Группировка с помощью скобок
6. Символьный класс
7. Указание диапазона
8. Инвертирование
9. Ряд более сложных примеров
10. Сводная таблица метасимволов
11. Общие модификаторы
12. Использование регулярных выражений в JavaScript

13.Использование регулярных выражений в PHP

14.Повторное отображение формы после проверки данных PHP-программой.

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

- Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11; ПК-15; ПК-22

Образовательные результаты: З-1; З-2; З-3; У-1; В-1; В-2.

Форма контроля: групповая дискуссия, опрос, сравнительный анализ.

### **Практическое занятие №6 «Использование технологии Ajax»**

Цель: предоставление теоретических знаний и практических навыков по работе с технологией Ajax, формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

- Использование XMLHttpRequest
- Ваша первая Ajax-программа
- Свойство readyState
- Серверная половина Ajax-процесса
- Использование GET вместо POST
- Отправка XML-запросов
- Несколько слов о XML
- А зачем вообще использовать XML?
- Использование YUI для Ajax и не только
- Выбор среды разработки
- Использование YUI
- Сжатые версии
- Использование YUI для работы с технологией Ajax
- Включение файлов среды
- YUI-метод asyncRequest
- Пример Ajax GET-запроса с использованием YUI
- Пример Ajax XML, использующий YUI
- Другие применения YUI

- Простой YUI-календарь

#### Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

- Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
- Интернет ресурсы о Ajax из списка: 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, сравнительный анализ, выполнение практических заданий.

### **Практическое занятие №7 «Объединение технологий»**

Цель: предоставление теоретических знаний и практических навыков по работе с изученными технологиями, формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Проектирование сайта социальной сети
2. О дополнениях сторонних производителей
3. Информация на веб-сайте
4. Файл `rnfunctions.php`
5. Функции
6. Файл `rnheader.php`
7. Файл `rnsetup.php`
8. Файл `index.php`
9. Файл `rnsignup.php`
10. Проверка возможности использования желаемого имени пользователя
11. Файл `rnsignup.php` (YUI-версия)
12. Файл `rncheckuser.php`
13. Файл `rnlogin.php`
14. Файл `rnprofile.php`
15. Добавление текста в поле About Me («Обо мне»)
16. Добавление изображения профиля
17. Обработка изображения
18. Отображение текущего профиля

- 19. Файл `rnmembers.php`
- 20. Файл `rnfriends.php`
- 21. Файл `rnmessages.php`
- 22. Файл `rnlogout.php`

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, сравнительный анализ, выполнение практических заданий.



## **ЛАБОРАТОРНЫЕ ЗАНЯТИЯ**

### **Тема №1 «Динамическое содержание веб-страницы»**

#### **Лабораторная работа №1 «Установка сервера, предназначенного для разработки»**

Цель: произвести установку и настройку сервера для разработки (для последующего освоения технологий (PHP, MySQL и JavaScript), формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Что такое WAMP, MAMP или LAMP?
2. Установка WAMP на систему Windows
3. Решение проблем установки —
4. Тестирование установки
5. Другие системы WAMP
6. Установка MAMP на систему Mac OS X
7. Ряд окончательных настроек
8. Другие варианты
9. Установка LAMP на Linux
10. Работа в удаленном режиме
11. Вход в систему
12. Использование FTP
13. Использование редактора программ
14. Использование IDE

Литература:

Основная – 1; 2; 4.

Дополнительная – 1; 2.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ПК-3; ПК-4.

Образовательные результаты: З-1; З-2; В-1.

Форма контроля: групповая дискуссия, опрос.

### **Тема №2 «PHP: основы, управление процессом выполнения программ, функции и объекты, массивы»**

#### **Лабораторная работа №2-3 «Практикум по программированию PHP»**

Цель: Ознакомить студента с основами работы с PHP; формирование

общекультурных и профессиональных компетенций.

Вопросы для обсуждения:

1. Использование функции printf
2. Настройка представления данных
3. Дополнение строк
4. Использование функции sprintf
5. Функции даты и времени
6. Константы, связанные с датами
7. Использование функции checkdate
8. Работа с файлами
9. Проверка существования файла
10. Создание файла
11. Чтение из файлов
12. Копирование файлов
13. Перемещение файла
14. Удаление файла
15. Обновление файлов
16. Блокирование файлов при коллективном доступе
17. Чтение всего файла целиком
18. Загрузка файлов на веб-сервер
19. Использование массива \$\_FILES
20. Проверка допустимости
21. Системные вызовы
22. XHTML
23. Преимущества XHTML
24. Версии XHTML
25. В чем же заключаются отличия?
26. Типы документов HTML 4.01
27. Типы документов XHTML 1.0
28. Проверка правильности XHTML

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Википедия [Электронный ресурс]: [свобод. Интернет-энцикл.] – Электрон. дан. и прогр. – Режим доступа: <http://ru.wikipedia.org>, свободный. – Русскояз. часть междунар. проекта «Википедия».
2. Учебный курс для студентов направления подготовки «Бизнес-

информатика» РФЭИ – <http://it.rfei.ru/~1r>.

- Интернет ресурсы о PHP из списка: 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ПК-3; ПК-4; ПК 15.

Образовательные результаты: З-1; З-2; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, сравнительный анализ, выполнение практических заданий.

### **Тема №4 «MySQL: основные характеристики, доступ к MySQL с использованием PHP»**

#### **Лабораторная работа №4 «Обработка форм»**

Цель: Дать теоретические знания и практические навыки работы с MySQL; формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

- Создание форм
- Извлечение отправленных данных
- register\_globals: склонность к использованию устаревших решений
- Значения по умолчанию
- Типы элементов ввода данных
- Текстовое поле
- Текстовая область
- Флажки
- Переключатели
- Скрытые поля
- Поле со списком
- Теги label
- Обезвреживание введенных данных

Основная – 1; 3.

Дополнительная – 1; 2; 3.

Интернет-ресурс:

- Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
- Интернет ресурсы о MySQL из списка: 18, 19, 20, 21, 22, 23, 24.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-14.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, выполнение практических заданий.

### **Лабораторная работа №5 «Работа с шаблонами в системе Smarty»**

Цель: Дать теоретические знания и практические навыки работы с MySQL; формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Для чего используется Smarty?
2. Установка
3. Создание сценариев
4. Создание шаблонов
5. Практический пример

Основная – 1; 3.

Дополнительная – 1; 2; 3.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
2. Интернет ресурсы о MySQL из списка: 18, 19, 20, 21, 22, 23, 24.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-14.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, выполнение практических заданий.

### **Лабораторная работа №6 «Cookie, сессии и аутентификации.»**

Цель: Дать теоретические знания и практические навыки работы с MySQL; формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

6. Почему именно Smarty?
7. Установка
8. Создание сценариев
9. Создание шаблонов
10. Практический пример

Основная – 1; 3.

Дополнительная – 1; 2; 3.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
2. Интернет ресурсы о MySQL из списка: 18, 19, 20, 21, 22, 23, 24.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-14.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, выполнение практических заданий.

### **Тема №7 «JavaScript: управлением процессом выполнения программ, функции и объекты, массивы»**

#### **Лабораторная работа №6**

Цель: предоставление теоретических знаний и практических навыков студенту по работе с технологией JavaScript, формирование общекультурных и профессиональных компетенций.

#### **Вопросы для обсуждения:**

1. Использование комментариев
2. Точка с запятой
3. Переменные
4. Строковые переменные
5. Числовые переменные
6. Массивы
7. Операторы
8. Арифметические операторы
9. Операторы присваивания
10. Операторы сравнения
11. Логические операторы
12. Инкремент и декремент переменной
13. Объединение строк
14. Управляющие символы
15. Типизация переменных
16. Функции
17. Глобальные переменные
18. Локальные переменные
19. Объектная модель документа
20. Несовместимость браузеров

## 21. Использование DOM

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.
2. Интернет ресурсы о JavaScript из списка: 25, 26, 27, 28, 29, 30, 31, 32.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: групповая дискуссия, опрос, сравнительный анализ, выполнение практических заданий.

### 5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Самостоятельная работа студентов нацелена на углубление практических навыков по использованию технологий PHP, MySQL, JavaScript, Ajax.

#### Задания для самостоятельной работы:

1. Доступ к MySQL с использованием PHP

#### Вставка и удаление данных с помощью программы *sqltest.php*

```
<?php // sqltest.php
require_once 'login.php';
$db_server = mysql_connect($db_hostname, $db_username, $db_password);

if (!$db_server) die("Невозможно подключиться к MySQL: " . mysql_error());
mysql_select_db($db_database, $db_server)
    or die("Невозможно выбрать базу данных: " . mysql_error());

if (isset($_POST['author']) &&
    isset($_POST['title']) &&
    isset($_POST['category']) &&
    isset($_POST['year']) &&
    isset($_POST['isbn']))
{
    $author = get_post('author');
```

```

$title = get_post('title');
$category = get_post('category');
$year = get_post('year');
$isbn = get_post('isbn');

if (isset($_POST[•delete1]) && $isbn != "")
{
$query = "DELETE FROM classics WHERE isbn=,$isbn,";

if (!mysql_query($query, $db_server))
echo "Сбой при удалении данных: $query<br />"
mysql_error() . "<br /><br />";
}
else
{
$query = "INSERT INTO classics VALUES"
"('$author'. '$title'. '$category', '$year', '$isbn')";

if (!mysql_query($query, $db_server))
echo "Сбой при вставке данных: $query<br />"
mysql_error() . "<br /><br />";
}
}
echo <<< _END
<form action="sqltest.php" method="post"><pre>
  Author <input type="text" name="author" />
  Title <input type="text" name="title" />
  Category <input type="text" name="category" />
  Year <input type="text" name="year" />
  ISBN <input type="text" name="isbn" />
  input type="submit" value="ADD RECORD" /> // кнопка ДОБАВИТЬ
  ЗАПИСЬ </pre></form>
_END:

$query = "SELECT * FROM classics";
$result = mysql_query($query);

if (!$result) die ("Сбой при доступе к базе данных: " . mysql_error());
$rows = mysql_num_rows($result);
for ($j = 0 : $j < $rows : ++$j)
{
$row = mysql_fetch_row($result);
echo <<< _END

```

```

<pre>
  Author $row[0]
  Title $row[1]
  Category $row[2]
  Year $row[3]
  ISBN $row[4]
</pre>
<form action="sqltest.php" method="post">
<input type="hidden" name="delete" value="yes" />
<input type="hidden" name="isbn" value="$row[4]" />
<input type="submit" value="HDELETE RECORD" /></form> // кнопка УДАЛИТЬ
ЗАПИСЬ _END;
}

mysql_c.ose($db_server);
function get_post($var)
{
  return mysql_real_escape_string($_POST[$var]);
}
?>

```

## 2. Обработка форм

### Программа перевода значений между шкалами Фаренгейта и Цельсия

```

<?php // convert.php
$f = $c = "";

if (isset($_POST['f'])) $f = sanitizeStnng($_POST['f']);
if (isset($_POST['c'])) $c = sanitizeString($_POST['c']);

if ($f != "")
{
  $c = intval((5 / 9) * ($f - 32));
  $out = "$f °f равно $c °c";
}
elseif($c != "")
{
  $f = intval((9 / 5) * $c + 32);
  $out = "$c °c равно $f °f";
}
else $out = "";

echo <<< _END

```



```

<html><head><title>программа перевода температуры</title >
</head><body><pre>
Введите температуру по Фаренгейту или по Цельсию и щелкните на кнопке
Перевести

<b>$out</b>
<form method="post" action="convert.php">
По Фаренгейту <input type="text" name="f" size="7" />
  По Цельсию <input type="text" name="c" size="7" />
                <input type="submit" value="Перевести" />
</form></pre></body></html >
_END;

```

```

function sanitizeString($var)
{
    $var = stripslashes($var);
    $var = htmlentities($var);
    $var = strip_tags($var);
    return $var;
}
?>

```

### 3. Работа с шаблонами Smarty

**Программа *sqltest.php*, переделанная для использования Smarty и сохраненная в файле *smartytest.php***

```

<?php // smartytest.php
$path = $_SERVER['DOCUMENT_ROOT'];
require "$path/Smarty/Smarty.class.php";

$smarty = new Smarty();
$smarty->template_dir = "$path/temp/smarty/templates";
$smarty->compile_dir = "$path/temp/smarty/tempiates_c";
$smarty->cache_dir = "$path/temp/smarty/cache";
$smarty->config_dir = "$path/temp/smarty/configs";

require_once("$path/temp/login.php");
$db_server = mysql_connect($db_hostname, $db_username, $db_password);

if (!$db_server) die("Невозможно подключиться к MySQL: " , mysql_error());

mysql_select_db($db_database)
or die("Невозможно выбрать базу данных: " . mysql_error());

```

```

if (isset($_POST['author']) &&
    isset($_POST['title']) &&
    isset($_POST['category']) &&
    isset($_POST['year']) &&
    isset($_POST['isbn']))
{
    $author = get_post('author');
    $title = get_post('title');
    $category = get_post('category');
    $year = get_post('year');
    $isbn = get_post('isbn');

    if (isset($_POST['delete']) && $isbn != "")
    {
        $query = "DELETE FROM classics WHERE isbn='$isbn'";

        if (!mysql_query($query))
        {
            echo "Сбой при удалении данных: $query<br>" .
                mysql_error() . "<p>";
        }
    }
    else
    {
        $query = "INSERT INTO classics VALUES" .
            "('$author', '$title', '$category', '$year', '$isbn')";

        if (!mysql_query($query))
        {
            echo "Сбой при вставке данных: $query<br>" .
                mysql_error() . "<p>";
        }
    }
}

$query = "SELECT * FROM classics";

$result = mysql_query($query);

if (!$result) die ("Сбой при доступе к базе данных: " . mysql_error());
$rows = mysql_num_rows($result);

for ($j = 0 ; $j < $rows ; ++$j)

```

```

{
    $results[] = mysql_fetch_array($result);
}

mysql_close($db_server);

$smarty->assign('results'. Sresults);
$smarty->display("$path/temp/smartytest.tpl");

function get_post($var)
{
return mysql_real_escape_string($_POST[$var]);
}
?>

```

#### 4. Использование YUI для Ajax

##### **calendar.html - календарь, обеспечивающий возможность множественного выбора**

```

<html ><head><title>YUI-Календарь</title>
<link rel="stylesheet"
    type="text/css" href="fonts-min.css" />
<link rel="stylesheet"
    type="text/css" href="assets/skins/sam/calendar.css" />
<script src="yahoo-dom-event.js"></script>
<script src="calendar-min.js"></script>
</head><body class="yui-skin-sam">
<div id="callContainer"></div>

<scri pt>
YAHOO.namespace("example.calendar")
YAHOO.example.calendar.init = function() {
    YAHOO.example.calendar.call =
        new YAHOO.widget.Calendar("call", "callContainer",
            { MULTI_SELECT: true } )
    YAHOO.example.calenda r.call.render()
}
YAHOO.util.Event.onDOMReady(YAHOO.example.calendar.init)
</script>
</body></html>

```

Остается только поместить следующий DIV-контейнер в то место на странице, где должен появиться календарь:

```
<div id="callContainer"></div>
```

Можно также непосредственно сослаться на разные файлы на сервере *Yahoo!*, изменив три строки со ссылками на библиотеки так, чтобы они выглядели следующим образом:

```
<link rel="stylesheet" type="text/css"
href="http://yui.yahooapis.com/2.7.0/build/calendar/assets/skins/sam/calendar.ess" />
<script src="http://yui.yahooapis.com/2.7.0/build/yahoo-dom-event/yahoo-dom-
event.js">
</script>
<script src="http://yui.yahooapis.com/2.7.0/build/calendar/calendar-min.js">
</script>
```

Литература:

Основная – 1; 3.

