

Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: ЧУМАЧЕНКО ТАТЬЯНА АЛЕКСАНДРОВНА
 Должность: РЕКТОР
 Дата подписания: 31.08.2022 11:48:21
 Уникальный программный ключ:
 9c9f7aaffa4840d284abe156657b8f85432bdb16



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУрГГПУ»)

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(ОЦЕНОЧНЫЕ СРЕДСТВА)

Шифр	Наименование дисциплины (модуля)
ФТД.В	Программирование с использованием PHP и MySQL в разработке веб-приложений

Код направления подготовки	09.03.02
Направление подготовки	Информационные системы и технологии
Наименование (я) ОПОП (направленность / профиль)	Информационные технологии в образовании
Уровень образования	бакалавр
Форма обучения	очная

Разработчики:

Должность	Учёная степень, звание	Подпись	ФИО
Доцент	кандидат педагогических наук		Паршукова Наталья Борисовна

Рабочая программа рассмотрена и одобрена (обновлена) на заседании кафедры (структурного подразделения)

Кафедра	Заведующий кафедрой	Номер протокола	Дата протокола	Подпись
Кафедра информатики, информационных технологий и методики обучения информатике	Рузаков Андрей Александрович	10	13.06.2019	
Кафедра информатики, информационных технологий и методики обучения информатике	Рузаков Андрей Александрович	1	10.09.2020	

Раздел 1. Компетенции обучающегося, формируемые в результате освоения образовательной программы с указанием этапов их формирования

Таблица 1 - Перечень компетенций, с указанием образовательных результатов в процессе освоения дисциплины (в соответствии с РПД)

Формируемые компетенции			
Индикаторы ее достижения	Планируемые образовательные результаты по дисциплине		
	знать	уметь	владеть
ПК-3 способность выполнять работы по созданию (модификации) и сопровождению информационных систем			
ПК-3.1 Знать принципы организации работ по выявлению и анализу требований к информационным системам, методы оценки и анализа рисков в IT-проектах, принципы планирования и управления IT-проектами.	3.1 Знать сервисы образовательных информационных систем и основные функциональные возможности их сервисов 3.2 Знать принципы разработки информационных систем в виде веб-приложений, способы выявления уязвимостей в IT-проектах		
ПК-3.2 Уметь осуществлять управление работами по созданию (модификации) и сопровождению информационных систем.		У.1 Уметь разрабатывать информационные системы в виде веб-приложений.	

Компетенции связаны с дисциплинами и практиками через матрицу компетенций согласно таблице 2.

Таблица 2 - Компетенции, формируемые в результате обучения

Код и наименование компетенции	
Составляющая учебного плана (дисциплины, практики, участвующие в формировании компетенции)	Вес дисциплины в формировании компетенции (100 / количество дисциплин, практик)
ПК-3 способность выполнять работы по созданию (модификации) и сопровождению информационных систем	
производственная практика (преддипломная)	20,00
Информационные системы поддержки качества образования	20,00
Теория информационных процессов и систем	20,00
Веб-дизайн / Визуальное оформление и верстка сайтов	20,00
Инструментальные средства разработки образовательных технологий	20,00

Таблица 3 - Этапы формирования компетенций в процессе освоения ОПОП

Код компетенции	Этап базовой подготовки	Этап расширения и углубления подготовки	Этап профессионально-практической подготовки
ПК-3	производственная практика (преддипломная), Теория информационных процессов и систем	Информационные системы поддержки качества образования, Веб-дизайн / Визуальное оформление и верстка сайтов, Инструментальные средства разработки образовательных технологий	производственная практика (преддипломная)

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

Таблица 4 - Показатели оценивания компетенций на различных этапах их формирования в процессе освоения учебной дисциплины (в соответствии с РПД)

