

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
БОРИСОГЛЕБСКИЙ ФИЛИАЛ
(БФ ФГБОУ ВО «ВГУ»)

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ
Основы РНР

1. Код и наименование направления подготовки:

44.03.01 Педагогическое образование

2. Профиль подготовки:

Информатика и информационные технологии в образовании

3. Квалификация (степень) выпускника:

Бакалавр

4. Форма обучения:

Заочная

5. Кафедра, отвечающая за реализацию дисциплины:

кафедра прикладной математики, информатики, физики и методики их преподавания

6. Составители:

Позднова Е.А., кандидат педагогических наук, доцент,

Хвостов М.Н., кандидат физико-математических наук

7. Методические указания для обучающихся по освоению дисциплины

Приступая к изучению учебной дисциплины, прежде всего обучающиеся должны ознакомиться с учебной программой дисциплины. Электронный вариант рабочей программы размещён на сайте БФ ВГУ.

Обучающиеся должны иметь четкое представление о:

- перечне и содержании компетенций, на формирование которых направлена дисциплина;
- основных целях и задачах дисциплины;
- планируемых результатах, представленных в виде знаний, умений и навыков, которые должны быть сформированы в процессе изучения дисциплины;
- количестве часов, предусмотренных учебным планом на изучение дисциплины, форму промежуточной аттестации;
- количестве часов, отведенных на контактную и самостоятельную работу;
- формах контактной и самостоятельной работы;
- структуре дисциплины, основных разделах и темах;
- системе оценивания ваших учебных достижений;
- учебно-методическом и информационном обеспечении дисциплины.

Знание основных положений, отраженных в рабочей программе дисциплины, поможет обучающимся ориентироваться в изучаемом курсе, осознавать место и роль изучаемой дисциплины, строить свою работу в соответствии с требованиями, заложенными в программе.

Основными формами контактной работы по дисциплине являются лекции и лабораторные работы, посещение которых обязательно для всех студентов.

В ходе лекционных занятий следует не только слушать излагаемый материал и кратко его конспектировать, но очень важно участвовать в анализе примеров, предлагаемых преподавателем, в рассмотрении и решении проблемных вопросов, выносимых на обсуждение. Необходимо критически осмысливать предлагаемый материал, задавать вопросы как уточняющего характера, помогающие уяснить отдельные излагаемые положения, так и вопросы продуктивного типа, направленные на расширение и углубление сведений по изучаемой теме, на выявление недостаточно освещенных вопросов, слабых мест в аргументации и т.п.

В ходе выполнения лабораторных работ студент выполняет задания, содержащиеся в методическом пособии дисциплины в соответствии с имеющимися указаниями. Далее студент самостоятельно выполняет индивидуальное задание.

Обязательно следует познакомиться с критериями оценивания каждой формы контроля – это поможет избежать недочетов, снижающих оценку за работу.

При подготовке к промежуточной аттестации необходимо повторить пройденный материал в соответствии с учебной программой, примерным перечнем вопросов, выносящихся на зачет. Рекомендуется использовать конспекты лекций и источники, перечисленные в списке литературы в рабочей программе дисциплины, а также ресурсы электронно-библиотечных систем. Необходимо обратить особое внимание на темы учебных занятий, пропущенных по разным причинам. При необходимости можно обратиться за консультацией и методической помощью к преподавателю.

8. Методические материалы для обучающихся по освоению теоретических вопросов дисциплины

№	Тема лекции	Рассматриваемые вопросы
1	Принципы работы Интернета и основы создания Web-приложений.	Принципы работы Интернета. Архитектура современных Web-приложений. Программирование на стороне клиента и сервера. Обзор подходов, инструментов и технологий создания web-приложений.
2	Программирование на стороне сервера.	Протоколы передачи данных. Протокол HTTP. HTTP- сервер Apache. Понятие и структура интерфейса CGI. Методы передачи

		параметров между страницами (GET, POST). Передача параметров серверу. Запоминание состояния. Меры безопасности. CGI и базы данных
3	Основы программирования на языке PHP	Назначение, принципы работы языка PHP и его преимущества. Синтаксис и грамматика. Типы данных PHP. Переменные языка PHP. Элементы языка PHP: константы и выражения; функции; классы; операторы; регулярные выражения. Циклы. Работа со строками, массивами. Встроенные функции PHP. Объектно-ориентированное программирование в PHP. Шаблоны. Работа с файлами и базами данных.
4	Web-дизайн.	Правила хорошего тона и модели поведения в Web-дизайне. Цветовые решения, оформление графики и текста. Форматы изображений, создание и оптимизация изображений для Web-сайта. Методы размещения изображений на Web-сайте. Web-анимация. Баннеры. Favicon - иконки Web-сайта.
5	Управление сессиями. Обеспечение безопасности.	Сессии и Cookies. Способы авторизации доступа.