Дополнительная – 1; 2; 3;.

Интернет-ресурс:

1. Учебный курс для студентов направления подготовки «Бизнес-информатика» РФЭИ - <http://it.rfei.ru/~1r>.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ОК-8; ОК-12; ПК-3; ПК-4; ПК-8; ПК-11.

Образовательные результаты: З-1; З-2; З-3; У-1; В-1.

Форма контроля: выполнение практических заданий.

## Научно-исследовательская работа студентов

### **Темы: «Динамическое содержание веб-страниц»**

**Цель:** формирование у студентов научного мышления, способности верно производить постановку проблемы исследования, самостоятельно осуществлять поиск информации, анализ проблем и формирование адекватных выводов, исходя из масштабов ее деятельности, целей и задач.

#### **Тематика докладов:**

1. Этапы разработки веб-сайта.
2. Структура веб-сайта.
3. Принципы создания пользовательского интерфейса и поддержка структуры сайта.
4. Основные принципы дизайна при разработке веб-сайта.
5. История HTML.
6. Спецификации HTML.
7. Семантические требования HTML.
8. Составные элементы HTML-документа. Атрибуты. Структура HTML-документа.
9. Разметка текста в HTML.
10. Списки и изображения в HTML.
11. Ссылки, таблицы, изображения в HTML.
12. Создание пользовательских форм в веб-документе.

#### **Литература:**

Основная – 1; 2; 4.

Дополнительная – 1; 2.

Интернет-ресурсы: 45 - 64.

Образовательные технологии, методы и формы обучения: дистанционные образовательные технологии, объяснительно-иллюстративного обучения с элементами проблемного изложения; развивающего обучения, проблемная лекция, практическое занятие.

Формируемые компетенции: ОК-1; ОК-4; ПК-3; ПК-4.

Образовательные результаты: З-1; З-2; В-1.

Форма контроля: групповая дискуссия, опрос.

Форма контроля: подготовка доклада.

### **Вопросы к зачету**

1. Динамическое содержимое веб-страницы
  2. Преимущества использования PHP, MySQL и JavaScript
  3. Основной синтаксис, операторы, команды PHP
  4. Выражения и управление процессом выполнения программы в PHP
  5. Функции и объекты PHP
  6. Массивы в PHP
  7. Основные характеристики MySQL
  8. Доступ к MySQL с использованием PHP
  9. Обработка форм на веб-сайтах
  10. Работа с шаблонами системы Smarty
  11. Cookie, сессии и аутентификация
  12. JavaScript и текст HTML
  13. Выражения и управление процессом выполнения сценариев в JavaScript
  14. Функции, объекты и массивы JavaScript
  15. Проверка данных и обработка ошибок в JavaScript и PHP
  16. Использование технологии Ajax
  17. Использование YUI для Ajax
- Объединение технологий для проектирование конкретных задач.

### **6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)**

См. Приложение №2 к рабочей программе.

### **7. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля).**

#### **Основная литература**

1. Е. С. Бенкен. PHP, MySQL, XML. Программирование для Интернета. - СПб.: БХВ-Петербург, 2011. - 304 с.
2. Люк Веллинг, Лора Томсон. Разработка веб-приложений с помощью PHP и MySQL. - М.: Вильямс, 2012. - 848 с.
3. Робин Никсон. Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript. - М.: Питер, 2012. - 496 с.
4. Мэтт Зандстра. PHP. Объекты, шаблоны и методики программирования. - М.: Вильямс, 2011. - 560 с.

#### **Дополнительная литература**

1. Чак Муссиано и Билл Кеннеди. HTML и XHTML. Подробное руководство. - М.: Символ-Плюс, 2011. - 752 с.
2. Линн Бейли, Майкл Моррисон. Изучаем PHP и MySQL. М.: ЭКСМО, 2010. - 800 с.

**8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), необходимых для освоения дисциплины (модуля)**

1. Электронная библиотека Регионального финансово-экономического института – <http://students.rfei.ru/a/students/library.jsp>
2. Учебный курс «Разработка веб-приложений» для студентов направления подготовки «Бизнес-информатика» <http://it.rfei.ru/~1r>
3. Википедия [Электронный ресурс]: [свобод. Интернет-энцикл.] – Электрон. дан. и прогр. – Режим доступа: <http://ru.wikipedia.org>, свободный. – Русскояз. часть междунар. проекта «Википедия».

***Веб-сайты, относящиеся к PHP***

4. <http://codewalkers.com>.
5. <http://developer.yahoo.com/php/>.
6. <http://forums.devshed.com>.
7. <http://free-php.net>.
8. <http://hotscripts.com/category/php/>.
9. <http://htmlgoodies.com/beyond/php/>.
10. <http://php.net>.
11. <http://php.resourceindex.com>.
12. <http://php-editors.com>.
13. <http://phpbuilder.com>.
14. <http://phpfreaks.com>.
15. <http://phpunit.de>.
16. <http://w3schools.com/php/>.
17. <http://zend.com>.

***Веб сайты, относящиеся к MySQL***

18. <http://code.google.com/edu/tools/01/mysql.html>.
19. <http://launchpad.net/mysql/>.
20. <http://mysql.com>.
21. <http://php.net/mysql>. О <http://planetmysql.org>.
22. <http://sun.com/software/products/mysql/>.
23. <http://sun.com/systems/solutions/mysql/resources.jsp>.
24. [http://w3schools.com/PHP/php\\_mysqlIntro.asp](http://w3schools.com/PHP/php_mysqlIntro.asp).

***Веб-сайты, относящиеся к JavaScript***

25. <http://developer.mozilla.org/en/JavaScript>.
26. <http://dynamicdrive.com>.
27. <http://javascript.about.com>.
28. <http://javascript.intemet.com>.
29. <http://javascript.com>.
30. <http://javascriptkit.com>.
31. <http://w3schools.com/JS/>.
32. <http://www.webreference.com/js/>.

***Веб-сайты, относящиеся к Ajax***

- 33. <http://ajax.asp.net>.
- 34. <http://ajaxian.com>.
- 35. <http://ajaxmatters.com>.
- 36. <http://developer.mozilla.org/en/AJAX>.
- 37. <http://dojotoolkit.org>.
- 38. <http://jquery.com>.
- 39. <http://mochikit.com>.
- 40. <http://mootools.net>.
- 41. <http://openjs.com>.
- 42. <http://prototypejs.org>.
- 43. <http://sourceforge.net/projects/clean-ajax>.
- 44. <http://w3schools.com/Ajax/>.

***Веб-сайты с разнообразными ресурсами***

- 45. <http://apachefriends.org>.
- 46. <http://easyphp.org>.
- 47. <http://eclipse.org>.
- 48. <http://editra.org>.
- 49. <http://fireftp.mozdev.org>.
- 50. <http://sourceforge.net/projects/glossword/>.
- 51. <http://mamp.info/en/>.
- 52. <http://pear.php.net>.
- 53. <http://programmingforums.org>.
- 54. <http://putty.org>.
- 55. <http://smarty.net>.
- 56. <http://wampserver.com/en/>.

***Веб-сайты с ресурсами издательства O'Reilly***

- 57. <http://onlamp.com>.
- 58. <http://onlamp.com/php/>.
- 59. <http://onlamp.com/onlamp/general/mysql.csp>.
- 60. <http://oreilly.com/ajax/>.
- 61. <http://oreilly.com/javascript/>.
- 62. <http://oreilly.com/mysql/>.
- 63. <http://oreillynet.com/javascript/>.

**9. Методические указания для обучающихся по освоению дисциплины (модулю)**

Методические указания по изучению дисциплины представляют собой комплекс рекомендаций и объяснений, позволяющих бакалавру оптимальным образом организовать процесс изучения данной дисциплины. Известно, что в структуре учебного плана значительное время отводится на самостоятельное изучение дисциплины. В рабочих программах дисциплин размещается примерное распределение часов аудиторной и внеаудиторной нагрузки по различным темам данной дисциплины.



Для успешного освоения дисциплины бакалавр должен:

1. Прослушать курс лекций по дисциплине.
2. Выполнить все задания, рассматриваемые на практических занятиях, включая решение задач.
3. Выполнить все домашние задания, получаемые от преподавателя.
4. Решить все примерные практические задания, рассчитанные на подготовку к промежуточной аттестации.

При подготовке к промежуточной аттестации особое внимание следует обратить на следующие моменты:

1. Выучить определения всех основных понятий.
2. Повторить все задания, рассматриваемые в течение семестра.
3. Проверить свои знания с помощью тестовых заданий.

#### **Рекомендации по работе на лекционном занятии**

На лекциях преподаватель излагает и разъясняет основные, наиболее сложные понятия темы, а также связанные с ней теоретические и практические проблемы, дает рекомендации на семинарское занятие и указания на самостоятельную работу. В ходе лекции бакалавр должен внимательно слушать и конспектировать лекционный материал.

#### **Рекомендации для самостоятельной работы**

Самостоятельная работа бакалавров – планируемая учебная, научно-исследовательская работа, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Цель самостоятельной работы бакалавра – научиться осмысленно и самостоятельно работать сначала с учебным материалом, затем с научной информацией, изучить основы самоорганизации и самовоспитания с тем, чтобы в дальнейшем непрерывно повышать свою квалификацию.

Целью самостоятельной работы бакалавров по дисциплине является овладение фундаментальными знаниями, профессиональными умениями и навыками решения задач и теоретическим материалом по дисциплине. Самостоятельная работа способствует развитию самостоятельности, ответственности и организованности, творческого подхода к решению различных проблем.

В зависимости от конкретных видов самостоятельной работы, используемых в каждой конкретной рабочей программе, следует придерживаться следующих рекомендаций.

Одной из форм текущего контроля знаний студентов является контрольная работа. Контрольная работа подразумевает знакомство с основной и дополнительной литературой, включая справочные издания, зарубежные источники, конспект основных положений, требующихся для запоминания и являющихся основополагающими в этой теме.

Выполняя контрольную работу, необходимо внимательно ознакомиться с условиями заданий и написать развернутый и аргументированный ссылкой на

нормативные акты и литературу ответ. При написании контрольной работы необходимо проанализировать научную и учебную специальную литературу, действующие нормативно-правовые акты, публикации в периодической печати, судебную практику, статистические данные. В процессе выполнения работы необходимо подтверждать свои выводы цифровыми примерами, представленными в виде таблиц, диаграмм, графиков, а также примерами судебной практики. Как правило, контрольные работы проводятся на семинарском занятии.

Подготовка к написанию реферата предполагает поиск литературы и составление списка используемых источников, изложение мнения авторов и своего суждения по выбранному вопросу; формулирование основных аспектов проблемы.

Коллоквиум представляет собой одну из форм учебных занятий, ориентированную на определение качества работы с конспектом лекций, подготовки ответов к контрольным вопросам и др. Коллоквиумы, как правило, проводятся в форме мини-экзамена, имеющего целью уменьшить список тем, выносимых на основной экзамен, и оценить текущий уровень знаний бакалавров.

При подготовке к практикуму/лабораторной работе бакалаврам предлагается выполнить задания, подготовить проекты, составленные преподавателем по каждой учебной дисциплине.

Следует также учитывать краткие комментарии при написании курсовой работы, если она предусмотрена рабочей программой, и подготовке к итоговому контролю, проводимого в форме зачета и (или) экзамена. Так, написание курсовой работы базируется на изучении научной, учебной, нормативной и другой литературы. Включает отбор необходимого материала, формирование выводов и разработку конкретных рекомендаций по решению поставленных цели и задач, проведение практических исследований по данной теме. Все необходимые требования к оформлению находятся в методических указаниях по написанию курсовой работы.

### **Рекомендации по подготовке к практическому (семинарскому) занятию**

Семинарское занятие представляет собой такую форму обучения в учреждениях высшего образования, которая предоставляет студентам возможности для обсуждения теоретических знаний с целью определения их практического применения, в том числе средствами моделирования профессиональной деятельности. Семинарские занятия служат для закрепления изученного материала, развития умений и навыков подготовки докладов, сообщений, приобретения опыта устных публичных выступлений, ведения дискуссии, аргументации и защиты выдвигаемых положений, а также для контроля преподавателем степени подготовленности бакалавров по изучаемой дисциплине. При наличии практических заданий по изучаемой дисциплине бакалавр выполняет все упражнения и задачи, подготовленные преподавателем.

Целью практического занятия является более углубленное изучение отдельных тем дисциплины и применение полученных теоретических навыков на практике.

Семинарское занятие не сводится к закреплению или копированию знаний, полученных на лекции. Его задачи значительно шире, сложнее и интереснее. Семинарское занятие одновременно реализует учебное, коммуникативное и профессиональное предназначение. Подготовка к практическому (семинарскому) занятию начинается с тщательного ознакомления с условиями предстоящей работы, т. е. с обращения к планам семинарских занятий.

Подготовка к практическим занятиям должна носить систематический характер. Это позволит бакалавру в полном объеме выполнить все требования преподавателя.

Тщательная подготовка к семинарским занятиям, как и к лекциям, имеет определяющее значение: семинар пройдет так, как аудитория подготовилась к его проведению.

Самостоятельная работа – столп, на котором держится вся подготовка по изучаемому курсу. Готовясь к практическим занятиям, следует активно пользоваться справочной литературой: энциклопедиями, словарями, альбомами схем и др. Владение понятийным аппаратом изучаемого курса является необходимостью.

При подготовке к семинару бакалавры имеют возможность воспользоваться консультациями преподавателя. Кроме указанных тем бакалавры вправе, по согласованию с преподавателем, избирать и другие интересующие их темы.

Определившись с проблемой, привлекающей наибольшее внимание, следует обратиться к рекомендуемой литературе. Следует иметь в виду, что в семинаре участвует вся группа, а потому задание к практическому занятию следует распределить на весь коллектив. Задание должно быть охвачено полностью и рекомендованная литература должна быть освоена группой в полном объеме.

Для полноценной подготовки к практическому занятию чтения учебника крайне недостаточно – в учебных пособиях излагаются только принципиальные основы, в то время как в монографиях и статьях на ту или иную тему поднимаемый вопрос рассматривается с разных ракурсов или ракурса одного, но в любом случае достаточно подробно и глубоко. Тем не менее, для того, чтобы должным образом сориентироваться в сути задания, сначала следует ознакомиться с соответствующим текстом учебника – вне зависимости от того, предусмотрена ли лекция в дополнение к данному семинару или нет. Оценив задание, выбрав тот или иной сюжет, и подобрав соответствующую литературу, можно приступать собственно к подготовке к семинару. Для получения более глубоких знаний бакалаврам рекомендуется изучать дополнительную литературу. Следует активно пользоваться справочной литературой: энциклопедиями, словарями, альбомами схем и др. Владение понятийным

аппаратом изучаемого курса является необходимостью. В ходе работы студент должен применить приобретенные знания при обобщении теоретического и практического материала, продемонстрировать навыки грамотного изложения своих мыслей с использованием общеправовой и отраслевой терминологии.