№	Раздел		
Формируемые компетенции			
	<table> <tr> <th>Показатели сформированности (в терминах «знать», «уметь», «владеть»)</th><th>Виды оценочных средств</th></tr> </table>	Показатели сформированности (в терминах «знать», «уметь», «владеть»)	Виды оценочных средств
Показатели сформированности (в терминах «знать», «уметь», «владеть»)	Виды оценочных средств		
1	Разработка веб-приложения средствами PHP и MySQL		
ПК-3			
Знать знать сервисы образовательных информационных систем и основные функциональные возможности их сервисов Знать знать принципы разработки информационных систем в виде веб-приложений, способы выявления уязвимостей в IT-проектах	Тест		
Уметь уметь разрабатывать информационные системы в виде веб-приложений.	Отчет по лабораторной работе		
2	Разработка веб-приложения средствами PHP и MySQL		
ПК-3			
Знать знать сервисы образовательных информационных систем и основные функциональные возможности их сервисов Знать знать принципы разработки информационных систем в виде веб-приложений, способы выявления уязвимостей в IT-проектах	Тест		
Уметь уметь разрабатывать информационные системы в виде веб-приложений.	Отчет по лабораторной работе Проект		

Таблица 5 - Описание уровней и критериев оценивания компетенций, описание шкал оценивания

Код	Содержание компетенции			
Уровни освоения компетенции	Содержательное описание уровня	Основные признаки выделения уровня (критерии оценки сформированности)	Пятибалльная шкала (академическая оценка)	% освоения (рейтинговая оценка)
ПК-3	ПК-3 способность выполнять работы по созданию (модификации) и сопровождению информационных систем			

Раздел 3. Типовые контрольные задания и (или) иные материалы, необходимые для оценки планируемых результатов обучения по дисциплине (модулю)

1. Оценочные средства для текущего контроля

Раздел: Разработка веб-приложения средствами PHP и MySQL

Задания для оценки знаний

1. Тест:

Пример. Как сделать редирект (автоматическое перенаправление), например, на google.ru, на PHP?

Варианты

```
header("Location: http://google.ru");  
document.location = "http://google.ru";  
location.href = "http://google.ru";  
header("Redirect: http://google.ru");
```

Пример. Программист написал такой код:

```
<?php for ($i = 0; $i < 5; $i++) {  
if ($i % 2 == 0) continue;  
echo $i; } ?>
```

Что он увидит в результате?

Варианты

13
013
024
24

Пример.

Как задаются комментарии в языке PHP?

Варианты

```
/* несколько строк комментариев */  
/ несколько строк комментариев /  
** строка комментариев **  
<!-- несколько строк комментариев - >
```

Приложение 1.

1. Название переменных в PHP чувствительны к регистру. Это выражение..

- 1) Правда
- 2) Ложь

2. Каждое PHP выражение должно заканчиваться..

- 1) Запятой (,)
- 2) Точкой с запятой (;)
- 3) Точкой (.)
- 4) Двоеточие (:)

3. Выберите префикс, который необходим для создания переменных?

- 1) @
- 2) Нет никаких специальных префиксов
- 3) \$
- 4) #

4. \$CustomVariable и \$customvariable это одна и та же переменная. Это выражение...

- 1) Выберите верный php тэг...
- 2) <#php #>
- 3)
- 4) ?..... ?
- 5) <%php %>

5. Как определить константу?

- 1) variable("FOO", "BAR");

- 2) `constant("FOO", "BAR");`
- 3) `define("FOO", "BAR");`
- 4) `defineconstant("FOO", "BAR");`

6. Что будет выведено в результате данного кода

```
<?php
$sport = 'soccer';
$statement = 'He plays $sport';
echo $statement;
?>
```

- 1) He plays \$sport
- 2) He plays soccer

7. Является ли использование сокращённых PHP тегов хорошей практикой?

- 1) Да
- 2) Нет

8. Как правильно писать комментарии?

- 1) `//` это комментарий
- 2) `@@` это комментарий
- 3) `**` это комментарий

9. Является ли переменная `$221BakerStreet` валидной?

- 1) Да
- 2) Нет

10. PHP код не может быть встроен в HTML. Это выражение...

- 1) Ложь
- 2) Правда

11. Какой из следующих операторов используется для конкатенации строк?

- 1) `.`
- 2) `=>`
- 3) `->`
- 4) `&`

12. Какой из перечисленных типов данных не является обычным?

- 1) `Date`
- 2) `String`
- 3) `Integer`
- 4) `Boolean`

13. Какая из этих переменных не является валидной?

- 1) `$my_variable`
- 2) `$myvariable`
- 3) `$_myvariable`
- 4) `$my-variable`

14. Как получить данные POST-запроса?

- 5) через константу `POST`
- 6) через функцию `getPostData()`
- 7) через переменную `$_POST`

15. Каким является язык PHP?

