

Лабораторная работа № 1

Переменные и организация вывода на экран

Тема работы: Изучение приемов вывода на экран и использования переменных в PHP.

Цель работы: Изучить основы языка PHP.

Продолжительность работы: 2 часа.

Ход работы:

1. С помощью текстового редактора "Блокнот" (Notepad) создайте файл *script_php1.phtml* и поместите в начало страницы следующий код:

```
<html>
<head>
<title>Вывод на экран и переменные в PHP</title>
<?php
echo "Привет, мир!";
?>
</head>
<body>
</body>
</html>
```

2. Скопируйте созданный вами файл в свою папку на сервере.
3. Теперь вы можете запустить свою страницу, введя в адресную строку браузера адрес страницы.
4. Этот скрипт не делает ничего кроме вывода на экран фразы "Привет, мир!". Для того, чтобы придать скрипту полезные функции, познакомимся с переменными. Переменная характеризуется именем, типом и значением. Имя может быть любым и включать в себя цифры, буквы английского алфавита и разрешенные символы (например, символ подчеркивания или тире). По типу переменные делятся на целые, с плавающей запятой и символьные. Введите в скрипт переменные каждого из перечисленных типов:

```
<html>
```

```

<head>
<title>Вывод на экран и переменные в PHP</title>
</head>
<body>
<?php
$i = 6; // целое
$d = 4.89; // дробное
$str = "PHP для всех!"; // строка
echo ($i + $d);
echo "<br>Привет, мир! ".$str;
?>
</body>
</html>

```

5. Обратите внимание, что все переменные в PHP должны начинаться с символа dollar (\$), тип переменной не задается в явном виде, он вычисляется препроцессором PHP в зависимости от контекста. В результате работы скрипта на экран первой строкой будет выведена сумма значений двух переменных *\$i* и *\$d* и результат конкатенации двух строк "*
Привет, мир!*" и значения переменной *\$str*.
6. Поддерживаются все арифметические операции и функции, многоуровневые скобки, логические операции, операции увеличения или уменьшения на единицу и многое другое. Кроме того, очень просто и естественно организуется сравнение *если — то — иначе*. Для этого в PHP применяется конструкция:

```

if (условие)
{
    код для выполнения N1;
}
else
{
    код для выполнения N2;
}

```

Рассмотрим пример:

```

<html>
<head>

```

```

<title>Вывод на экран и переменные в PHP</title>
</head>
<body>
<?php
$i = 6; // целое
$v = 7;
$d = 4.89; // дробное
$str = "PHP для всех!"; // строка
echo ($i + $d);
echo "<br>Привет, мир!".$str;

if($i == $v)
{
    echo $i + $v;
}
else
{
    echo $i.$v;
}

?>
</body>
</html>

```

7. При сравнении на истину применяется два знака равенства для того, чтобы интерпретатор мог без труда отличить сравнение от присваивания. Результат работы скрипта — *67*, т.к. *\$i* не равно *\$v*, а команда *echo \$i.\$v*; объединяет две строки: "6" и "7". Неравенство (ложь) обозначается символами *!=*, допустимы все остальные арифметические и логические символы и операторы (например, *or*, *and*, *>*, *<=* и т.д.).
8. В PHP есть средства быстрого изменения переменной на единицу в сторону увеличения или уменьшения. Для этого нужно указать имя переменной и за ней, без знака равенства, — подряд два плюса или минуса соответственно. Например, *\$a++*; — переменная *\$a* будет увеличена на единицу. Поддерживается одновременное присваивание одного значения нескольким переменным — *\$a = \$b = 4*;. Введите на странице *script_php1.phtml* следующий код:

```

<html>
<head>

```

```
<title>Вывод на экран и переменные в PHP</title>
</head>
<body>
<?php
$b = $a = 5;
echo "<br>переменная a=$a, b=$b";
$c = $a++;
echo "<br>переменная a=$a, c=$c";
$d = $c = ++$b;
echo "<br>переменная d=$d, b=$b";
$g = 10;
$h = $g += 10;
echo "<br>переменная g=$g, h=$h";
?>
</body>
</html>
```

9. Обратите внимание, что операция присваивания тоже дает свой результат, так операция $\$a = 5$ дала результат 5, именно поэтому переменная b была проинициализирована в 5. В операции $\$c = \$a++$; сначала было произведено присваивание, а затем переменная $\$a$ была увеличена на 1. В операции $\$d = ++\b ; сначала была увеличена на 1 переменная $\$b$, а уже затем присвоено получившееся значение.

10. Измените страницу таким образом, чтобы на экран выводился результат извлечения квадратного корня из числа 2.

Требования к отчету: Отчет должен содержать детально прокомментированные тексты программ, скриншоты результатов работы программ.