Семинар (практическое занятие) предполагает свободный обмен мнениями по избранной тематике. Преподаватель формулирует цель занятия и характеризует его основную проблематику. Заслушиваются сообщения бакалавров. Обсуждение сообщения совмещается с рассмотрением намеченных вопросов. Кроме того заслушиваются сообщения, предполагающие анализ публикаций по отдельным вопросам семинара. Поощряется выдвижение и обсуждение альтернативных мнений. Преподаватель подводит итоги обсуждения и объявляет оценки выступавшим бакалаврами. В целях контроля подготовленности бакалавров и привития им навыков краткого письменного изложения своих мыслей преподаватель в ходе семинарских занятий может осуществлять текущий контроль знаний в виде тестовых заданий.

На семинаре идёт не проверка вашей подготовки к занятию (подготовка есть необходимое условие), но степень проникновения в суть материала, обсуждаемой проблемы. Поэтому беседа будет идти не по содержанию прочитанных работ; преподаватель будет ставить проблемные вопросы, не все из которых могут прямо относиться к обработанной вами литературе.

В ходе практических занятий бакалавры под руководством преподавателя могут рассмотреть различные методы решения задач по дисциплине. Продолжительность подготовки к практическому занятию должна составлять не менее того объема, что определено тематическим планированием в рабочей программе. Практические занятия по дисциплине могут проводиться в различных формах:

1) устные ответы на вопросы преподавателя по теме занятия; 2) письменные ответы на вопросы преподавателя; 3) групповое обсуждение той или иной проблемы под руководством и контролем преподавателя; 4) заслушивания и обсуждение контрольной работы; 5) решение задач.

При работе необходимо не только привлечь наиболее широкий круг литературы, но и суметь на ее основе разобраться в степени изученности темы. Стоит выявить дискуссионные вопросы, нерешенные проблемы, попытаться высказать свое отношение к ним, привести и аргументировать свою точку зрения или отметить, какой из имеющихся в литературе точек зрения по данной проблематике придерживается автор и почему.

### **Рекомендации по работе с литературой**

Изучение литературы очень трудоемкая и ответственная часть подготовки к семинарскому занятию, написанию эссе, реферата, доклада и т.п. Работа над литературой, статья ли это или монография, состоит из трёх этапов – чтения работы, её конспектирования, заключительного обобщения сути изучаемой работы.

Работа с литературой, как правило, сопровождается записями в следующих

формах:

- конспект – краткая схематическая запись основного содержания научной работы. Целью конспектирования является выявление логики, схемы доказательств, основных выводов произведения;
- план – краткая форма записи прочитанного, перечень вопросов, рассматриваемых в книге, статье, составление плана раскрывает логику произведения, способствует ориентации в его содержании;
- выписки – либо цитаты из произведения, либо дословное изложение мест из источника, способствуют более глубокому пониманию читаемого текста;
- тезисы – сжатое изложение основных мыслей и положений прочитанного материала;
- аннотация – очень краткое изложение содержания прочитанной работы, составляется после полного прочтения и осмысливания работы;
- резюме – краткая оценка прочитанного произведения, отражает наиболее общие выводы и положения работы, ее концептуальные итоги.

Прежде, чем браться за конспектирование, скажем, статьи, следует её хотя бы однажды прочитать, чтобы составить о ней предварительное мнение, постараться выделить основную мысль или несколько базовых точек, опираясь на которые можно будет в дальнейшем работать с текстом.

Конспектирование – дело очень тонкое и трудоёмкое, в общем виде может быть определено как фиксация основных положений и отличительных черт рассматриваемого труда вкупе с творческой переработкой идей, в нём содержащихся. Конспектирование – один из эффективных способов усвоения письменного текста. Хотя само конспектирование уже может рассматриваться как обобщение, тем не менее есть смысл выделить последнее особицей, поскольку в ходе заключительного обобщения идеи изучаемой работы окончательно утверждаются в сознании изучающего. Достоинством заключительного обобщения как самостоятельного этапа работы с текстом является то, что здесь читатель, будучи автором обобщений, отделяет себя от статьи, что является гарантией независимости читателя от текста.

Если программа занятия предусматривает работу с источником, то этой стороне подготовки к семинару следует уделить пристальное внимание. В сущности, разбор источника не отличается от работы с литературой – то же чтение, конспектирование, обобщение.

### **Рекомендации к написанию реферата**

Использование реферата в качестве промежуточного или итогового отчета студента о самостоятельном изучении какой-либо темы учебного курса предполагает, прежде всего, установление целей и задач данной работы, а также его функциональной нагрузки в процессе обучения.

*Реферат* – это композиционно-организованное, обобщенное изложение

содержания источника информации (в учебной ситуации – статей, монографий, материалов конференции, официальных документов и др., но не учебника по данной дисциплине). Тема реферата может быть предложена преподавателем или выбрана студентом из рабочей программы соответствующей дисциплины.

Возможно, после консультации с преподавателем, обоснование и формулирование собственной темы.

**Тема реферата** должна отражать проблему, которая достаточно хорошо исследована в науке. Как правило, внутри такой проблемы выбирается для анализа какой-либо единичный аспект.

Тематика может носить различный характер:

- межпредметный,
- внутрипредметный,
- интегративный,
- быть в рамках программы дисциплины или расширять ее содержание (рассмотрение истории проблемы, новых теорий, новых аспектов проблемы).

**Целью реферата** является изложение какого-либо вопроса на основе обобщения, анализа и синтеза одного или нескольких первоисточников. Другими словами, реферат отвечает на вопрос «какая информация содержится в первоисточнике, что излагается в нем?».

Принимая во внимание, что реферат – одна из форм интерпретации исходного текста одного или нескольких первоисточников, следует сформулировать задачу, стоящую перед студентами: создать новый текст на основе имеющихся текстов, т.е. текст о тексте. Новизна в данном случае подразумевает собственную систематизацию материала при сопоставлении различных точек зрения авторов и изложении наиболее существенных положений и выводов реферируемых источников.

### **1. Требования к рефератам.**

Прежде всего, следует помнить, что реферат не должен отражать субъективных взглядов референта (студента) на излагаемый вопрос, а также давать оценку тексту.

Основными требованиями к реферату считаются:

1. информативность и полнота изложения основных идей первоисточника;
2. точность изложения взглядов автора – неискаженное фиксирование всех положений первичного текста,
3. объективность – реферат должен раскрывать концепции первоисточников с точки зрения их авторов;
4. изложение всего существенного – «чтобы уметь схватить новое и существенное в сочинениях» (М.В. Ломоносов);
5. изложение в логической последовательности в соответствии с обозначенной темой и составленным планом;

6. соблюдение единого стиля – использование литературного языка в его научно-стилевой разновидности;

7. корректность в характеристике авторского изложения материала.

## **2. Виды рефератов.**

По характеру воспроизведения информации различают рефераты репродуктивные и продуктивные.

Репродуктивные рефераты воспроизводят содержание первичного текста:

- реферат-конспект содержит в обобщенном виде фактографическую информацию, иллюстративный материал, сведения о методах исследования, о полученных результатах и возможностях их применения;
- реферат-резюме приводит только основные положения, тесно связанные с темой текста.

Продуктивные рефераты предполагают критическое или творческое осмысление литературы:

- реферат-обзор охватывает несколько первичных текстов, дает сопоставление разных точек зрения по конкретному вопросу;
- реферат-доклад дает анализ информации, приведенной в первоисточниках, и объективную оценку состояния проблемы.

По количеству реферируемых источников:

- монографические – один первоисточник;
- обзорные – несколько первичных текстов одной тематики.

По читательскому назначению:

- общие – характеристика содержания в целом; ориентация на широкую аудиторию;
- специализированные – ориентация на специалистов.

## **3. Этапы работы над рефератом.**

1. Выбор темы.
2. Изучение основных источников по теме.
3. Составление библиографии.
4. Конспектирование необходимого материала или составление тезисов.
5. Систематизация зафиксированной и отобранной информации.
6. Определение основных понятий темы и анализируемых проблем.
7. Разработка логики исследования проблемы, составление плана.
8. Реализация плана, написание реферата.
9. Самоанализ, предполагающий оценку новизны, степени раскрытия сущности проблемы, обоснованности выбора источников и оценку объема реферата.
10. Проверка оформления списка литературы.
11. Редакторская правка текста.
12. Оформление реферата и проверка текста с точки зрения грамотности и стилистики.

#### ***4. Структура реферата.***

В структуре реферата выделяются три основных компонента: библиографическое описание, собственно реферативный текст, справочный аппарат.

Библиографическое описание предполагает характеристику имеющихся на эту тему работ, теорий; историографию вопроса; выделение конкретного вопроса (предмета исследования); обоснование использования избранных первоисточников.

##### ***Собственно реферативный текст:***

Введение – обоснование актуальности темы, проблемы; предмет, цели и задачи реферируемой работы, предварительное формулирование выводов.

Основная часть – содержание, представляющее собой осмысление текста, аналитико-синтетическое преобразование информации, соответствующей теме реферата.

Основную часть рекомендуется разделить на два-три вопроса. В зависимости от сложности и многогранности темы, вопросы можно разделить на параграфы. Чрезмерное дробление вопросов или, наоборот, их отсутствие приводят к поверхностному изложению материала. Каждый вопрос должен заканчиваться промежуточным выводом и указывать на связь с последующим вопросом.

Заключение – обобщение выводов автора, область применения результатов работы.

##### ***Справочный аппарат:***

Список литературы – список использованных автором реферата работ (может состоят из одного и более изданий).

Приложения (необязательная часть) – таблицы, схемы, графики, фотографии и т.д.

##### ***Реферат как образец письменной научной речи***

###### **1. Качества научной речи.**

Функциональные стили различаются:

- характером передаваемой информации;
- сферой функционирования;
- адресатом;
- использованием языковых средств различных уровней.

Главной коммуникативной задачей реферата является выражение научных понятий и умозаключений.

Реферат должен быть написан научным стилем, что предполагает:

- передачу информации научного характера;
- функционирование в образовательной среде;
- в качестве адресата преподавателя, т.е. специалиста, или студентов,
- заинтересованных в получении данной информации;



- демонстрацию характерных языковых особенностей письменной разновидности научно-учебного подстиля литературного языка.

Научный стиль обладает рядом экстралингвистических характеристик, или качеств:

- точность – строгое соответствие слов обозначаемым предметам и явлениям действительности (знание предмета и умение выбрать необходимую лексику);
- понятность – доступность речи для тех, кому она адресована (правильное использование терминов, иностранных слов, профессионализмов);
- логичность, последовательность – четкое следование в изложении логике и порядку связей в действительности (первоисточнике);
- объективность – отсутствие субъективных суждений и оценок в изложении информации;
- абстрактность и обобщенность – отвлеченность от частных, несущественных признаков;
- преобладание рассуждения как типа речи над описанием и повествованием;
- графическая информация наличие схем, графиков, таблиц, формул и т.п.

## 2. Особенности письменной научной речи

Письменная речь, в отличие от устной, подразумевает:

- определенную степень подготовленности к работе;
- возможность исправления и доработки текста;
- наличие композиции строения, соотношения и взаимного расположения частей реферата;
- выдержанность стиля изложения; строгое следование лексическим и грамматическим нормам.

Доминирующим фактором организации языковых средств в научном стиле является их обобщенно-отвлеченный характер на лексическом и грамматическом уровнях языковой системы.

Лексический уровень предполагает:

- использование абстрактной лексики, преобладающей над конкретной: мышление, отражение, изменяемость, преобразование, демократизация и т.п.;
- отсутствие единичных понятий и конкретных образов, что подчеркивается употреблением слов обычно, постоянно, регулярно, систематически, каждый и т.п.;
- преобладание терминов различных отраслей науки: лексикология, коммуникация, эмпиризм, гносеология, адаптация и т.п.;

- использование слов общенаучного употребления: функция, качество, значение, элемент, процесс, анализ, доказательство и т.п.;
- употребление многозначных слов в одном (реже двух) значениях: предполагать (считать, допускать); окончание (завершение), рассмотреть (разобрать, обдумать, обсудить) и т.п.;
- наличие специфических фразеологизмов: рациональное зерно, демографический взрыв, магнитная буря и т.п.;
- клиширование: представляет собой..., включает в себя..., относится к..., заключается в... и т.п.;
- преобладание отвлеченных существительных над однокоренными глаголами: взаимодействие, зависимость, классификация, систематизация и т.п.

#### Грамматический уровень:

- использование аналитической степени сравнения: более сложный, наиболее простой, менее известный и т.п. в отличие от эмоционально окрашенных: наиважнейший, сложнейший, ближайший и т.п.;
- преимущественное употребление глаголов 3 лица ед. и мн.ч. настоящего времени (реже 1 лица будущего времени сравним, рассмотрим): исследуются, просматривается, подразумевается, доказывает и т.п.;
- активность союзов, предлогов, предложных сочетаний: в связи..., в соответствии..., в качестве..., в отношении..., сравнительно с ... и т.п.;
- преобладание пассивных (страдательных) конструкций: рассмотрены вопросы,
- описаны явления, сделаны выводы, отражены проблемы и т.п.;
- выражение четкой связи между частями сложного предложения: следует сказать, что...; наблюдения показывают, что..., необходимо подчеркнуть, что... и т.п.;
- усиленная связующая функция наречий и наречных выражений: поэтому, итак, таким образом, наконец... и т.п.;
- осложнение предложений обособленными конструкциями: «Стремлением к смысловой точности и информативности обусловлено употребление в научной речи конструкций с несколькими вставками и пояснениями, уточняющими содержание высказывания, ограничивающими его объем, указывающими источник информации и т.д.».

Обобщая отличительные языковые особенности письменного научного стиля, можно сказать, что он характеризуется:

- употреблением книжной, нейтральной и терминологической лексики;

- преобладанием абстрактной лексики над конкретной;
- увеличением доли интернационализмов в терминологии;
- относительной однородностью, замкнутостью лексического состава;
- неупотребительностью разговорных и просторечных слов; слов с эмоционально-экспрессивной и оценочной окраской;
- наличием синтаксических конструкций, подчеркивающих логическую связь и последовательность мыслей.

***Оформление реферата. Критерии оценки.***

Правила оформления реферата регламентированы. Объем – не более 10-15 стр. машинописного текста, напечатанного в формате Word 7,0, 8,0; размер шрифта – 14; интервал – 1,5, формат бумаги А 4, сноски постраничные, сплошные; поле (верхнее, нижнее, левое, правое) 2 мм; выравнивание – по ширине; ориентация книжная; шрифт Times New Roman Cyr.

Работа должна иметь поля; каждый раздел оформляется с новой страницы.

Титульный лист оформляется в соответствии с установленной формой.

На первой странице печатается план реферата, включающий в себя библиографическое описание; введение, разделы и параграфы основной части, раскрывающие суть работы, заключение; список литературы; приложения.

В конце реферата представляется список использованной литературы с точным указанием авторов, названия, места и года ее издания.

Критерии оценки реферата.

1. Степень раскрытия темы предполагает:

- соответствие плана теме реферата;
- соответствие содержания теме и плану реферата;
- полноту и глубину раскрытия основных понятий;
- обоснованность способов и методов работы с материалом;
- умение работать с литературой, систематизировать и структурировать материал;
- умение обобщать, делать выводы, сопоставлять различные точки зрения по рассматриваемому вопросу.