- 1) компилируемым
- 2) интерпретируемым

16. Как присвоить переменной значение?

- 1) `$x === 5`
- 2) `$x => 5`
- 3) `$x = 5`
- 4) `$x == 5`

17. Какая из этих конструкций не относится к циклам?

- 1) isset
- 2) for
- 3) foreach
- 4) while

18. Продолжит ли код выполнение после выражения require 1.php; если файл 1.php не будет найден?

- 1) нет, выведет ошибку и завершится
- 2) да, без каких либо ошибок
- 3) да, но выведет ошибку

19. Что делает оператор continue при вызове внутри цикла?

- 1) останавливает работу цикла
- 2) переводит цикл на следующую итерацию, обрывая выполнение текущей

Задания для оценки умений

1. Отчет по лабораторной работе:

Представить работающий сайт, выполненный на следующих лабораторных работах.

1. Создание каркаса PHP сайта.
2. Создание базы данных для сайта.
3. Создание гостевой книги для сайта.

Задания для оценки владений

Раздел: Разработка веб-приложения средствами PHP и MySQL

Задания для оценки знаний

1. Тест:

Пример. Что выведет данный скрипт:

```
<?php function myfunc() {  
static $id = 0;  
$id++;  
echo $id; }  
myfunc();  
myfunc();  
myfunc(); ?>
```

Варианты

- 123
- 111
- 222
- 333

Пример

С помощью какой функции можно удалить файл?

Варианты

- Unlink()
- Fclose()
- File_delete()

Пример

Каким образом можно уничтожить переменную сессии?

Варианты

session_destroy()

Функция session_unregister(имя_переменной) удаляет глобальную переменную из текущей сессии (т.е. удаляет ее из списка зарегистрированных переменных)

Приложение 1.

#2/

Что выведет данный код:

```
<?php
echo "1" + "2";
?>
```

\$

12

3

1+2

#2/

Как сделать редирект (автоматическое перенаправление), например, на google.ru, на PHP?

\$

```
header("Location: http://google.ru");
document.location = "http://google.ru";
location.href = "http://google.ru";
header("Redirect: http://google.ru");
```

#2/

Что выведет данный скрипт: <?php

```
function myfunc() {
    static $id = 0;
    $id++;
    echo $id;
}
myfunc();
myfunc();
myfunc();
?>
```

\$

123

111

222

333

#2/

Программист написал такой код:

```
<?php
for ($i = 0; $i < 5; $i++) {
    if ($i % 2 == 0) continue;
    echo $i;
}
?>
```

\$

Что он увидит в результате?

\$

13

013

024

24

#2/

Как задаются комментарии в языке PHP?

\$

```
/* несколько строк комментариев */
```

```
/ несколько строк комментариев /
```

```
** строка комментариев **
```

```
<!-- несколько строк комментариев - >
```

#2/

Укажите числовые типы данных, которые имеются в языке PHP?

\$

Float, integer

String

Text

Boolean, integer

Resource, float

#2/

Какая функция предназначена для того, чтобы закрыть соединение с сервером mysql? Какие параметры можно передать в эту функцию?

\$

mysql_close. Параметры: указатель на ресурс, связанный с БД.

mysql_connect. Параметры: адрес сервера, имя пользователя, флаг нового соединения, пользовательские флаги.

mysql_query. Параметры: запрос, указатель на ресурс, связанный с БД.

#2/

В базе данных book в таблице persons хранится следующая информация о человеке: имя (поле first_name), фамилия (поле last_name), адрес электронной почты (поле email). Получить из этой базы данных информацию обо всех людях с фамилией «Иванов». Укажите правильный фрагмент кода для данной задачи.

\$

#2/

Дана команда:

```
if ($var) echo "Hello"; else echo "Bye";
```

В каком случае на экран будет выведено слово «Bye»?

\$

Если \$var == false

Если \$var == true

#2/

Чем отличается оператор break от оператора continue

\$

break заканчивает выполнение текущего цикла, а continue – текущей итерации цикла

break используется для принудительной остановки циклов, а continue – для условных операторов

continue не может быть вызван с дополнительным числовым аргументом, а break – может

#2/

С помощью какой функции можно удалить файл?