9. Методические материалы для обучающихся по подготовке к лабораторным занятиям

№	Тема занятия	Рассматриваемые вопросы
1	Программирование на стороне сервера.	Основы языка PHP. Математические и строковые функции.
2	Основы программирования на языке PHP	Условные операторы языка PHP. Циклы. Массивы. Функции пользователя.
3	Управление сессиями. Обеспечение безопасности.	Применение файлов для хранения данных. Использование баз данных для хранения информации.

10. Тематика рефератов/докладов/эссе, методические рекомендации по выполнению контрольных и курсовых работ, иные материалы

Примерный перечень вопросов к зачету по дисциплине «Основы PHP»

1. Типы данных языка PHP. Скалярные типы данных.
2. Типы данных языка PHP. Строковый тип данных.
3. PHP-скрипты. Комментарии.
4. Переменные.
5. Арифметические, строковые, логические операторы.
6. Операторы присваивания и сравнения.
7. Оператор присваивания. Присваивание по значению и по ссылке.
8. Константы.
9. Тип данных массив.
10. Оператор вывода.
11. Условные операторы.
12. Циклы. While.
13. Циклы. Do ... while.
14. Циклы. For.
15. Циклы. foreach.
16. Операторы передачи управления.
17. Функции, определяемые пользователем.

18. Операторы включения.
19. Суперглобальные массивы.
20. Передача данных на сервер.
21. Работа с файлами без применения дескрипторов.
22. Работа с файлами с применением дескрипторов.
23. Работа с файловой системой.
24. Суперглобальный массив `$_FILES`. Загрузка файлов на сервер.
25. Основные функции для работы с базами данных.

**Примерный перечень индивидуальных заданий по дисциплине
«Основы PHP»**

1. Гостевая книга.
2. Сборник задач.
3. Магазин.
4. Каталог книг.
5. Голосование.
6. Каталог продукции.
7. Сборник тестов.
8. Статистика посещения страниц сайта.
9. Каталог фотографий.
10. Каталог музыки.

**Примерный вариант заданий по дисциплине
«Основы PHP»**

1. $4R \cdot \sin \frac{a}{2} \cdot \sin \frac{b}{3} \cdot \sin \frac{c}{4}$
2. $\sqrt{a^2 + b^2 - 2ab \cos c}$
3. Дана сторона квадрата. Найти его периметр и площадь.
4. Найти площадь кольца, ограниченного окружностями с заданными радиусами.
5. Дано расстояние в метрах. Определить полное число километров в нем.
6. Дано двузначное число. Найти сумму его цифр.
7. Вычислить значение логического выражения $\neg(A \vee B) \wedge (A \vee \neg B \vee C)$.
8. Даны радиус круга и сторона квадрата. У какой фигуры площади больше.
9. Дано натуральное число. Определить, делится ли оно на 7, 11, 13. Сообщения о делимости вывести разным цветом.
10. Дано три числа. Вывести их в виде нумерованного списка в порядке возрастания.
11. Дано трехзначное число. Подсветить розовым цифры 1, 3, 7.
12. Дано 3 целых числа. Определить какие из них являются четными и определить для них гиперссылку на страницу, содержащую сведения о А.С. Пушкине.
13. Составить программу, которая в зависимости от порядкового номера месяца выводит время года.
14. Составить программу, которая в зависимости от порядкового номера месяца выводит его название.
15. Вывести на экран первых 10 членов арифметической прогрессии с первым членом 1,2 и разностью 0,33.
16. Вычислить сумму ряда $\frac{1}{n^2}$ с точностью 0,001. Вывести все суммируемые члены ряда в виде маркированного списка. После чего вывести сумму, снабдив информацией, необходимыми комментариями.

17. Вывести на экран таблицу умножения размером 16 на 16 в шестнадцатеричной системе счисления.
18. Дан ΔABC , с вершинами $A(2; -1; 3)$, $B(2; -1; 11)$, $C(5; 3; 3)$. Определить площадь ΔABC , используя формулу Герона $S = \sqrt{p \cdot (p-a) \cdot (p-b) \cdot (p-c)}$, $p = \frac{a+b+c}{2}$, a , b , c – длины сторон AB , BC , AC , $|AB| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2}$. Точки задать в виде одномерных массивов. Для вычисления длин сторон записать функцию, от двух массивов.
19. Записать функцию, которая преобразует число в массив его цифр.
20. Записать функцию формирования массива случайных чисел, в качестве аргументов принимающая имя массива, размер массива, границы для поиска случайных чисел.
21. Записать функцию вывода одномерного массива в виде нумерованного списка.
22. Записать функцию вывода двумерного массива в виде таблицы.
23. Разработать форму и скрипт для авторизации на сайте.
24. Разработать скрипт, формирующий массив сведений (не менее 20 записей), соответствующих одному из вариантов индивидуальных заданий. Обеспечить доступ к сведениям только после успешной авторизации. Вывести сведения на экран в виде таблицы, первым столбцом которой должен быть номер по порядку. Разработать скрипт для фильтрации элементов массива и вывода их в определенном порядке.
25. Разработать скрипт, позволяющий выводить сведения, хранящиеся в файле, об объектах соответствующих одному из вариантов индивидуальных заданий. Обеспечить доступ к сведениям только после успешной авторизации. Вывести сведения на экран в виде таблицы, первым столбцом которой должен быть номер по порядку. Разработать скрипт для фильтрации элементов массива и вывода их в определенном порядке.