2. Обоснованность выбора источников оценивается:

- полнотой использования работ по проблеме;
- привлечением наиболее известных и новейших работ по проблеме (журнальные публикации, материалы сборников научных трудов и т.д.).

3. Соблюдение требований к оформлению определяется:

- правильным оформлением ссылок на используемую литературу;
- оценкой грамотности и культуры изложения;
- владением терминологией и понятийным аппаратом проблемы;
- соблюдением требований к объему реферата;

- культурой оформления.

### ***Защита реферата***

Рефераты обычно представляются на заключительном этапе изучения дисциплины как результат итоговой самостоятельной работы студента. Защита реферата осуществляется или на аудиторных занятиях, предусмотренных учебным планом, или на зачете как один из вопросов билета (последнее определяется преподавателем).

Если реферат подразумевает публичную защиту, то выступающему следует заранее подготовиться к реферативному сообщению, а преподавателю и возможным оппонентам – ознакомиться с работой.

Реферативное сообщение отличается от самого реферата прежде всего объемом и стилем изложения, т.к. учитываются особенности устной научной речи и публичного выступления в целом. В реферативном сообщении содержание реферата представляется подробно (или кратко) и, как правило, вне оценки, т.е. изложение приобретает обзорный характер и решает коммуникативную задачу (передать в устной форме информацию, которая должна быть воспринята слушателями). Учитывая публичный характер высказываний, выступающий должен:

- составить план и тезисы выступления;
- кратко представить проблематику, цель, структуру и т.п.;
- обеспечить порционную подачу материала не в соответствии с частями, разделами и параграфами, а сегментировать в зависимости от новизны информации;
- соблюдать четкость и точность выражений, их произнесение; обращать внимание на интонацию, темп, громкость и т.п. особенности публичного выступления;
- демонстрировать подготовленный характер высказываний, допуская, как в любой другой устной речи, словесную импровизацию.

### **Рекомендации по написанию эссе**

*Эссе* – средство, позволяющее оценить умение обучающегося письменно излагать суть поставленной проблемы, самостоятельно проводить анализ этой проблемы с использованием концепций и аналитического инструментария соответствующей дисциплины, делать выводы, обобщающие авторскую позицию по поставленной проблеме.

*Цель* эссе состоит в развитии таких навыков, как самостоятельное творческое мышление и письменное изложение собственных мыслей.

*Структура* эссе определяется предъявляемыми требованиями:

- мысли автора по проблеме излагаются в форме кратких тезисов.
- мысль должна быть подкреплена доказательствами – поэтому за тезисом следуют аргументы.

Аргументы – это факты, явления общественной жизни, события,

жизненные ситуации и жизненный опыт, научные доказательства, ссылки на мнение ученых и др.

Эссе обычно имеет кольцевую структуру (количество тезисов и аргументов зависит от темы, избранного плана, логики развития мысли):

- вступление
- тезис, аргументы
- тезис, аргументы
- тезис, аргументы
- заключение.

При написании эссе надо учитывать следующее:

Вступление и заключение должны фокусировать внимание на проблеме (во вступлении она ставится, в заключении – резюмируется мнение автора).

Необходимо выделение абзацев, красных строк, установление логической связи абзацев: так достигается целостность работы.

**Стиль изложения:** эмоциональность, экспрессивность, художественность.

**Правила написания эссе:**

- из формальных правил можно назвать только одно – наличие заголовка;
- внутренняя структура может быть произвольной. Поскольку это малая форма письменной работы, то не требуется обязательное повторение выводов в конце, они могут быть включены в основной текст или в заголовок;
- аргументация может предшествовать формулировке проблемы.

Формулировка проблемы может совпадать с окончательным выводом.

В качестве примера можете познакомиться с широко известными эссе И.А. Бунина («Недостатки современной поэзии»), Д.С. Мережковского («О причинах упадка и новых течениях современной русской литературы»), К.Д. Бальмонта («Элементарные слова о символической поэзии»), В.Я. Брюсова («Ключи тайн»), Вяч. Иванова («Символизм как миропонимание»), А.А. Блока («О лирике»).

#### **Учебно-методические указания к выполнению тестовых заданий**

Тестовый контроль отличается от других методов контроля (устные и письменные экзамены, зачеты, контрольные работы и т.п.) тем, что он представляет собой специально подготовленный контрольный набор заданий, позволяющий надежно и адекватно количественно оценить знания обучающихся посредством статистических методов.

Все вышеуказанные преимущества тестового контроля могут быть достигнуты лишь при использовании теории педагогических тестов, которая сложилась на стыке педагогики, психологии и математической статистики. Основными достоинствами применения тестового контроля являются:

- объективность результатов проверки, так как наличие заранее определенного эталона ответа (ответов) каждый раз приводит к одному и тому же результату;
- повышение эффективности контролирующей деятельности со стороны преподавателя за счет увеличения её частоты и регулярности;
- возможность автоматизации проверки знаний учащихся, в том числе с использованием компьютеров;
- возможность использования в системах дистанционного образования.

**Тест** – инструмент, состоящий из системы тестовых заданий с описанными системами обработки и оценки результата, стандартной процедуры проведения и процедуры для измерения качеств и свойств личности, изменение которых возможно в процессе систематического обучения.

Преимущество тестового контроля состоит в том, что он является научно обоснованным методом эмпирического исследования и в определенной сфере позволяет преодолеть умозрительные оценки знаний студентов. Следует отметить, что задания, используемые многими преподавателями и называемые ими тестовыми, на самом деле таковыми вовсе не являются. В отличие от обычных задач тестовые задания имеют четкий однозначный ответ и оцениваются стандартно на основе ценника. В самом простом случае оценка студента есть сумма баллов за правильно выполненные задания. Тестовые задания должны быть краткими, ясными и корректными, не допускающими двусмысленности. Сам же тест представляет собой систему заданий возрастающей трудности. Тестовый контроль может применяться как средство текущего, тематического и рубежного контроля, а в некоторых случаях и итогового.

Текущее тестирование осуществляется после изучения отдельной темы или группы тем. Текущее тестирование, прежде всего, является одним из элементов самоконтроля и закрепления слушателем пройденного учебного материала.

#### ***Виды тестовых заданий***

Тестовое задание (ТЗ) может быть представлено в одной из следующих стандартизированных форм:

- закрытое ТЗ, предполагающее выбор ответов (испытуемый выбирает правильный ответ (ответы) из числа готовых, предлагаемых в задании теста);
- открытое ТЗ (испытуемый сам формулирует краткий или развернутый ответ);
- ТЗ на установление правильной последовательности;
- ТЗ на установление соответствия между элементами двух множеств.

#### ***Закрытое тестовое задание***

Закрытое ТЗ состоит из неполного тестового утверждения с одним

ключевым элементом и множеством допустимых вариантов ответов, один или несколько из которых являются правильными. Тестируемый студент определяет правильные ответы из данного множества. Рекомендуется пять или шесть вариантов ответов, из которых два или три являются правильными.

Открытое тестовое задание

Открытое ТЗ имеет вид неполного утверждения, в котором отсутствует один или несколько ключевых элементов и требует самостоятельной формулировки ответа тестируемого. В качестве отсутствующих ключевых элементов могут быть: число, буква, слово или словосочетание. При формулировке задания на месте ключевого элемента необходимо поставить прочерк или многоточие.

Тестовое задание на установление правильной последовательности

ТЗ на установление правильной последовательности состоит из однородных элементов некоторой группы и четкой формулировки критерия упорядочения этих элементов.

Тестовое задание на установление соответствия

ТЗ на установление соответствия состоит из двух групп элементов и четкой формулировки критерия выбора соответствия между ними. Внутри каждой группы элементы должны быть однородными. Количество элементов во второй группе должно превышать количество элементов первой группы, но не более чем в 2 раза. Максимально допустимое количество элементов во второй группе не должно превышать 10. Количество же элементов в первой группе должно быть не менее двух.

**Требования к тестовым заданиям**

Для обеспечения адекватности оценки знаний тесты должны обладать следующими свойствами:

- тест должен быть **репрезентативным** с точки зрения изучаемого материала (ответы на вопросы, поставленные в тесте, не должны выходить за пределы данной учебной дисциплины);
- тест должен быть **уместным**: формулировка и состав вопросов должны соответствовать основной цели дисциплины (при тестировании по определенной теме вопросы должны соответствовать одной из основных задач дисциплины, упомянутых в программе курса);
- тест должен быть **объективным**, что заключается в неизбежности выбора правильного варианта ответа различными экспертами, а не только преподавателем, оставившим тест;
- тест должен быть **специфичным**, т.е. в тесте не должно быть таких вопросов, на которые мог бы ответить человек, не знающий данной дисциплины, но обладающий достаточной эрудицией;
- тест должен быть **оперативным**, что предусматривает возможность быстрого ответа на отдельный вопрос, поэтому вопросы формулируются

коротко и просто и не должны включать редко используемые слова, конечно, если эти слова не являются понятиями, знание которых предусмотрено в учебной дисциплине.

*Перечисленные свойства тестовых заданий обеспечивают необходимый качественный уровень проведения итогового контроля, к которому предъявляются следующие требования.*

Процесс тестирования должен быть **валидным** (значимым), когда результаты подтверждают конкретные навыки и знания, которые экзамен подразумевает проверить.

Тестирование является **объективным**, если результаты не отражают мнения или снисходительность проверяющего.

Убедиться в **надежности** тестирования можно, если результаты повторно подтверждены последующими контрольными мероприятиями.

**Эффективность** тестирования определяется, если его выполнение и оценивание не занимает больше времени или денег, чем необходимо.

Тестирование можно считать **приемлемым**, если студенты и преподаватели воспринимают контрольное мероприятие адекватно его значимости.

Изучение динамики процесса проверки знаний с помощью тестов позволяет установить индивидуальное время тестирования для каждого конкретного набора тестовых заданий. Нередко время тестирования для различных дисциплин устанавливается одинаковым на основании некоторого стандарта, не принимая во внимание специфику конкретной дисциплины и ее раздела.

### **Указания по подготовке к зачету/экзамену**

Формой итогового контроля знаний и умений, полученных в процессе изучения дисциплины является зачет и экзамен.

Экзамен (зачет) дает возможность преподавателю:

- выяснить уровень освоения студентами учебной программы дисциплины;
- оценить формирование у студентов определенных знаний и навыков их использования, необходимых и достаточных для будущей профессиональной деятельности;
- оценить умение студентов творчески мыслить и логически правильно излагать ответы на поставленные вопросы.

При подготовке к экзамену (зачету) необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и др. Сдача экзамена и (или) зачета предполагает полное понимание, запоминание и применение изученного материала на практике. Для успешной подготовки к промежуточной аттестации студентам необходимо вновь обратиться к пройденному материалу. Литература для подготовки к экзамену (зачету) рекомендуется преподавателем, либо указана в рабочей программе по дисциплине.

При подготовке к промежуточной аттестации в качестве ориентира



студент может использовать перечень контрольных вопросов для самопроверки. Подготовка ответов на эти вопросы позволит:

- выяснить уровень освоения студентами учебных программ;
- оценить формирование у студентов определенных знаний и навыков их использования, необходимых и достаточных для будущей профессиональной деятельности;
- оценить умение студентов творчески мыслить и логически правильно излагать ответы на поставленные вопросы.

Оценка знаний студентов должна опираться на строго объективные критерии, научно обоснованные педагогикой и обязательные для выполнения всех преподавателей.

Среди таких критериев важнейшими являются принципы подхода к оценке. В наиболее общем виде эти принципы можно представить следующим образом:

- глубокие знания и понимание существа вопроса, но не всех его деталей, а лишь основных;
- степень сознательного и творческого усвоения изучаемых наук как базы личных убеждений и полезных обществу действий;
- понимание сущности науки, места каждой темы в общем курсе и её связи с предыдущими и последующими темами;
- выделение коренных проблем науки и умение правильно использовать это знание в самостоятельной научной деятельности или практической работе по специальности.

Экзамен (зачет) может проводиться в устной, письменной форме и с применением тестов. Экзамен (зачет) проводится по вопросам, охватывающим весь пройденный материал. По окончании экзамена (зачета) преподаватель может задать студенту дополнительные и уточняющие вопросы.

Студентам необходимо тщательно готовиться к итоговому экзамену. Процесс подготовки к итоговому экзамену начинается, по существу, с самого первого этапа изучения предмета. Он включает в себя самостоятельную работу над рекомендованной литературой. Как правило, он начинается за полтора-два месяца до экзаменационной сессии. Изучив и законспектировав рекомендованные источники, выполнив предусмотренные учебным планом письменные работы и имея рецензии на них, студент начинает непосредственную подготовку к экзамену с тщательной отработки курса в соответствии с требованиями учебной программы и выполнения рекомендаций преподавателя, данных в рецензии. На этом этапе студент должен повторить изученное по учебникам и учебным пособиям, личным конспектам, записям лекций и другим материалам. При этом особое внимание должно быть обращено на тщательную отработку тех конкретных вопросов и тем учебной программы, которые слабо усвоены.

При повторении материала перед итоговым экзаменом необходима самопроверка или взаимная проверка знаний. В этом случае по каждой теме надо еще раз хорошо продумать материал, найти соответствующие статьи из нормативных актов, подобрать примеры. Вполне себя оправдывает групповая взаимная проверка. Для этого рекомендуется собираться по 3-4 человека и проводить разбор вопросов по курсу. Экзамен проводится по билетам. Если какой-либо из поставленных в билете вопросов студенту кажется неясным, он может обратиться к преподавателю за разъяснением. Пользоваться наглядными пособиями, словарями или справочниками можно только с разрешения преподавателя. При подготовке к ответу, а также при ответе не обязательно придерживаться той последовательности вопросов, которая дана в билетах. Записи ответов лучше делать в виде развернутого плана, их можно дополнить цифрами, примерами, фактами, а также сослаться на необходимые нормативные акты и другие источники.

Ответ должен быть построен в форме свободного рассказа. Важно не только верно изложить соответствующее положение, но и дать его глубокое теоретическое обоснование. При ответах надо избегать больших выступлений, отклонений от существа вопросов, но не следует вдаваться и в такую крайность, как погоня за краткостью. Такой ответ не раскроет содержания вопроса и не даст возможности преподавателю правильно судить о знаниях студента. После ответов на вопросы билета преподаватель может задать дополнительные вопросы, на которые студент обязан ответить.

Экзаменатор оценивает знания по четырехбалльной системе: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Все положительные оценки записываются в экзаменационную ведомость и зачетную книжку. Неудовлетворительные оценки проставляются в экзаменационную ведомость.

#### **10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем**

При осуществлении образовательного процесса используется ряд информационных технологий обеспечения дистанционного обучения, включающий, но не исчерпывающийся, технологиями онлайн и оффлайн распространения образовательной информации (почтовая рассылка печатных материалов и бланков тестирования или электронных версий образовательных материалов на физических носителях, либо интерактивный доступ к материалам через интернет, доступ к электронно-библиотечным системам института и сторонних поставщиков), технологиями взаимодействия студентов с преподавателем (видео-лекции и семинары, групповые и индивидуальные консультации через интернет, индивидуальные консультации по телефону), технологиями образовательного контроля (интерактивные онлайн тесты в интернет, оффлайн тесты с использованием персональных печатных бланков).