\$

Unlink()

Fclose()

File_delete()

#2/

Каким образом можно уничтожить переменную сессии?

\$

session_destroy()

Функция session_unregister(имя_переменной) удаляет глобальную переменную из текущей сессии (т.е. удаляет ее из списка зарегистрированных переменных)

#2/

Какие тэги были использованы при создании формы ниже?

\$

<input type='text'>, <input type='checkbox'> и <input type='radio'>

<input type='textfield'>, <input type='checkbox'> и <input type='radiobutton'>

<input type='text'> и <input type='check'>

<input type='text'>, <input type='check'> и <input type='radio'>

#2/

Укажите тег HTML позволяющий определить поле для ввода пароля.

\$

<input type='password' />

<select type='password' />

<input name='password' />

<input type='text' name='password' />

<password>

#2/

Укажите запрос MySQL, который выбирает одну запись из таблицы guest и удаляет ее. Учсть, что переменная \$id определена и содержит числовое значение.

\$

DELETE FROM guest WHERE id = '\$id'

SELECT * FROM guest WHERE id = '\$id'

DELETE FROM guest

DELETE * FROM guest WHERE id = '\$id'

#2/

Таблица articles содержит поля id, title (название), description (описание), text, author (автор), data (в формате YYYY-MM-DD, т.е. 2011-11-04 – 4 ноября 2011 г.).

Укажите запрос MySQL, который выбирает из таблицы articles все записи, поле автор которых Петров, и записи созданы после 1.02.2012.

\$

```
SELECT * FROM articles WHERE data > '2012-02-01' AND author = 'Петров'
```

```
SELECT * FROM articles WHERE id > '1.02.2012' AND author = 'Петров'
```

```
SELECT id, author, * FROM articles WHERE id > '1.02.2012' AND author = 'Петров'
```

```
SELECT * FROM articles WHERE data > '2012-01-02' AND author = 'Петров'
```

#2/

Укажите условные конструкции языка программирования PHP

\$

If, if ... else ..., switch

If, if ... then ... else, case

If, foreach

If, do, while

#2/

Укажите правильный формат оператора выбора в PHP

\$

```
switch (выражение или переменная){  
    case значение1: блок_действий1; break;  
    case значение2: блок_действий2; break;  
    default: блок_действий_по_умолчанию; break;  
}
```

```
case (выражение или переменная){  
    switch значение1: блок_действий1; break;  
    switch значение2: блок_действий2; break;  
    default: блок_действий_по_умолчанию;  
}
```

```
switch (выражение или переменная){  
    case значение1: блок_действий1;  
    case значение2: блок_действий2;  
    default: блок_действий_по_умолчанию  
}
```

```
switch (выражение или переменная){  
    case значение1: блок_действий1; die();  
    case значение2: блок_действий2; die();  
    default: блок_действий_по_умолчанию; die();  
}
```

#2/

Выберите НЕВЕРНОЕ описание цикла с предусловием

\$

while (выражение): блок_выполнения end;

while (выражение) { блок_выполнения }

while (выражение): блок_выполнения endwhile;

while (выражение) оператор;

#2/

Выберите ВЕРНОЕ описание цикла с параметром

\$

```
for (начальное_выражение; условие; выражение_управления_циклом)
```

```
{
```

```
// последовательность операторов
```

```
}
```

```
foreach (начальное_выражение; условие; выражение_управления_циклом)
```

```
{
```

```
// последовательность операторов
```

```
}
```

```
while (начальное_выражение; условие; выражение_управления_циклом)
```

```
{
```

```
// последовательность операторов
```

```
}
```

#2/

Есть ли разница между одинарными и двойными кавычками при интерпретации значения литералов строкового типа?

\$
Да
Нет
#2/

Что напечатает следующий код:

\$
1
10
9
0
#2/

Необходимо перенаправить пользователя средствами PHP на другую страницу, но при тестировании данного кода Вы заметили, что при работе программы возникает ошибка.

```
<?php
    echo "Сейчас вы будете перенаправлены на другую страницу";
    header("Location: http://example.com");
?>
```

\$
Функцию header() необходимо вызывать до отправки любого вывода
Функции header() не существует. Необходимо использовать функцию headers_list()
Функции header() имеет второй обязательный параметр, который необходимо указать в данном фрагменте кода
Функции echo не существует. Необходимо использовать другую функцию
Функции header() не существует. Необходимо использовать функцию headers_sent()

#2/
Какие массивы называют ассоциативными?
\$

те, у которых в качестве индексов используются строки
те, у которых в качестве индексов используются дробные числа
те, у которых в качестве индексов используются идентификаторы переменных
те, у которых в качестве индексов используются целые числа

#2/
Сколько разных типов циклов есть в PHP?