**Ключевые примеры скриптов, рассматриваемых при изучении дисциплины
«Основы PHP»**

Комментарии

```
<?php
// Это однострочный комментарий в стиле C++
# Это комментарий в стиле Unix
/*
    Это многострочный комментарий.
    Здесь можно написать несколько строк.
    При исполнении программы все, что находится
    здесь, будет игнорировано.
*/
?>
```

Присвоение переменной по значению

```
<?php
$first = ' Text ';
$second = $first;
$first = ' New text ';
echo "Переменная с именем first ". "равна $first<br>";
echo "Переменная с именем second ". "равна $second";
```

```
?>
```

Присвоение переменной по ссылке

```
<?php
    $first = ' Text ';
    $second = &$first;
    $first = ' New text ';
    echo "Переменная с именем first "."равна $first<br>";
    echo "Переменная с именем second "."равна $second";
```

```
?>
```

Использование констант

```
<?php
    // определяем константу PASSWORD
    define("PASSWORD","qwerty");
    // определяем регистронезависимую константу PI
    // со значением 3.14
    define("PI","3.14", True);
    // выведет значение константы PASSWORD
    echo PASSWORD;
    // тоже выведет значение константы PASSWORD
    echo constant("PASSWORD");
    // выведет password и предупреждение,
    // поскольку мы ввели регистрозависимую константу
    echo password;
    // выведет 3.14, поскольку константа PI регистронезависима
    echo pi;
```

```
?>
```

Способы объявления массива

```
<?php
    $books = array ("php" => 1, 12 => "PHP");
    echo $books["php"]; // выведет 1
    echo $books[12]; // выведет "PHP"
```

```
?>
```

```
<?php
    // массивы $arr1, $arr2 и $arr3 эквиваленты
    $arr1 = array(5 => 43, 32, 56, "b" => 12);
    $arr2 = array("5" => 43, 32, 56, "b" => 12);
    $arr3 = array(5 => 43, 6 => 32, 7 => 56, "b" => 12);
```

```
?>
```

Изменение значений элементов массива

```
<?php
    $books["key"] = "PHP";
    $books[12] = 37;
    $books[] = 54;
```

```
?>
```

Использование условного оператора

```
<?php
    $books["key"] = "PHP";
```

```

$books[12] = 37;
$books[] = 54;
?>
<?php
if ($x > 2)
    {
    echo '<h1> Поздравляю!</h1>';
    echo '<p> x>2 </p>';
    }
else
    {
    echo '<h1> Сожалею!</h1>';
    echo '<p> x<=2 </p>';
    }
?>

```

```

<?php
if ($x < -1) $f = -$x;
elseif ($x < 1) $f = pow($x, 2);
elseif ($x < 2) $f = 1;
else $f = 0.5*$x;
?>

```

Оператор switch

```

<?php
$n = 3;
switch ($n)
{
    case 1: $d='Понедельник'; break;
    case 2: $d='Вторник'; break;
    case 3: $d='Среда'; break;
    case 4: $d='Четверг'; break;
    case 5: $d='Пятница'; break;
    case 6: $d='Суббота'; break;
    case 7: $d='Воскресенье'; break;
    default: $d='Не существует';
}
?>

```

Цикл while

```

<?php
$i = 1;
echo '<p>';
while ($i<=10)
    {
    echo $i,' ';
    $i++;
    }
echo '</p>';
?>

```

Цикл do while

```
<?php
$i = 1;
echo '<p>';
do
    {
        echo $i, ' ';
        $i++;
    }
while ($i<=10)
echo '</p>';
?>
```

Цикл for

```
<?php
for ($i=1; $i<=10; $i++)
    { echo $i, ' '; }
?>
```

```
<?php for ($i=1; $i<=10; print $i.' ', $i++); ?>
```

Цикл foreach

```
<?php
    $arr = array
        ("Россия"=>"Москва",
        "Франция"=>"Париж",
        "Великобритания"=>"Лондон");
    foreach ($arr as $val)
        {
            echo "$val <br>";
        }
    foreach ($arr as $k => $val)
        {
            echo "Город $val – столица государства $k.<br>";
        }
?>
```