Для реализации указанных технологий используется набор программного обеспечения и информационных систем, включающий, но не ограничивающийся, следующим списком.

18. операционные системы Microsoft Windows (различных версий);
19. операционная система GNU/Linux;
20. свободный офисный пакет LibreOffice;
21. система управления процессом обучения «Lete e-Learning Suite» (собственная разработка);
22. система электронного обучения студентов направления подготовки «Бизнес-информатика» EduTerra.pro
23. система интерактивного онлайн тестирования (собственная разработка);
24. система телефонной поддержки и консультаций сотрудниками колл-центра «Центральная служба поддержки» (собственная разработка);
25. система онлайн видео конференций Adobe Connect;
26. электронно-библиотечная система «Айбукс»;
27. электронно-библиотечная система «Издательства «Лань»;
28. интернет-версия справочника «КонсультантПлюс»;
29. приложение для мобильных устройств «КонсультантПлюс: Студент»;
30. справочная правовая система «Гарант»;
31. иные ИСС.

#### **11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)**

15. Аудиторная база (лекционная аудитория, аудитория для проведения практических занятий, виртуальные классные комнаты на портале РФЭИ)
16. Организационно-технические средства и аудиовизуальный фондовый материал, мультимедийное оборудование.
17. Комплекты видеофильмов, аудиокниг, CD-дисков по проблемам дисциплины.
18. Интернет.

## ПРИЛОЖЕНИЯ

### Приложение 1

Таблица. Результаты обучения по дисциплине «Разработка web-приложений», соотнесенные с планируемыми результатами освоения образовательной программы

Название дисциплины	Планируемые результаты обучения		Компетенции								
	код	описание	ОК-9	ОК-11	ОК-12	ОК-13	ОК-16	ОК-17	ПК-7	ПК-16	ПК-18
Разработка WEB-приложений	3-1	знает основные принципы функционирования Internet	x	x	x	x	x	x	x	x	x
	3-2	разбирается в современных сервисах и службы глобальной сети Internet									
	3-3	знает основные технологические этапы проектирования сайтов									
	3-4	освоил базовые концепции и приемы Web-программирования									
	3-5	применяет методы создания Web-страниц и их оптимизации									
	3-6	владеет инструментальными средствами создания Web-сайтов									
	У-1	умеет находить необходимую информацию в глобальной сети Internet, подготавливать информацию для её дальнейшего размещения в Internet									
	У-2	программирует Web-сайт с использованием технологий PHP, JavaScript, MySQL, Ajax									
	У-3	проектирует и реализовывать структуру Web-сайта									
	В-1	владеет навыками использования изученных технологий (PHP, JavaScript, MySQL, Ajax) и возможного их объединения при работе над конкретным Web-приложением									

## Приложение 2

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине, входящей в состав рабочей программы дисциплины **Разработка web-приложений**

Направление подготовки **38.03.05 (080500) Бизнес-информатика**  
 Профиль **Информационный бизнес**  
 Квалификация (степень) **Бакалавр**  
 Утверждена **21 декабря 2015 г.**

### 1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Региональный финансово-экономический институт при формировании компетенций студентов направления подготовки 38.03.05 (080500) «Бизнес-информатика» выделяет три этапа формирования компетенции:

- **начальный.** На этом этапе формируются знаниевые и инструментальные основы компетенции, осваиваются основные категории, формируются базовые умения. В целом, знания и умения носят репродуктивный характер. Студент воспроизводит термины, факты, методы, понятия, принципы и правила. На этом этапе он решает задачи, преимущественно, по образцу. Если студент удовлетворительно отвечает этим требованиям, можно говорить об освоении им базового (начального) уровня компетенции;
- **основной** этап – знания, умения, навыки, обеспечивающие формирование компетенции, значительно возрастают, но ещё не достигают целевых (итоговых) значений. На этом этапе студент осваивает действия с предметными знаниями в конкретной дисциплине и, часто, в междисциплинарном характере действий. Способен самостоятельно решать учебные задачи, внося коррективы в алгоритм своих действий, осуществлять саморегуляцию в ходе работы, переносить знания и умения на новые, возникающие в ходе выполнения работ, условия. Успешное прохождение этого этапа позволяет достичь удовлетворительного уровня сформированности компетенции;
- **завершающий** этап – на этом этапе студент достигает итоговых (целевых) показателей по заявленной компетенции. Он осваивает весь необходимый объём знаний, овладевает всеми умениями и навыками в сфере заявленной компетенции. Он способен использовать эти знания, умения и навыки при решении реальных задач и в нестандартных учебных условиях.

Дисциплина имеет целью участие в формировании следующих компетенций (список в соответствии с РУП направления подготовки, составленным в соответствии с государственным стандартом на направление подготовки 080500, утверждён ПРИКАЗОМ от 14 января 2010 г. N 27 «ОБ УТВЕРЖДЕНИИ И ВВЕДЕНИИ В ДЕЙСТВИЕ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ 080500 БИЗНЕС-ИНФОРМАТИКА (КВАЛИФИКАЦИЯ (СТЕПЕНЬ) "БАКАЛАВР"»), зарегистрировано в Минюсте РФ 27 февраля 2010 г. N 16524):

1. ОК-12
2. ОК-13
3. ОК-17
4. ОК-7
5. ОК-16
6. ОК-18

Этапы формирования компетенций обычно распределены следующим образом:

1. **Начальный** – формируется в процессе изучения отдельных разделов дисциплины, а успешность его освоения определяется с помощью критериев оценивания компетенции, подробно описанной в разделе [2] этого документа.
2. **Основной** – формируется на этапе успешного завершения всех дисциплин, участвующих в процессе формирования компетенции.
3. **Завершающий** – достигается на основании комплексной междисциплинарной работы, в ходе итоговых практик, экзаменов, выполнении дипломной работы и подтверждении успешного овладения компетенцией.

Завершение дисциплины с точки зрения показателей раздела [2] означает успешное освоение как минимум начального уровня овладения компетенцией.

## **2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания**

Контроль достижения целевых критериев на этапе текущего формирования компетенции при изучении любых дисциплин направления подготовки осуществляется на основании следующих инструментов (средств оценивания):

1. индивидуальные задания расчётного типа;
2. индивидуальные задания графического типа;
3. индивидуальные задания вербального типа;
4. индивидуальные задания расчётно-графического типа;
5. индивидуальные темы рефератов по заданной теме;
6. индивидуальные темы эссе по заданной теме;

7. индивидуальные задания для выполнения контрольных работ;
8. тесты в ЭИОС по темам дисциплины:
  - а. базовый уровень
  - б. высокий уровень
  - с. повышенный уровень
9. задания для выполнения лабораторных работ;
10. вопросы для защиты лабораторных работ;
11. задания для подготовки и защиты докладов;
12. сценарии ролевых игр;
13. сценарии мастер-классов;
14. задания для выполнения курсовых работ (проектов);
15. задания для выполнения научно-исследовательских работ;
16. задания для прохождения практик;
17. вопросы к экзамену;
18. вопросы к государственному экзамену;
19. задания для выполнения выпускных квалификационных работ.

Основными типами промежуточного контроля являются тестирования вербального и невербального типов в ЭИОС РФЭИ.

Эти тесты различаются по характеру стимульного материала.

В вербальных типах заданий основным содержанием работы испытуемых являются операции с понятиями, мыслительные действия, осуществляемые в словесно-логической форме. Составляющие эти методики задания апеллируют к памяти, воображению, мышлению в их опосредованной языковой форме. Они очень чувствительны к различиям в языковой культуре, уровню образования, профессиональным особенностям. Вербальный тип заданий наиболее распространён в компетентностных тестах, тестах достижений, при оценке специальных способностей. Невербальные тесты — это такой тип методик, в которых тестовый материал представлен в наглядной форме (в виде картинок, чертежей, графических изображений и т. п.). От испытуемых требуется понимание вербальных инструкций, само же выполнение заданий опирается на перцептивные и моторные функции.

Невербальные тесты уменьшают влияние языковых различий на результат испытания. Они также облегчают процедуру тестирования испытуемых с нарушением речи, слуха или с умеренным уровнем подготовки. Невербальные тесты широко используются при оценке начального этапа формирования компетенции.

Программа изучения дисциплины составлена таким образом, что успешное её освоение возможно с различными результатами. Все задания разделены на обязательные и необязательные. Успешное выполнение всех обязательных заданий означает достижение удовлетворительного уровня по освоению дисциплины.

Количество обязательных заданий текущего контроля не менее 65% от общего

количества заданий. Все обязательные задания предполагают возможность повторного выполнения (как автоматически, так и в ряде случаев по согласованию/дополнительному разрешению). Успешное выполнение всех обязательных заданий гарантирует студенту оценку «удовлетворительно» в зачётной книжке, если изучение этой дисциплины предполагает выставление оценки.

Необязательный уровень включает задания высокой и повышенной (относительно высокой) сложности. Их успешное выполнение необязательно для студента, однако их выполнение непосредственно влияет на оценку по дисциплине, а также более глубокий уровень освоения предметной областью дисциплины. Успешное завершение всех заданий высокой сложности предполагает получение оценки «хорошо», а повышенной сложности «отлично» при оценивании результатов освоения дисциплины.

Текущий подход является формализованным для всех дисциплин направления подготовки «Бизнес-информатика» и **обязателен к применению в рамках текущей дисциплины.**

В связи с различиями в части применения дисциплины на разных формах обучения и конкретных профилях здесь приводятся полные сведения о способе формирования оценки.

1. Если по дисциплине в РУПе не предусмотрен промежуточный контроль (в РУПе по дисциплине указан только ОДИН итоговый экзамен)

**Накопленная оценка по дисциплине** рассчитывается с помощью взвешенной суммы оценок за отдельные формы текущего контроля знаний следующим образом:

$O_{\text{накопленная}} = n_1 \cdot O_{\text{текущий1}} + n_2 \cdot O_{\text{текущий2}} + n_3 \cdot O_{\text{текущий3}} + \dots + n_i \cdot O_{\text{текущийi}}$ , где

$O_{\text{текущий1}}$  – оценка за текущее компьютерное тестирование (базовый, минимальный уровень)

$O_{\text{текущий2}}$  – оценка за текущее компьютерное тестирование (высокий уровень освоения)

$O_{\text{текущий3}}$  – оценка за текущее компьютерное тестирование (повышенной сложности)

$O_{\text{текущий4}}$  – оценка за эссе

...

$O_{\text{текущийi}}$  – оценка за реферат, доклад и т.п.

$n_1, n_2, n_3, \dots, n_i$  - веса оценок за отдельные формы текущего контроля ( $O_{\text{текущий1}}, O_{\text{текущий2}}, O_{\text{текущий3}}, \dots, O_{\text{текущийi}}$ )

$n_1=0.6, n_2=0.2, n_3=0.1, n_4=0.1$

Сумма весов оценок за отдельные формы текущего контроля, которые учитываются в накопленной оценке, должна быть равна единице (нормализуются):

$$\sum n_i = 1$$

Способ округления накопленной оценки текущего контроля: **в пользу студента.**



**Результирующая оценка по дисциплине** (которая пойдёт в диплом и является критерием оц) рассчитывается следующим образом:

$$O_{\text{результ}} = k_1 \cdot O_{\text{накопл}} + k_2 \cdot O_{\text{экз}}, \text{ где}$$

$O_{\text{накопл}}$  – накопленная оценка по дисциплине

$O_{\text{экз}}$  – оценка за экзамен

$k_1$  – вес накопленной оценки по дисциплине

$k_2$  – вес экзаменационной оценки по дисциплине

Сумма весов ( $k_1 + k_2$ ) должна быть равна единице:  $\sum k_i = 1$ , при этом,  $0,2 \leq k_i \leq 0,8$ . Вес итоговой аттестации не может быть менее 20% от всей дисциплины.

Для текущей дисциплины  $k_1 = 0,8$

Способ округления экзаменационной и результирующей оценок: среднее арифметическое.

2. Если по дисциплине в РУПе предусмотрен промежуточный контроль (в РУПе по дисциплине указано БОЛЕЕ одного экзамена)

Итоговая накопленная оценка по дисциплине рассчитывается следующим образом:

$$O_{\text{накопленная Итоговая}} = (O_{\text{промежуточная 1}} + O_{\text{промежуточная 2}} + \dots + O_{\text{накопленная i}}) : \text{на число этапов,}$$

$O_{\text{промежуточная 1}}$  – промежуточная оценка 1 этапа/модуля

$$O_{\text{промежуточная 1}} = m_1 \cdot O_{\text{накопленная 1 этапа}} + m_2 \cdot O_{\text{промежуточный экзамен 1 этапа}}$$

Сумма весов ( $m_1 + m_2$ ) должна быть равна единице, при этом,  $0,2 \leq m_i \leq 0,8$

$O_{\text{промежуточная 2}}$  – промежуточная оценка 2 этапа/модуля

$$O_{\text{промежуточная 2}} = m_3 \cdot O_{\text{накопленная 2 этапа}} + m_4 \cdot O_{\text{промежуточный экзамен 2 этапа}}$$

Сумма весов ( $m_3 + m_4$ ) должна быть равна единице, при этом,  $0,2 \leq m_i \leq 0,8$

$O_{\text{накопленная 1 этапа}}$ ,  $O_{\text{накопленная 2 этапа}}$  рассчитываются по приведенной выше формуле расчета накопленной оценки (за каждый этап)

$O_{\text{накопленная i}}$  – накопленная оценка последнего этапа/модуля перед ИТОВЫМ экзаменом

$O_{\text{накопленная i}}$  рассчитывается по приведённой выше формуле расчёта накопленной оценки (для последнего этапа/модуля перед итоговым экзаменом)

**Результирующая оценка по дисциплине** (которая идёт в диплом и является одним из критериев оценивания достижения основного этапа освоения компетенции) рассчитывается следующим образом:

$$O_{\text{результ}} = k_1 \cdot O_{\text{накопленная Итоговая}} + k_2 \cdot O_{\text{Итоговый экз}}$$

$O_{\text{Итоговый экз}}$  – оценка за **ИТОВЫЙ** экзамен

Сумма весов ( $k_1 + k_2$ ) должна быть равна единице:  $\sum k_i = 1$ , при этом,  $0,2 \leq k_i \leq 0,8$

Способ округления накопленных, промежуточных, экзаменационных и результирующей оценок: **среднее арифметическое**

**3. Типовые контрольные задания и иные материалы, необходимые для оценки знаний, умений, навыков и опыта деятельности, характеризующие этапы формирования компетенций в процессе**

### **освоения образовательной деятельности.**

В соответствии с описанием показателей и критериев оценивания, подробно описанные в пункте 2 этого документа, здесь приводится неполный список **примеров** тестовых заданий.

См. приложение 3.1 «Типовые контрольные задания», являющееся частью рабочей программы дисциплины.

#### **4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и опыта деятельности, характеризующие этапы формирования компетенций**

Система текущего контроля успеваемости и промежуточной аттестации студентов предусматривает решение следующих задач:

- оценка качества освоения студентами основной профессиональной образовательной программы (оцениваются знания, умения и навыки);
- аттестация студентов на соответствие их персональных достижений поэтапным требованиям соответствующей основной профессиональной образовательной программы;
- поддержание постоянной обратной связи и принятие оптимальных решений в управлении качеством обучения студентов на уровне преподавателя, кафедры, факультета и института целиком.