\$
4
3
2
1
#2/

Какая функция PHP используется для удаления файла?

\$
unlink()
remove()
del()
move()
unset()

#2/
Дана форма

В каком массиве будут храниться данные после сабмита формы на сервер?

\$
\$_POST[]
\$_GET[]
\$_FORM[]
\$_FORMS[]
\$POST[]
\$GET[]
#2/

Где обычно выполняются скрипты, написанные на PHP?

\$
На сервере
Внутри операционной системы
На клиенте
В окне браузера

#2/

Как правильно вызвать функцию func с одним параметром?

\$

func(2);

invoke func(3);

func(param=4);

call func(1);

#2/

Для чего предназначена функция isset()?

\$

Она проверяет, была ли инициализирована переменная

Она проверяет, существует ли массив

Она проверяет, существует ли объект

Ничего из вышеперечисленного

#2/

Укажите оператор включения кода из другого файла

\$

include

\$include

\$isset

Isset

require_one

#3/

Соотнесите оператор включения кода из другого файла с его особенностями

\$

include

include_once

require

require_once

\$

позволяет включать код, содержащийся в указанном файле, и выполнять его столько раз, сколько программа встречает этот оператор

если код из файла уже был включен, то его нельзя

включить повторно

позволяет включать в программу и исполнять какой-либо файл, при этом, если подключить файл не удалось выдается фатальная ошибка прекращающая дальнейшие действия алгоритма

позволяет включать в программу и исполнять какой-либо файл только один раз, при этом, если подключить файл не удалось выдается фатальная ошибка прекращающая дальнейшие действия алгоритма

Задания для оценки умений

1. Отчет по лабораторной работе:

Представить работающие модули для сайта по результатам выполнения следующих лабораторных работ:

1. Создание алгоритма авторизации пользователя на сайте.
2. Создание тестирования на сайте.
3. Разработка фотогалереи на сайте.

2. Проект:

Примеры индивидуальных проектов для своего сайта (каждому студенту свой вариант).

Вариант 1. Спроектировать базу данных для хранения комментариев на сайте.

Реализовать алгоритм комментирования статьи на сайте. Во фронтальной части проекта должен выводиться полный список постов, отсортированный по дате в обратном порядке. Выводиться должны только те сообщения, которые были опубликованы администратором (т.е. в отзывах присутствует модерация). С левой стороны должна находиться форма авторизации пользователей. Если авторизацию прошел пользователь с именем admin, то он может публиковать сообщения, удалять сообщения, а также редактировать их. Все остальные пользователи, в т.ч. и неавторизованные, могут оставлять сообщения на сайте.

Также, в проекте должна быть реализована система оценивания прочитанного материала по 5-ти балльной шкале и вывод средней оценки всех опубликованных комментариев.

Вариант 2. Спроектируйте базу данных для хранения информации о пользователях на сайте. Разработайте алгоритм формы для регистрации пользователей. Форма должна включать проверку правильности заполнения полей (имя, ник, email, пароль, подтверждение пароля). Если пользователь вводит логин, который уже зарегистрирован, то должна выдаваться информация об ошибке сразу же (без отправки данных с формы обычным образом). То же самое, если пароль и подтверждение пароля не совпадут. Далее должна происходить стандартная регистрация с оповещением по email (подтверждение регистрации по ссылке делать необязательно). Также должна присутствовать ссылка на восстановление пароля.

Вариант 3. Спроектируйте на сайте базу данных для хранения отзывов на сайте. Реализуйте алгоритм, по которому зарегистрированный пользователь может оставить отзыв о сайте и поставить оценку по 5-ти балльной шкале. Отзывы не модерируются, а публикуются как есть. При отправке отзыва владельцу сайта (администратору сайта) должно приходить письмо текстом отзыва. На главной странице сайта должны размещаться 3 случайных отзыва и ссылка «Посмотреть все отзывы» с переходом на страницу всех отзывов. Самые последние отзывы должны выводиться в начале списка. Изменить алгоритм поиска, предложенный на лабораторном занятии так, чтобы он работал иском и по контрольным вопросам.