Примеры работы с двумерным массивом

```
<?php
for($i=0;$i<5;$i++)
    for($j=0;$j<5;$j++)
        $arr[$i][$j]=$i*5+$j+1;
print_r($arr);
for($i=0;$i<5;$i++)
    {
        echo "<tr>";
        for($j=0;$j<5;$j++)
            {
                $a=$arr[$i][$j];
                echo "<td> $a </td>";
            }
    }
```

```

    }
    echo "</tr>";
}
?>
<?php
foreach($sarr as $i=>$ar)
{
    echo "<tr>";
    foreach($ar as $j=>$val)
    {
        echo "<td> $val </td>";
    }
    echo "</tr>";
}
?>

```

Пример работы со структурированными данными

```

<?php
$menu=array(
    'Апельсиновый десерт'=>array(
        'апельсин'=>'4 шт.',
        'сахар'=>'2 ст. л.',
        'мука'=>'2 ст. л.',
        'сливочное масло'=>'40 гр.',
        'сливки'=>'200 гр.',
        'консервированная клубника'=>'100 гр.'),
    'Печеные яблоки'=>array(
        'яблоки'=>'10 шт.',
        'грецкие орехи'=>'50 гр.',
        'изюм'=>'30 гр.',
        'мед'=>'10 ст. л.));
echo '<ol> Десерты:.';
foreach($menu as $i=>$ar)
{
    echo "<li> $i <ul>";
    foreach($ar as $j=>$val)
    {
        echo "<li> $j - $val </li>";
    }
    echo "</ul> </li>";
}
echo '</ol>.';
?>

```

Функции, определяемые пользователем

```

<?php
function polusumma($a, $b)
{

```

```

        $s=($a+$b)/2;
        return $s;
    }
    $alpha = 4;
    $beta = 5;
    $half = polusumma($alpha, $beta);
?>

```

Работа с файлами

```

<?php
$my_file = fopen("My_big_file.txt", "r");
while (!feof($my_file))
{
    print fread($my_file, 1024);
}
fclose($my_file);
?>

```

```

<?php
if (file_exists("D:\My_dir\file.bmp"))
{
    echo " file.bmp существует!\n";
} else {
    echo " file.bmp не существует!\n";
}
?>

```

```

<?php
if (is_writable("D:\My_dir\file.txt"))
{
    echo " file.txt существует и
                                доступен для записи!\n";
} else {
    echo " file.txt не существует или
                                не доступен для записи!\n";
}
?>

```

```

<?php
if (is_readable("D:\My_dir\file.txt"))
{
    echo " file.txt существует и
                                доступен для чтения!\n";
} else {
    echo " file.txt не существует или
                                не доступен для чтения!\n";
}
?>

```

```

<?php
$handle = opendir("D:\MyDir");

```

```

if ($handle !== False)
    {
    while(($fn=readdir($handle)) !== False)
        {
        echo "$fn </br>";
        }
    closedir($handle);
    }
?>

```

Передача файла на сервер

```

<html>
  <head>
    <title> Форма авторизации</title>
  </head>
  <body>
    <form enctype="multipart/form-data"
    action="get_my_file.php" method="post">
    <input type="hidden" name="MAX_FILE_SIZE" value="30000">
    Загрузить файл: <input type="file" name="myfile"><br>
    <input type="submit" value="Отправить файл">
    </form>
  </body>
</html>
<?php
<html>
  <head>
    <title> Форма авторизации</title>
  </head>
  <body> <p>
    <?php
    $destination = 'D:/NewDir/' . $_FILES['myfile']['name'];
    if (move_uploaded_file($_FILES['myfile']['tmp_name'],
                          $destination))
        {echo "Файл успешно загружен "};
    else
        {echo "Произошла ошибка",$_FILES['myfile']['error'];};
    ?>
  </p> </body>
</html>
?>

```

Реляционные базы данных

```

<HTML>
<Header>
</Header>
<Body>
<?php

```

```

$connection =
    mysql_connect( "localhost", "User", "PSW");
mysql_query("SET NAMES 'utf8'");
mysql_select_db ("Sklad");
$result_index = mysql_query
    ("Select Tip.Nam as Tip_tov,
     Tovar.Nam as Name_tov
     From Tip inner join Tovar
     on Tip.Kod=Tovar.Tip");
$count = mysql_num_rows($result_index);
echo "$count - Записей";
echo "<Table border=1>";
echo "<th>Тип товара</th>";
echo "<th>Название</th>";
for ($i=1;$i<=$count;$i++)
{
    echo "<tr>";
    $row = mysql_fetch_assoc($result_index);
    foreach($row as $pole=>$znach)
    {echo "<td>$znach</td>";}
    echo "</tr>";
}
echo "</Table>";
mysql_free_result($result_index);
mysql_close ($connection);
?>
</Body>
</HTML>

```