Текущий контроль успеваемости и промежуточная аттестация является основным механизмом оценки качества подготовки студентов (согласно требованиям ФГОС) и формой контроля учебной работы студентов.

Оценка качества подготовки студентов осуществляется в двух основных направлениях: оценка уровня освоения дисциплины и оценка компетенций студентов. Предметом оценивания являются знания, умения, компетенции обучающихся.

Промежуточная аттестация студентов проводится по учебной дисциплине в сроки, предусмотренные учебными планами и годовыми календарными учебными графиками в порядке, утверждённом в вузе.

Каждая компетенция формируется на всех этапах обучения студента в процессе изучения ряда дисциплин, а после, использования междисциплинарных знания для выполнения дипломной работы и практик.

Знания, умения и навыки постепенно формируют целевую компетенцию. Поэтому существенно отличаются и методы контроля промежуточной и итоговой оценки достижения компетенций.

Промежуточные методы контроля включают в себя автоматические и неавтоматические методы контроля, такие как тестирование или аттестация/не аттестация по выполнению требуемых видов работ.

С целью определения уровня овладения компетенциями, в заданные логикой преподавания дисциплины сроки проводится текущий и промежуточный контроль знаний, умений и навыков каждого обучающегося. Все виды

текущего контроля осуществляются в соответствии с формой задания (см. п.2 «описание показателей и критериев оценивания...»).

Процедура оценивания компетенций обучающихся основана на следующих условиях:

1. Периодичность проведения оценки (минимум 1 раз на каждую рассматриваемую тему в дисциплине).
2. Многоступенчатость: оценка (как автоматически с помощью ЭИОС или преподавателем) и самооценка обучающегося, обсуждение результатов и комплекс мер по устранению недостатков.
3. Единство используемой технологии для всех обучающихся, выполнение условий сопоставимости результатов оценивания.

### **5. Показатели и критерии оценивания сформированности компетенций**

Основным критерием итоговой сформированности любой компетенции является успешное завершение обучения студентом, выполнение и защита дипломной работы и государственного экзамена, прохождение и защита практик.

Успешное завершение дисциплины означает достижение очередного шага в формировании компетенции. Критерием успешного завершения дисциплины является как минимум выполнение всех обязательных требований (заданий) из перечня в пункте 2 этого документа. Критерии успешного завершения каждого из заданий определяются в самих заданиях. Примеры заданий можно посмотреть в п.3 этого документа.

Шкалы оценивания предусматривают детальный ответ на вопрос об уровне освоения дисциплины и, посредством оценивания процедур знаний, умений и навыков, показателей оценивания сформированности компетенции.

# Введение в динамическое содержимое веб-страницы

## Проверка знаний: Введение

### Общая группа

Укажите последовательность, в которой выполняется ответ сервера на запрос пользователем некоторой страницы, адрес которой был введен в веб-браузере.

- Запрос на главную страницу
- Поиск IP-адреса, соответствующего указанному доменному имени
- Получение запроса веб-сервером
- Поиск и извлечение страницы веб-сервером
- Отправка страницы в адрес браузера
- Отображение веб-страницы
- Поступление запроса на веб-сервер

Выберите верное утверждение:

- код JavaScript интерпретируется локально на машине клиента
- Ни одно из перечисленных утверждений не является верным
- код JavaScript всегда выполняется на стороне веб-сервера
- Интерпретатор PHP выполняет код JavaScript

Какие компоненты необходимы для создания полностью динамических веб-сайтов?

- база данных
- веб-сервер
- интерпретатор
- структурированный язык запросов
- язык сценариев на стороне клиента
- язык сценариев на стороне сервера

И PHP, и JavaScript являются языками программирования, генерирующими динамическое содержимое веб-страниц. В чем состоят их основные различия?

- Средствами PHP невозможно провести быстрое и динамическое изменение веб-страницы, просматриваемой пользователем
- JavaScript является языком гипертекстовой разметки
- У языка PHP есть средства общения с базой данных, чего нет у JavaScript
- Сценарии PHP работают на сервере, а сценарии JavaScript – на машине клиента
- У языка JavaScript есть средства, позволяющие хранить и извлекать данные

# Установка сервера, предназначенного для разработки

## Проверка знаний: Установка сервера, предназначенного для разработки

### Общая группа

Для IP-адреса `127.0.0.1` и URL-адреса `http://localhost` верно следующее утверждение:

- Оба адреса — и `127.0.0.1`, и `http://localhost` — являются способами ссылки на локальный компьютер.
- Для вызова исходной страницы на локальном сервере в адресную строку браузера можно вводить только `http://localhost`
- По адресу `http://127.0.0.1` всегда будет тестовая html-страница

Выберите верные утверждения:

- FTP это протокол передачи файлов — File Transfer Protocol
- Программа FTP используется для передачи файлов между клиентом и сервером в обе стороны.
- По умолчанию MAMP-сервер использует порт 8080 для Apache
- При посещении сайтов фактически идет обращение к порту 3306 того сервера, на котором находится веб-сайт.

IDE - это:

- специализированный редактор программ
- интегрированная среда разработки
- приложение для проведения отладки и тестирования программ

# Введение в PHP

## Проверка знаний: Введение в PHP

### Общая группа

Что выведет на экран следующий код?

```
$a = 'p';  
$a++;  
echo $a;
```

```
++$a;  
echo $a;
```

```
$a+=1;  
echo $a;
```

- pr2
- qr1
- qr0
- pq1

В чем разница между командами `echo` и `print`?

- Команды `echo` и `print` идентичны
- `echo` является функцией PHP и не принимает аргументов
- `echo` является конструкцией, которая может принимать несколько аргументов
- `print` является функцией PHP и принимает один аргумент

У какого из операторов более высокий уровень приоритета `&&` и `and`?



- Операторы `&&` и `and` являются полностью взаимозаменяемыми.
- У оператора `&&` более высокий уровень приоритета, чем у оператора `and`.
- У оператора `and` более высокий уровень приоритета, чем у оператора `&&`.

Выберите верные утверждения:

- Для того чтобы перевести значение переменной из одного типа в другой, нужно сослаться на эту переменную, и PHP автоматически преобразует ее тип.
- Имена переменных не чувствительны к регистру букв
- Объявив переменную статической, ее можно сделать доступной для всех частей PHP-программы.
- Переменная `$current-user` является корректной
- `$This_Variable` и `$this_variable` являются совершенно разными переменными
- Все имена PHP-переменных, исключая имена констант, должны начинаться с символа `$`

Закончите верно следующее утверждение:

Использование выражения `++$j` приводит к увеличению значения `$j` на единицу до выполнения проверки или другой инструкции, а использование выражения `++$j` приводит к тому, что сначала выполняется инструкция, а затем значение переменной `$j` увеличивается на единицу.

Что является результатом объединения строки с числом?

- При объединении строки с числом получается еще одна строка.
- будет ошибка несоответствия типов данных.
- При объединении строки с числом получается число.

Какой оператор используется для конкатенации строк?

- :
- &
- +
- .

Какой результат выполнения следующего кода?

```
$i = 4;  
$j = 30;  
$k = 0;  
$k = $j++/$i++;  
echo $i . " " . $j . " " . $k . " " ;
```

- 4 31 7.5
- 5 31 7.5
- 5 31 6
- 5 31 6.2
- 5 31 7

# Выражения и управление процессом выполнения программы в PHP

## Проверка знаний: Выражения и управление процессом выполнения программ

### Общая группа

Какие основные значения представлены ключевыми словами `TRUE` и `FALSE` ?

- `FALSE` представляет значение `NULL`
- `FALSE` представляет значение `0`
- `TRUE` представляет значение `1`
- `FALSE` представляет значение `NULL`, `NULL` можно рассматривать как значение `0`

Что представляют собой самые простые формы выражений?

- Только числа и строки, и операции с ними
- Любая переменная
- Инструкция, результатом вычисления которой является `TRUE` или `FALSE`
- Это литералы, к которым относятся числа и строки, и переменные, которые просто вычисляются в самих себя

В чем разница между унарными, бинарными и трехкомпонентными операторами?

- В уровне приоритета операторов
- В количестве необходимых им операндов
- В направлении обработки выражений

Какие из перечисленных ключевых слов относятся к оператору ветвления?

- if
- then
- else
- continue
- elseif

Что выведет следующий код:

```
<?
$note = 3;
switch($note) {
default:
    echo "c";
case 1:
    echo "a";
    break;
case 2:
    echo "b";
}
?>
```

- c
- ca
- ab
- cab
- abc

Какой будет результат в переменной \$c после выполнения?

```
$a = 2;  
$b = 3;  
$c = $b++ - ++$a;
```

- 0
- 2
- 4
- 1

Что выведет следующий код:

```
<?  
$n = 5;  
echo $n++;  
>
```

- 6
- ЭТОТ КОД ВЫДАСТ ОШИБКУ
- 4
- 5

Что выведет данный код?

```
for ($i = 0; $i < 5; $i++) {  
    if ($i == 2) break;  
    print ++$i;  
}
```

- 13
- 1
- Сообщение об ошибке
- 0

# Функции и объекты PHP

## Проверка знаний: Функции и объекты PHP

### Общая группа

В какой строке кода содержится синтаксическая ошибка:

```
function func($a) {  
    if ($a > 0) return;  
    echo $a;  
}
```

- в первой: `func` это зарезервированное слово
- в третьей: этот код не достижим в случае если `$a` больше нуля
- в этом коде нет ошибок
- во второй: не указано значение, которое возвращается функцией

Дан следующий фрагмент кода:

```
<?php  
  
class foo  
{  
    public function printItem($string)  
    {  
        echo 'Foo: ' . $string . PHP_EOL;  
    }  
  
    public function printPHP()  
    {  
        echo 'PHP is great.' . PHP_EOL;  
    }  
}  
  
class bar extends foo  
{  
    public function printItem($string)
```

```
        {
            echo 'Bar: ' . $string . PHP_EOL;
        }
    }

    $foo = new foo();
    $bar = new bar();
    $foo->printItem('baz');
    $foo->printPHP();
    $bar->printItem('baz');
    $bar->printPHP();

    ?>
```

Что будет выведено в результате его выполнения?

1.

```
Foo: baz
PHP is great
Bar: baz
PHP is great
```

2.

```
Foo: Bar: baz
PHP is great
Bar: baz
PHP is great
```

3.

```
Foo: Bar: baz
PHP is great
Bar: baz
Bar: PHP is great
```

4.

```
Foo: baz
PHP is great
Bar: baz
Bar: PHP is great
```

- 1
- 3
- 2
- 4

Выберите верные утверждения относительно следующего кода:

```
<?php

class MyClass
{
    public $public = 'one_field';
    protected $protected = 'two_field';
    private $private = 'tree_field';

    function printField()
    {
        echo $this->public;
        echo $this->protected;
        echo $this->private;
    }
}

$obj = new MyClass();
echo $obj->public;      // (1)
echo $obj->protected;  // (2)
echo $obj->private;    // (3)
$obj->printField();    // (4)
```

- В строке (2) ошибка
- Будет выведено `one_field two_field tree_field` и предупреждение об ошибке
- В строке (3) ошибка
- Код выполнится без ошибок
- Код не выполнится, на экран будет выведено сообщение об ошибке
- Будет выведено `one_field` и ошибка доступа



Выберите верные утверждения:

- Изменение значения переменной, при обращении к ней по ссылке не приводит к изменению исходного значения переменной
- На одно и то же значение переменной может ссылаться сразу несколько имен
- На одно и то же значение переменной может быть только 1 указатель
- При обращении к переменной по имени, значение переменной копируется
- Изменение значения переменной, при обращении к ней по ссылке приводит к изменению исходного значения

Выберите верные утверждения относительно следующего фрагмента кода:

```
<?php
class MyClass
{
    protected function myFunc() {
        echo "MyClass::myFunc()\n";
    }
}

class OtherClass extends MyClass
{
    public function myFunc()
    {
        parent::myFunc();
        echo "OtherClass::myFunc()\n";
    }
}

$class = new OtherClass();
$class->myFunc();
?>
```

- Будет осуществлен автоматический вызов методов, принадлежащих классу-родителю, их не следует определять перегружаемом в дочернем классе
- Будет выведено: `MyClass::myFunc()` `OtherClass::myFunc()`
- Будет выведено: `OtherClass::myFunc()`
- Вызов методов, принадлежащих классу-родителю осуществляется в перегружаемом в дочернем классе

Какой результат выполнения данного кода?

```
<?php
function byRef(&$apples)
{
    $apples++;
}
$oranges = 5;
$apples = 5;
byRef($oranges);
echo "У меня есть $apples яблок и $oranges апельсинов";
?>
```

- У меня есть 5 яблок и 5 апельсинов
- У меня есть 6 яблок и 5 апельсинов
- У меня есть 5 яблок и 6 апельсинов
- У меня есть 6 яблок и 6 апельсинов

Что будет на экране в результате выполнения данного кода?

```
class A {
    private $var1;
    public function __construct($var1) {
        $this->var1=$var1;
    }
    public function method1() {
        $a = new A(5);
        echo $a->var1;
    }
}
```

```
}
```

```
$c1 = new A(10);
```

```
$c1->method1();
```

- Сообщение о том, что поле неопределенно
- Сообщение о фатальной ошибке
- 10
- 5
- Ничего

# Массивы в PHP

## Проверка знаний: Массивы в PHP

### Общая группа

Что отобразит следующий код:

```
$arr = array('a' => '27', 'b' => '12', 'd' => '2alpha2');  
$tmpVar = &$arr['c'];  
$tmpVar = 2 / $arr['d'];  
echo $arr['a'] % $arr['b'] + $arr['c'];
```

- Предупреждение: деление на ноль
- Ошибка: элемент с ключом `c` не определен в массиве `$arr`
- 4
- 3.25

Какие значения будет содержать массив `$c` ?

```
$a = array(0=>'a',1=>'b',2=>'c');  
$b = array(3=>'1',4=>'2',0=>'3');  
$c = $a + $b;
```

- появится уведомление об ошибке
- a b c 1 2
- a b c 1 2 3
- 3 b c 1 2

Что выведет следующий код:

```
$input=array (1, 2, 3);  
$result1 = array_pad($input,4,4);
```

```
$result2 = array_pad($result1,4,5);  
foreach ($result2 as $key => $val)  
{  
    echo $val;  
}
```

- 12345
- 1234
- 1235

Что напечатает следующий код:

```
<?php  
$mas = array('a', 1, array(1,4,6));  
echo $mas[2];  
?>
```

- 1
- ничего не выдаст
- ошибка при создании массива с элементами разного типа
- 1, 4, 6
- Array

Укажите верные утверждения о функциях `foreach` и `each` ?