Приложение 1.

Индивидуальные задания

1. Спроектируйте базу данных для хранения фотографий на сайте. Бородин В.

Во фронтальной части проекта должен быть выведен список опубликованных фотографий (можно сделать в виде красивой галереи или слайдера JavaScript). В бэкенде (панели администратора) проекта должна быть возможность просматривать список всех фото с небольшими эскизами фотографий, редактировать фото (загружать новое, изменять название фото). Также необходимо реализовать механизм публикации фото для фронтэнда проекта (сайта), удаление фотографии (и в БД и физически с сервера) и пересортировку.

2. Спроектировать базу данных для хранения комментариев на сайте.

Реализовать алгоритм комментирования статьи на сайте. Во фронтальной части проекта должен выводиться полный список постов, отсортированный по дате в обратном порядке. Выводиться должны только те сообщения, которые были опубликованы администратором (т.е. в отзывах присутствует модерация). С левой стороны должна находиться форма авторизации пользователей. Если авторизацию прошел пользователь с именем admin, то он может опубликовывать сообщения, удалять сообщения, а также редактировать их. Все остальные пользователи, в т.ч. и неавторизованные, могут оставлять сообщения на сайте.

Также, в проекте должна быть реализована система оценивания прочитанного материала по 5-ти балльной шкале и вывод средней оценки всех опубликованных комментариев.

3. Спроектируйте базу данных для хранения информации о пользователях на сайте. Разработайте алгоритм формы для регистрации пользователей. Форма должна включать проверку правильности заполнения полей (имя, ник, email, пароль, подтверждение пароля). Если пользователь вводит логин, который уже зарегистрирован, то должна выдаваться информация об ошибке сразу же (без отправки данных с формы обычным образом). То же самое, если пароль и подтверждение пароля не совпадут. Далее должна происходить стандартная регистрация с оповещением по email (подтверждение регистрации по ссылке делать необязательно). Также должна присутствовать ссылка на восстановление пароля.

4. Спроектируйте на сайте базу данных для хранения отзывов на сайте.

Реализуйте алгоритм, по которому зарегистрированный пользователь может оставить отзыв о сайте и поставить оценку по 5-ти балльной шкале. Отзывы не модерируются, а публикуются как есть. При отправке отзыва владельцу сайта (администратору сайта) должно приходить письмо текстом отзыва. На главной странице сайта должны размещаться 3 случайных отзыва и ссылка «Посмотреть все отзывы» с переходом на страницу всех отзывов. Самые последние отзывы должны выводиться в начале списка.

5. Спроектируйте базу данных для хранения опросов на сайте. Разработайте алгоритм, по которому пользователь может выбрать опрос из списка, ответить на этот опрос и получить статистическую информацию об этом опросе (сколько человек прошли опрос, как распределились ответы на опрос). Разработайте админ-панель для опросов, с помощью которой можно редактировать, публиковать опросы, а также просматривать статистику по опросам.

6. Спроектируйте базу данных для хранения цитат на сайте. Разработайте алгоритм, с помощью которого на вашем ресурсе будет выводиться случайная цитата ученого, философа или писателя. Предусмотреть вывод цитаты в виде тега <blockquote>, фамилию ученого и год, когда эта фраза была озвучена. Разработайте админ-панель для редактирования, сортировки и публикации цитат ученых. Котельников

7. Спроектируйте и реализуйте алгоритм тестирования по всем имеющимся в базе данных контрольным вопросам. В выполненных лабораторных работах контрольные вопросы выводятся для каждой статьи. Добавьте возможность пройти тест по всем имеющимся вопросам, причем для каждого вопроса должна указываться тема статьи/учебного материала, на проверку которого он направлен. Продумайте и реализуйте админ-панель для добавления, редактирования и публикации вопросов, а также прикрепления их к нужной статье/учебному материалу.