- Функция `foreach... as` возвращает TRUE при достижении конца массива
- Конструкция `foreach... as` является циклом, выполняемым до тех пор, пока не закончатся элементы
- функция `each` возвращает только один элемент
- Функция `each` возвращает TRUE при достижении конца массива

Какой результат выполнения следующего кода?

```
<?php
function
vec_add (&$a, $b)
{
    $a['x'] += $b['x'];
    $a['y'] += $b['y'];
    $a['z'] += $b['z'];
}

$a = array ('x' => 3, 'y' => 2, 'z' => 5);
$b = array ('x' => 9, 'y' => 3, 'z' => -7);

vec_add (&$a, $b);

print_r ($a);
?>
```

- Array ( [x] => 9 [y] => 3 [z] => -7 )
- Array ( [x] => 3 [y] => 2 [z] => 5 )
- Ошибка
- Array ( [x] => 12 [y] => 5 [z] => -2 )
- Ничего из вышеперечисленного

# Практикум по программированию на PHP

## Проверка знаний: Практикум по программированию на PHP

### Общая группа

Что не соответствует правилам в следующем теге XHTML 1.0: `<input type=file name=file size=10>?`

- в синтаксисе данного тега нет ошибок
- теги, не имеющие закрывающего тега, должны быть закрыты использованием сочетания `</>`
- следует указать `input type="file" name="file" size="10"`, все параметры должны быть заключены в кавычки
- следует указать `input type="file" name="file" size=10`
- следует указать `name="file" size="10"`, все параметры должны быть заключены в кавычки

Что выведет следующий код:

```
$var = 'string1string2';  
echo substr_replace($var, 'string3', -7);
```

- выдаст ошибку
- string1string3
- string3string2
- Ничего не напечатает
- string1string3string2

В директории скрипта есть файл 1.jpg. Что случится в результате выполнения кода?

```
<?
  $file = fopen("1.jpg","w");
  if(!file)
  {
    echo("Err");
  }
  else
  {
    fpassthru($file);
  }
?>
```

- Будет выведено: "Err"
- Будет выведено количество символов в файле
- Код ничего не выведет на экран
- Будет выведено: `false`
- Файл `1.jpg` будет урезан до пустой строки
- Файл будет открыт для чтения и записи

В каких единицах вернет время функция `time()` ?

- в микросекундах
- в миллисекундах
- в секундах
- зависит от операционной системы

Какая инструкция `printf` может быть использована для приема строки «Harry Birthday» и вывода строки «Harry»?



- `printf ("%7.5s". "Happy Birthday");`
- `printf ("%*". "Happy Birthday");`
- `printf ("**", 'Happy', 0);`
- `sprintf ("%*". "Happy Birthday");`

Что будет на экране в результате выполнения данного кода?

```
<?php
    $str = "\n\t\x44";
    echo strlen('$str');
?>
```

- 4
- 3
- 5
- 8

# Введение в MySQL

## Проверка знаний: Введение в MySQL

### Общая группа

Для чего нужна точка с запятой в запросах MySQL?

- для завершения команд
- для остановки выполнения команд
- для разделения команд
- для улучшения читаемости кода

Как на локальном хосте создается новый пользователь MySQL с именем `newuser` и паролем `newpass`, которому открыт доступ ко всему содержимому базы данных `newdatabase`?

- `GRANT PRIVILEGES ON newdatabase.* TO 'newuser' IDENTIFIED BY 'newpassword';`
- `GRANT ON newdatabase.* TO 'newuser' IDENTIFIED BY 'newpassword';`
- `GRANT ALL ON newdatabase.* TO 'newuser' IDENTIFIED BY 'newpassword';`

Какие преимущества дает индекс FULLTEXT?

- Полностью аналогичен использованию конструкций `WHERE...=` или `WHERE...LIKE`
- Позволяет запросам, в которых используется естественный язык, находить ключевые слова, если они имеются в графе или графах, проиндексированных в режиме FULLTEXT.
- Позволяет вести поиск, как при работе с обычными поисковыми механизмами Интернета

Оба спецификатора, и `SELECT DISTINCT`, и `GROUP BY`, приводят к отображению только одной строки для каждого значения в столбце, даже если такое значение имеют несколько строк. Какое основное различие между `SELECT DISTINCT` и `GROUP BY`?

- спецификатор `SELECT DISTINCT` воздействует только на отображение, выбирая всего лишь одну строку и исключая все ее дубликаты
- Спецификатор `GROUP BY` объединяет все строки, у которых есть одно и то же значение в графе
- Спецификатор `GROUP BY` отображает всего лишь одну строку и исключая все ее дубликаты
- Спецификатор `SELECT DISTINCT` осуществляет такие операции, как `COUNT`, для подсчета в группе строк

Как можно с помощью инструкции `SELECT...WHERE` вернуть только те строки, в которых в каком-нибудь месте столбца `author` таблицы `classics`, используемой в этой главе, содержится слово `Langhorne`?

- ALTER TABLE classics ADD FULLTEXT(author, "Langhorner");
- ALTER TABLE classics ADD FULLTEXT(author, "%Langhorner%");
- SELECT \* FROM classics WHERE author LIKE "%Langhorner%";
- SELECT \* FROM classics WHERE author LIKE "Langhorner";

Чтобы появилась возможность объединения двух таблиц:

- должна быть общей хотя бы одна графа, например номер ID
- должны быть созданы индексы хотябы для одного из столбцов в каждой таблице
- у обеих таблиц должны быть определены первичные ключи

Пусть у вас есть таблица, поле `id` которой заполняется автоинкрементном. Вы удаляете запись из данной таблицы. Учитывая это, выберите верные утверждения:

- После удаления записи в таблице базы данных, автоинкрементное `id` уже никогда не будет использовано.
- При добавлении, следующая запись будет иметь номер самого большого `id`
- После удаления записи в такой таблице базы данных, автоинкрементное `id` освобождается.
- При добавлении, следующая запись в таблице будет с номером свободного `id`
- При добавлении, следующая запись будет иметь номер самого большого `id + 1`

Выберите из таблицы `brands` уникальный список брендов, названия которых начинаются на букву `N`?

- `SELECT DISTINCT title FROM brands WHERE title = 'N%';`
- `SELECT DISTINCT title FROM brands WHERE title LIKE 'N%';`
- `SELECT DISTINCT title FROM brands WHERE title LIKE 'N';`
- `SELECT title FROM brands ORDER BY brands.title WHERE title LIKE 'N%';`

Пусть есть таблица, содержащая данные о доставленных на склад товарах - идентификатор, наименование товара и цена. Одинаковым ли будет результат выполнения следующих двух запросов:

1. `{.hl.sql}`

```
SELECT productid, name, price FROM mincome; ``
```

1. `{.hl.sql}`

```
SELECT name, price, productid FROM mincome; ``
```

- Поля данных таблиц будут отображены в разной последовательности
- Последовательность указания полей в запросе `SELECT` не имеет значения
- Результаты будут идентичными

Выберите верные утверждения:

- В команде `INSERT ... VALUES` каждому полю, присутствующему в таблице, присваивается значение в виде `имя_поля='значение'`
- К стоповым относятся слова, имеющие настолько широкое распространение, что нет никакого смысла включать их в индекс `FULLTEXT` или использовать при поиске. Но все же они должны участвовать в поиске, когда входят в состав большой строки, заключенной в кавычки.
- Символ звездочки ( `*` ) используется для вывода всех столбцов таблицы
- Команды `SHOW databases` и `SHOW tables` нечувствительны к регистру букв
- В команде `INSERT ... SET` четко указывается порядок следования устанавливаемых полей и их значений
- Тип данных `CHAR(n)` - строка символов переменной длины максимальным размером до `n` символов;

Даны 2 таблицы: `m_income` - товары поступившие на склад (поле `product_id` - идентификатор, `title` - наименование товара), `m_outcome` - проданные со склада товары. Какой из перечисленных запросов верно выведет уникальный список кодов и названий товаров, коды которых есть в таблице `m_income`, но которых нет в таблице `m_outcome` ?

1.

```
SELECT DISTINCT product_id, title
FROM m_income INNER JOIN m_product
ON m_income.product_id=m_product.id
WHERE product_id NOT IN (SELECT product_id FROM m_outcome);
```

2.

```
SELECT DISTINCT product_id, title
FROM m_income INNER JOIN m_product
ON m_income.product_id=m_product.id
WHERE product_id NOT IN m_outcome;
```

3.

```
SELECT *  
FROM m_product  
WHERE id NOT IN (SELECT product_id FROM m_outcome);
```

- 3
- 1
- 2

# Освоение MySQL

## Проверка знаний: освоение MySQL

### Общая группа

Что означает слово отношение (relationship) применительно к реляционным базам данных?

- Хранение и извлечение связанных данных
- Процесс удаления повторяющихся данных и оптимизации таблиц
- Учитывающая отношения база данных - связь между элементами данных, обладающими взаимными ассоциациями

Какое понятие применяется к процессу удаления повторяющихся данных и оптимизации таблиц?

- первичный ключ
- нормализация
- внешний ключ
- отношения

Как формулируются три правила первой нормальной формы?



- в таблице не должно быть никаких повторяющихся граф, содержащих одни и те же типы данных
- каждая из таблиц должна быть связана с любой другой с помощью внешнего ключа
- для уникальной идентификации каждой строки должен использоваться первичный ключ
- все графы должны содержать только одно значение
- любой столбец, который не является ключевым полем, должен зависеть от первичного ключа

Как привести таблицу в соответствие с правилом второй нормальной формы?

- Приводимые таблицы должны быть уже в первой нормальной форме и графы, в которых данные повторяются в нескольких строках, должны быть перемещены в собственные таблицы.
- Данные, не имеющие непосредственной зависимости от первичного ключа, но имеющие зависимость от другого значения в таблице, перемещаются в отдельные таблицы в соответствии с тем, к чему они имеют отношение.
- Графы, в которых данные повторяются в нескольких строках, перемещаются в собственные таблицы.

Выберите верные утверждения:

- Для прекращения транзакции и отмены всех действий выдается команда `ROLLBACK`
- Для создания базы данных, в которой имеется отношение «многие ко многим», нужно создать промежуточную таблицу, содержащую ключи из двух других таблиц. В результате эти две другие таблицы смогут ссылаться друг на друга посредством третьей таблицы.
- Для завершения транзакции и совершения всех действий выдается команда `ROLLBACK`
- Изменить состояние любого запроса можно командой `EXPLAIN`
- При отношении «один ко многим» первичный ключ из таблицы со стороны «один» должен быть добавлен в качестве отдельной графы (внешнего ключа) к таблице на стороне «многие».
- Команда `COMMIT` используется для создания резервной копии базы данных

Для создания в файле `MyBD.sql` резервной копии базы данных `library` необходимо выполнить следующую команду:

```
mysql -u user -u password library > MyBD.sql
```

Дана информация о товарах на складах производственной компании:

Наименование	Стоимость	Склад	Почтовый адрес
Пылесос Karcher	6000	Основной	308907, г. Новосибирск а/я 1234
Пылесос Philips	7000	Основной	308907, г. Новосибирск а/я 1234
Пылесос Panasonic	5000	Основной	308907, г. Новосибирск а/я

Наименование	Стоимость	Склад	Почтовый адрес
			1234
Мультиварка Panasonic	3000	Основной	308907, г. Новосибирск а/я 1234
Фен Philips	2300	Производственный	489308, г. Москва а/я 986
Фен Rowenta	1400	Основной	308907, г. Новосибирск а/я 1234

Приведите таблицу ко **2й нормальной форме**.

## Вариант ответа 1

ID	Наименование	Стоимость	Склад
1	Пылесос Karcher	6000	1
2	Пылесос Philips	7000	1
3	Пылесос Panasonic	5000	1
4	Мультиварка Panasonic	3000	1
5	Фен Philips	2300	2
6	Фен Rowenta	1400	1

ID	Склад	Почтовый адрес
1	Основной	308907, г. Новосибирск а/я 1234
2	Производственный	489308, г. Москва а/я 986

## Вариант ответа 2

ID	Наименование	Производитель	Стоимость
1	1	Karcher	6000
2	1	hilips	7000
3	1	Panasonic	5000
4	2	Panasonic	3000
5	3	Philips	2300
6	3	Rowenta	1400

ID	Наименование	Склад
1	Пылесос	308907, г. Новосибирск а/я 1234
2	Мультиварка	489308, г. Москва а/я 986
3	Фен	08907, г. Новосибирск а/я 1234

## Вариант ответа 3

ID	Наименование	Стоимость	Склад
1	Пылесос Karcher	6000	1
2	Пылесос Philips	7000	1
3	Пылесос Panasonic	5000	1
4	Мультиварка Panasonic	3000	1
5	Фен Philips	2300	2
6	Фен Rowenta	1400	1

ID	Склад	Индекс
1	Основной	308907
2	Производственный	489308

Индекс	Город
308907	1
489308	2

ID	Город
1	Новосибирск
2	Москва

- 1
- 3
- 2

Дана информация о товарах на складах производственной компании:

Наименование	Стоимость	Склад	Почтовый адрес
Пылесос Karcher	6000	Основной	308907, г. Новосибирск а/я 1234
Пылесос Philips	7000	Основной	308907, г. Новосибирск а/я 1234
Пылесос Panasonic	5000	Основной	308907, г. Новосибирск а/я 1234
Мультиварка Panasonic	3000	Основной	308907, г. Новосибирск а/я 1234
Фен Philips	2300	Производственный	489308, г. Москва а/я 986
Фен Rowenta	1400	Основной	308907, г. Новосибирск а/я 1234

Приведите таблицу к **3 нормальной форме**.

## Вариант ответа 1

ID	Наименование	Стоимость	Склад
1	Пылесос Karcher	6000	1
2	Пылесос Philips	7000	1
3	Пылесос Panasonic	5000	1
4	Мультиварка Panasonic	3000	1
5	Фен Philips	2300	2
6	Фен Rowenta	1400	1

ID	Склад	Почтовый адрес
1	Основной	308907, г. Новосибирск а/я 1234
2	Производственный	489308, г. Москва а/я 986

## Вариант ответа 2

ID	Наименование	Производитель	Стоимость
1	1	Karcher	6000
2	1	hilips	7000
3	1	Panasonic	5000
4	2	Panasonic	3000
5	3	Philips	2300
6	3	Rowenta	1400



ID	Наименование	Склад
1	Пылесос	308907, г. Новосибирск а/я 1234
2	Мультиварка	489308, г. Москва а/я 986
3	Фен	08907, г. Новосибирск а/я 1234

## Вариант ответа 3

ID	Наименование	Стоимость	Склад
1	Пылесос Karcher	6000	1
2	Пылесос Philips	7000	1
3	Пылесос Panasonic	5000	1
4	Мультиварка Panasonic	3000	1
5	Фен Philips	2300	2
6	Фен Rowenta	1400	1

ID	Склад	Индекс
1	Основной	308907
2	Производственный	489308

Индекс	Город
308907	1
489308	2

ID	Город
1	Новосибирск
2	Москва

- 3
- 1
- 2

Пусть у вас есть магазин по продаже одежды. *Продавец А* продает 3 рубашки и регистрирует продажу. В то же время, *продавец В* продает 2 рубашки и тоже регистрирует продажу. Какая из предложенных операций будет верна для того, чтобы остаток количества данного товара на складе был зафиксирован верно?

- `START TRANSACTION; UPDATE goods SET col=col-3 WHERE id='tShirt'; UPDATE goods SET col=col-2 WHERE id='tShirt'; COMMIT;`
- `UPDATE goods SET col=col-2 WHERE id='tShirt'; UPDATE goods SET col=col-3 WHERE id='tShirt';`
- Ни 1 из перечисленных
- `UPDATE goods SET col=col-5 WHERE id='tShirt';`

# Доступ к MySQL с использованием PHP

## Проверка знаний: Доступ к MySQL с использованием PHP

### Общая группа

Какая стандартная PHP-функция предназначена для подключения к базе данных MySQL?

- `db_database`
- `db_server`
- `db_connect`
- `mysql_connect`

В каком случае работу функции `mysql_result` нельзя признать оптимальной?

- в случае необходимости проверки значения переменной, содержащей результат
- в случае необходимости отображения по одному полю на каждой строке с дополнительным символом перевода строки между строками
- в том случае, если запрашивается более одной ячейки из базы данных
- если данная функция вызывается многократно

Назовите причину, по которой в большинстве случаев для отправки данных формы лучше воспользоваться методом `POST`, а не методом `GET`.

- поскольку поля отправляются непосредственным образом, а не добавляются к URL-адресу
- т.к. метод `POST` обеспечивает полноценной защиты от вставки поддельных данных
- т.к. метод `POST` не позволяет вставлять поддельные данные в адресную строку
- оба этих метода могут быть использованы абсолютно одинаковым способом

Как определить последнее значение, введенное в столбец со свойством AUTO\_INCREMENT?

- воспользоваться MySQL-функцией `LAST_INSERT_ID()`
- воспользоваться функцией `mysql_insert_id`
- Путем подсчета количества строк в таблице

Какая PHP-функция отключает опасные символы в строке, делая ее пригодной к использованию в MySQL?

- `stripslashes`
- `mysql_entities_fix_string`
- `mysql_fix_string`
- `mysql_real_escape_string`

Какая функция может использоваться для предотвращения атак внедрения при межсайтовом скриптинге?

- с помощью функции `htmlentities`
- с помощью функции `mysql_entities_fix_string`
- с помощью функции `get_magic_quotes_gpc()`
- с помощью функции `mysql_real_escape_string`

Дан следующий фрагмент кода:

```
$query = mysql_query("SELECT COUNT(user_id) FROM users WHERE user_login='".mysql_real_escape_string($_POST['login'])."'");
```

Выберите верное утверждение:

- Переменная `query` будет содержать количество пользователей с логином, введенным в поле "login"
- Переменная `query` будет содержать идентификатор пользователя с именем, введенным в поле "login", если такой логин будет найден
- Переменная `query` будет содержать идентификатор пользователя с именем, введенным в поле "login"
- Переменная `query` будет содержать имя пользователя введенное в поле "login"

# Обработка форм

## Проверка знаний: Обработка форм

### Общая группа

Данные, содержащиеся в форме, могут быть отправлены с использованием одного из двух методов – POST или GET. Какие ассоциативные массивы используются для передачи этих данных PHP-программе?

- Для метода `GET` используется массив `$_GET`
- Для метода `GET` используется массив `$_POST`
- В обоих случаях используется PHP-функция `isset`
- Для метода `POST` используется массив `$_POST`.
- Для метода `POST` используется массив `$isset`

Почему использование `register_globals` не приветствуется?

- Свойство `register_globals` нарушает систему безопасности
- Отправленные данные полей формы автоматически присваиваются в качестве значений PHP-переменным при включенном свойстве `register_globals`
- Свойство `register_globals` является устаревшим

Чем отличаются друг от друга текстовое поле и текстовая область?

- в текстовом поле может быть несколько строк
- в текстовом поле осуществляется перенос слов на новую строку
- в текстовой области может быть несколько строк
- в текстовом поле может содержаться только одна строка
- в текстовой области может содержаться только одна строка

Как из веб-формы отправить группу значений из поля со списком, используя только одно имя поля?

- `$_POST['name']`
- `$_GET['name']`
- `htmlentities($name)`
- `isset($_POST['name'])`

Пусть необходимо создать выпадающий список с годами с 2000 по 2050. Проанализируйте приведенные фрагменты кода и выберите оптимальный и корректный способ реализации поставленной задачи.

1.

```
<select class="input" type="text" name="years">
<option value='2000'>2000</option>
<option value='2001'>2001</option>
<option value='2002'>2002</option>
.....
<option value='2050'>2050</option>
</select>
```

2.

```
<select class="input" type="text" name="years">
<?php
$year = 2000;
for ($i = 0; $i <= 50; $i++)
{
    $new_years = $year + $i;
```



```
    echo '<option value='.$new_years.'>'.$new_years.</option>';
}
?>
</select>
```

3.

```
<select class="submitted" type="text" name="years">
<?php
$year = 2000;
for ($i = 0; $i <= 50; $i++)
{
    $new_years = $year ++;
    echo '<option value='.$new_years.'>'.$new_years.</option>';
}
?>
</select>
```

- 3
- 2
- 1

Пусть, вы получили значения введенные пользователем:

```
$variable = $_POST['user_input']
```

Для того чтобы проверить введенные пользователем данные следует провести проверку. В чем будет различие, при выполнении проверки с помощью функции `strip_tags($variable)` и `htmlentities($variable)` ?

- `strip_tags` полностью очистит введенные данные от HTML
- `htmleentities` используется для того чтобы избавиться от нежелательных слэш-символов
- При выполнении `htmleentities` код интерпретируемого HTML заменяется спецсимволами
- `strip_tags` заменит HTML-код на спец-символы `&lt;; &gt;`
- `htmleentities` используется для полного удаления из строки любого HTML-кода

Пусть у вас есть форма, и вы передаете целочисленные данные, введенные пользователем, с использованием метода POST. Выберите верное утверждение, учитывая данные условия:

- данные, передаваемые через POST будут переданы целым числом
- данные, передаваемые через POST всегда передаются целым числом
- данные, передаваемые через POST всегда передаются тем типом данных, который был указат в значении поля формы
- данные, передаваемые через POST или GET, всегда передаются строкой

# Cookie, сессии и аутентификация

## Проверка знаний: Cookie, сессии и аутентификация

### Общая группа

Почему cookie должны быть переданы в начале работы программы?

- потому что Cookie могут быть считаны только перед загрузкой содержимого веб-страницы
- потому что Cookie отсылаются в виде составной части заголовков
- потому что Cookie отсылаются в начале кода HTML веб-страницы

Где в PHP-программе сохраняются имя пользователя и пароль при использовании HTTP-аутентификации?

- В заданных переменных `$username` , `$password`
- в составной части заголовка `WWW-Authenticate: Basic realm="Restricted Section"`
- в элементах массива `$_SERVER['PHP_AUTH_USER']` и `$_SERVER['PHP_AUTH_PW']`

Дан следующий фрагмент кода:

```
<?php
echo "Hello";
if (SetCookie("Test","Value")) echo "<h3>Cookie успешно установлен!
</h3>";
else echo "<h3>Cookie установить не удалось!</h3>";
?>
```

Что будет выведено на экран в результате его выполнения?

- Cookie успешно установлен!
- Cookie установить не удалось!
- Сообщение об ошибке
- Hello

Дан фрагмент кода:

```
<?php
setcookie("test", "Hello", time()+3600);
echo @$_COOKIE['test'];
?>
```

В какой момент будет выведено сообщение 'Hello'?

- При первом же обращении к скрипту
- При повторном обращении к скрипту
- Никогда
- Через 1 час

Выберите верные утверждения:

- Идентификатор сессии (ID) всегда уникален
- При использовании сессий вся информация хранится на стороне клиента
- Идентификатор сессии (ID) хранится на стороне веб-сервера
- Все переменные сессий доступны из глобального массива `SESSION['var_name']`
- При присвоении какого-либо значения любому полю массива `SESSION['var_name']`, переменная с таким же именем автоматически регистрируется, как переменная сессии,

Пусть у вас есть фрагмент кода, где выполняется несколько операций с сессией, а

именно, устанавливаются переменные сессии, устанавливаются cookie, затем выводится некоторая информация на экран и сессия уничтожается:

```
<?php  
  
session_start();  
  
$a= "";  
  
session_register("a");  
  
<html>  
  
<head>  
  
</head>  
  
...  
  
session_destroy();  
  
>
```

Выберите верные утверждения, исходя из данных условий:

- `session_destroy()` возвращает TRUE при успехе и FALSE при неудаче разрушения данных сессии
- `session_destroy()` не разустанавливает никакие глобальные переменные, ассоциированные с сессией, и не разустанавливает куки cookie.
- Функция `session_start()` должна быть вызвана в тот момент, когда будет необходимо сохранить переменные сессии
- `session_destroy()` разрушает все данные, ассоциированные с текущей сессией, в том числе и глобальные переменные, ассоциированные с сессией, cookie сессии.
- Функция `session_start()` должна быть вызвана до вывода на экран любого html кода

Дан фрагмент кода:

```
<?php  
  
session_name("test");  
  
session_start();  
  
$_SESSION['count'] = @$_SESSION['count'] + 1;  
  
?>  
  
<h2>Счетчик t=</h2> <?=$_SESSION['count']?> раз(a). <br>  
  
?>
```

Если вы обновите страницу браузера, то

- Значение счетчика увеличится на 1
- Ничего не произойдет
- Значение счетчика будет сброшено

Если вы закроете окно браузера, а затем откроете снова, то

- Значение счетчика увеличится на 1
- Значение счетчика будет сброшено
- Ничего не произойдет

# Изучение JavaScript

## Проверка знаний: Изучение JavaScript

### Общая группа

Укажите неверные идентификаторы:

- `idt$q`
- `$12`
- `id:1`
- `ID_qt`
- `12$`
- `12_id`

Выберите все корректные варианты объявления функции в javascript.

- `f()`
- `function f() {}`
- `function: f {}`
- `function=f() {}`
- `f=function() {}`

Что вернет данная функция при вызове?

```
function whatIsReturned() {  
    var total = "one";  
    var TOTAL = "two";  
  
    return total;  
}
```

- JavaScript выдаст ошибку о том, что переменная “total” уже определена.
- undefined
- one
- two

Каким будет значение c после выполнения кода:

```
a = "5";  
b = 2;  
c = a+++b;
```

- 53
- 62
- 8
- 7
- 52
- Ошибка

Каких бинарных операторов нет в JavaScript (выберите все подходящие варианты)?

- %
- >>>
- #
- \*
- +
- !

Дан фрагмент кода:



```
var x = 5;
function myFunction()
{
  document.write ("x="+x) ;
  x = 7;
  document.write ("x="+x) ;
}

document.write ("x="+x) ;
myFunction();
document.write ("x="+x) ;
```

Что будет напечатано в документ?

- 5575
- 5577
- 5557
- 5777

Какие значения выдадут эти два метода при param = 10?

```
function A(param) {
  var m = param;
  m = m + 10;
  return m + param;
}

function B(param) {
  var n = param;
  n = 20;
  return param + n;
}
```

- 30 и 10
- 40 и 30
- 40 и 40
- 30 и 30

Выберите все варианты правильно составленных условных операторов.

- `if i == 5 {}`
- `if (i == 5)`
- `if (i == 5) then {}`
- `unless (i != 5) {}`
- `if i eq 5 then {}`
- `if i == 5 then {}`

Дан следующий код:

```
function Test() { }  
var test = new Test();  
Test.prototype.member='World';  
var test2 = new Test();  
  
var a = test.member;    // 1  
var b = test2.member;   // 2
```

Какие утверждения из перечисленных верны?

- Возникнет исключение в строке 1
- `b` содержит значение 'World'
- `b` содержит значение `undefined`
- `a` содержит значение 'World'
- Возникнет исключение в строке 2
- `a` содержит значение `undefined`

Что произойдет в результате выполнения следующего кода:

```
var obj = new Object();  
alert(obj.a);  
obj.a = 5;  
alert(obj.a);
```

- Произойдет ошибка выполнения скрипта
- Код выведет сначала пустую строку, а потом "5"
- Код выведет сначала "null", а потом "5"
- Код выведет сначала "undefined", а потом "5"
- Код выведет сначала "NaN", а потом "5"

Как с помощью javascript считать свойство `age` объекта `person` (выберите все правильные варианты)?

- `person->age`
- `person['age']`
- `person[age]`
- `person::age`
- `person.age`
- `person["age"]`

Выберите синтаксически правильные варианты создания массива.

- `var a = new [1,2,3];`
- `var a = new Array[1,2,3];`
- `var a = [1,2,3];`
- `var a = new Array();`
- `var a = new Array(1,2,3);`

Укажите корректное утверждение относительно цикла `for`.

- второе выражение в описании цикла обычно определяет условие продолжения выполнения цикла
- второе выражение в описании цикла обычно создает контрольную переменную
- первое выражение в описании цикла обычно определяет условие выхода из цикла
- третье выражение в описании цикла обычно определяет условие выхода из цикла
- первое выражение в описании цикла обычно увеличивает счетчик цикла

Дан код:

```
String..compareTo = function(otherObject) { // код метода };  
  
var i = "abc".compareTo("xyz");
```

Нужно добавить в стандартный JavaScript объект `String` метод `compareTo()`.  
Что должно для этого быть вставлено в пустом поле? Впишите верный ответ.

Выберите кроссбраузерный фрагмент html, который подключает скрипт `mootools.js`.

- `<javascript src="mootools.js"/>`
- `<script type="text/javascript" src="mootools.js">  
</script>`
- `<script src="mootools.js"/>`
- `<script source="mootools.js"/>`

# Использование технологии Ajax

## Проверка знаний: Использование технологии Ajax

### Общая группа

Выберите верные утверждения:

- Ajax GET-запросы используют данные в качестве параметра метода `send`
- Ajax POST-запросы требуют правильной формы заголовков, отправляемых в первоочередном порядке
- Ajax POST-запросы присоединяют данные к URL-адресу
- Ajax POST-запросы передают данные в качестве параметра метода `send`
- Ajax GET-запросы присоединяют данные к URL-адресу

Закончите верно утверждения:

Определить завершение Ajax-вызова можно по значению , которое примет свойство  .

Когда Ajax-вызов успешно завершится, принадлежащее объекту свойство

получит значение  .

Дан `bookbase.xml` файл с информацией о некотором количестве книг, примерно следующего содержания:

```
<?xml version="1.0" encoding="UTF-8" ?>
<bookstore>
<book id='1'>
<title>Гарри Поттер и философский камень </title>
<author>Джоан Ролинг</author>
<cost>500</cost>
```

```

<quantity>15</quantity>
</book>
<book id='2'>
<title>Дракула</title>
<author>Брэм Стокер</author>
<cost>430</cost>
<quantity>5</quantity>
</book>
...
</bookstore>

```

Необходимо, с помощью AJAX наполнить таблицу данными, извлеченными из XML файла представляющего информацию о названии и цене книг. Дополните приведенный ниже фрагмент кода таким образом, чтобы он выполнялся корректно. Впишите необходимый код **без пробелов** в текстовые поля.

```

function extract() {
    var xhttp=new XMLHttpRequest();
    xhttp.onreadystatechange=function() {
        if (xhttp.readyState==4 && xhttp.status==200) {
            var base="<table id='booktab'><tr><th>Название</th>
<th>Стоимость (руб.)</th>";

            var title={{
xhttp.responseXML.getElementsByTagName("title") }};
            var cost={{ xhttp.responseXML.getElementsByTagName("cost")
}};

            for (var i=0;i<book.length;i++){
                base=base+ "<tr>";
                base=base+"<td>"+title[i].{{ childNodes[0].nodeValue
}}+"</td>";
                base=base+"<td>"+cost[i].{{ childNodes[0].nodeValue
}}+"</td>";
                base=base+"</tr>";
            }
            base=base+"</tr></table>";
            document.getElementById('res').innerHTML=base;
        }
    }
    xhttp.open('GET','bookbase.xml',true);
    xhttp.send();
}

```