8. Спроектируйте базу данных для хранения новостей на сайте. Реализуйте алгоритм, выводящий 3 последних новости и ссылку «Все новости». При переходе по ссылке «Все новости» должна быть возможность посмотреть полный список опубликованных новостей. При этом должны быть доступны параметры сортировки: сортировка по возрастанию, сортировка по убыванию. Разработайте админ-панель для редактирования, добавления, удаления и публикации новостей. Кузнецова В.
9. Спроектируйте базу данных для хранения прикрепленных материалов. К каждой статье можно прикреплять неограниченное количество материалов: архивов, презентаций, текстовых документов и др. Все ссылки должны открываться в новом окне. Разработайте удобную админ-панель для загрузки прикрепленных материалов на сервер и прикрепления их к статьям/учебным материалам.
10. Разработайте поисковую систему на сайте. Поиск должен осуществляться по имеющимся таблицам в базе данных: поиск по статьям, контрольным вопросам, гостевой книги и пр. При выводе результатов должен показываться поисковый запрос, количество найденных результатов и список результатов с информацией, в какой таблице был найден материал (статьи, контрольные вопросы, гостевая книга и пр.)
11. Разработайте блок на сайте для вывода статистической информации. Это может быть количество статей на сайте, количество пользователей, самый популярный опрос, пользователь с самыми высокими баллами за тестирование, средний балл за тестирование по текущей статье/учебному материалу.
12. Спроектируйте специальный раздел для авторизованного пользователя «Личный кабинет пользователя». В нем должна содержаться информация обо всех пройденных тестах, количестве набранных баллов, сравнение со средним значением по данному тесту. Также в личном кабинете пользователя должна содержаться информация о его сообщениях в гостевой книге.
13. Спроектируйте базу данных для хранения часто задаваемых вопросов на сайте. Разработайте алгоритм, по которому выводятся часто задаваемые вопросы на сайте и ответы на них. Продумайте стилизацию вопросов и ответов. Разработайте форму, через которую пользователь может отправить свой вопрос. После отправки вопроса программа должна отправлять 2 сообщения для пользователя и админа, о том, что вопрос получен. Разработайте админ-панель для создания ответа на вопрос и публикации его на сайте.
14. Спроектируйте базу данных для хранения дополнительной литературы на сайте. Разработайте алгоритм, согласно которому на странице со статьёй будет выводиться необходимая литература. Для литературы должны выводиться: автор, название, издательство, год издания, количество страниц. Разработайте админ-панель для редактирования списка литературы к статье/учебному материалу.
15. Спроектируйте базу данных для хранения писем рассылки.
Разработайте алгоритм, который отправляет на email всех пользователей писем с приглашением на ваш ресурс (изучить новый материал, пройти тестирование, прочитать новые сообщения гостевой книги и пр.). В скрипте с помощью специальных тегов отметьте имя (ник) пользователя. Пример: Уважаемый(ая) Елена! Шаблон текста сообщения должен сохраняться в базе данных. Его можно использовать многократно.

Задания для оценки владений

2. Оценочные средства для промежуточной аттестации

1. Зачет по факультативу

Вопросы к зачету:

1. В чем отличие понятия «сайт» от понятия «портал». Привести примеры.
2. Приведите примеры сервисов образовательных web-ресурсов.
3. Какие требования предъявляются к сайтам образовательных учреждений?
4. Какими документами регламентируется структура сайта образовательного учреждения?
5. Особенности языка программирования PHP.
6. Суть клиент-серверного взаимодействия.
7. Переменные, типы переменных, операции с переменными в PHP
8. Понятие формы HTML
9. Типы элементов HTML форм и их синтаксис
10. Методы передачи данных с HTML форм на сервер: GET и POST.
11. Команды вывода информации echo, print_r, print_f
12. Основные алгоритмические конструкции языка программирования PHP и их синтаксис.
13. Операторы безусловного перехода в PHP
14. Инструментальные системы для работы с базами данных
15. Функции инструментальных систем для работы с базами данных
16. Особенности системы управления базами данных MySQL.
17. Виды запросов к базе данных.
18. С помощью каких функций происходит подключение к базе данных MySQL из PHP скрипта?
19. С помощью какой утилиты можно управлять базой данных MySQL?

20. Опишите алгоритм подключения к базе данных из PHP.

2. Зачет по факультативу

Вопросы к зачету:

1. Глобальные, локальные, статические переменные в PHP
2. Функции для работы с файлами в PHP
3. Функции для работы со строками в PHP
4. Способы описания и вызов функций в PHP
5. Как происходит авторизация на сайте?
6. Какую роль играет сессия при авторизации пользователя на сайте.
7. Опишите алгоритм проверки правильности ответа на тест пользователя.
8. Что позволяет сделать функция header ("location:...")
9. С помощью какой функции происходит подключение кода другого файла?
10. Для чего предназначены сессии?
11. Каково назначение cookie?
12. Какие типы запросов может выполнять MySQL?
13. Что понимается под версткой страницы?
14. Перечислите инструментальные средства обеспечения безопасности в базах данных?
15. Каким образом осуществляется резервное копирование баз данных?
16. Опишите реализацию алгоритма поиска на сайте.
17. Опишите реализацию алгоритма авторизации на сайте.
18. Опишите реализацию алгоритма тестирования на сайте.
19. Опишите реализацию алгоритма загрузки файлов на сервер.
20. Опишите реализацию алгоритма админ-панели для сайта.

Раздел 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

1. Для текущего контроля используются следующие оценочные средства:

1. Отчет по лабораторной работе

При составлении и оформлении отчета следует придерживаться рекомендаций, представленных в методических указаниях по выполнению лабораторных работ по дисциплине.

2. Проект

Проект – это самостоятельное, развёрнутое решение обучающимся, или группой обучающихся какой-либо проблемы научно-исследовательского, творческого или практического характера.

Этапы в создании проектов.

1. Выбор проблемы.
2. Постановка целей.
3. Постановка задач (подцелей).
4. Информационная подготовка.
5. Образование творческих групп (по желанию).
6. Внутригрупповая или индивидуальная работа.
7. Внутригрупповая дискуссия.
8. Общественная презентация – защита проекта.

3. Тест

Тест это система стандартизированных вопросов (заданий), позволяющих автоматизировать процедуру измерения уровня знаний и умений обучающихся. Тесты могут быть аудиторными и внеаудиторными. Преподаватель доводит до сведения студентов информацию о проведении теста, его форме, а также о разделе (теме) дисциплины, выносимой на тестирование.

При самостоятельной подготовке к тестированию студенту необходимо:

- проработать информационный материал по дисциплине. Проконсультироваться с преподавателем по вопросу выбора учебной литературы;
- выяснить все условия тестирования заранее. Необходимо знать, сколько тестов вам будет предложено, сколько времени отводится на тестирование, какова система оценки результатов и т.д.
- работая с тестами, внимательно и до конца прочесть вопрос и предлагаемые варианты ответов; выбрать правильные (их может быть несколько); на отдельном листке ответов выписать цифру вопроса и буквы, соответствующие правильным ответам. В случае компьютерного тестирования указать ответ в соответствующем поле (полях);
- в процессе решения желательно применять несколько подходов в решении задания. Это позволяет максимально гибко оперировать методами решения, находя каждый раз оптимальный вариант.
- решить в первую очередь задания, не вызывающие трудностей, к трудному вопросу вернуться в конце.
- оставить время для проверки ответов, чтобы избежать механических ошибок.

2. Описание процедуры промежуточной аттестации

Оценка за зачет/экзамен может быть выставлена по результатам текущего рейтинга. Текущий рейтинг – это результаты выполнения практических работ в ходе обучения, контрольных работ, выполнения заданий к лекциям (при наличии) и др. видов заданий.

Результаты текущего рейтинга доводятся до студентов до начала экзаменационной сессии.

Цель зачета – проверка и оценка уровня полученных студентом специальных знаний по факультативу и соответствующих им умений и навыков, а также умения логически мыслить, аргументировать избранную научную позицию, реагировать на дополнительные вопросы, ориентироваться в массиве информации. Подготовка к зачету начинается с первого занятия по факультативу, на котором обучающиеся получают предварительный перечень вопросов к зачёту и список рекомендуемой литературы, их ставят в известность относительно критериев выставления зачёта и специфике текущей и итоговой аттестации. С самого начала желательно планомерно осваивать материал, руководствуясь перечнем вопросов к зачету и списком рекомендуемой литературы, а также путём самостоятельного конспектирования материалов занятий и результатов самостоятельного изучения учебных вопросов.

По результатам сдачи зачета выставляется оценка «зачтено» или «не зачтено».