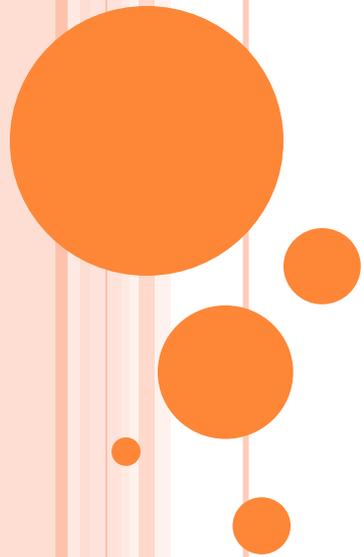


PHP

(Personal Home Page Tools)

– язык программирования



RNR

— наиболее простой скриптовый язык программирования, широко применяющийся при создании динамически генерируемых веб-страниц.

ПОЧЕМУ ИМЕННО ЯЗЫК RNR?

- Простота
- Скорость разработки
- Наличие библиотек
- Поддержка
- Безопасность



Возможности языка PHP

- написание скриптов, работающих на стороне сервера
- доступ для большинства операционных систем
- поддержка большинства современных веб-серверов
- формирование изображений, файлов PDF, роликов Flash
- генерирование любых текстовых данных
- применение совместно с использованием баз данных



СИНТАКСИС ЯЗЫКА PHP

```
<?php
```

```
...
```

```
текст программы;
```

```
...
```

```
?>
```

```
<?
```

```
...
```

```
текст программы;
```

```
...
```

```
?>
```

```
<script language =
```

```
"php">
```

```
...
```

```
текст программы;
```

```
...
```

```
</script>
```

МЕТОДЫ ДЛЯ ВСТАВКИ КОММЕНТАРИЕВ

```
// комментарий
```

```
# комментарий
```

```
/*
```

```
многострочный комментарий
```

```
*/
```



ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННЫХ В РНР

начинаются со знака \$

\$

могут содержать любое количество буквенно-цифровых символов (A÷Z, a÷z, 0 ÷ 9) и символов подчеркивания (_)

`$H1_g2...g1h_`

первый символ не может быть цифра, подчеркивание_

~~`$1g2`~~

~~`$_g2`~~

чувствительны к регистру

`$`
`h`

и

`$`
`H`

- различны



Типы данных в PHP

- тип переменной не указывается
- переменная инициализируется в момент присваивания ей значения

`$h=1 // переменная h целого типа`

`$h=1.5 // переменная h дробного типа`

`$h='Привет' // переменная h строкового типа`

`$h=true // переменная h логического типа`



ТИПЫ ДАННЫХ ПЕРЕМЕННЫХ

INTEGE

```
$a = 1234; // десятичное число
```

```
$a = -123; // отрицательное число
```

```
$a = 0123; // восьмеричное число
```

```
$a = 0x1A; // шестнадцатеричное число
```

FLOA

```
$a = 1.234;
```

```
$b = 1.2e3; // 1.2 * 10^3
```

BOOLEA

```
$flag = true; // истина
```

```
$flag = false;  
//ложь
```



ТИПЫ ДАННЫХ ПЕРЕМЕННЫХ

STRIN

С

```
$a = 'строка в одиночных кавычках';
```

```
$a = "строка в двойных кавычках";
```

```
echo <<<HEREDOC1
```

```
Привет всем!
```

```
Я хороший человек
```

```
Я печатаю
```

```
HEREDOC1;
```

ARRA

V

```
$array1 = Array('Первый','Второй')
```

```
;
```

```
$array2 = Array(1,2);
```



ИЗМЕНЕНИЕ ТИПА ДАННЫХ

intval(переменная)

floatval(переменная)

strval(переменная)

settype(переменная, параметр)

значениями параметра type:

- "boolean"
- "integer" (или "int")
- "float"
- "string"
- "array"
- "null"



ОПЕРАТОР ВЫВОДА ECHO

```
echo "Привет мир!";
```

```
$f = "foobar";  
echo $f;
```

```
echo "f - это $f";
```

```
echo 'f - это $f';
```

```
$b = array("value" => «f»);  
echo "это {$b['value']} !";
```

```
echo <<<END
```

Здесь используется синтаксис "here document" для вывода нескольких строк с подстановкой переменных \$variable. Заметьте, что закрывающий идентификатор должен располагаться в отдельной строке. никаких пробелов!

```
END;
```

```
echo "Это займет несколько  
строк. Переводы строки тоже  
выводятся";
```

```
echo "Это займет\nнесколько  
строк. Переводы строки  
тоже\nвыводятся";
```

```
echo "Экранирование  
символов делается \"  
Так \".\";
```

ОПЕРАТОР ВЫВОДА
PRINT ("...")



ОПЕРАТОР УСЛОВИЯ IF

Общий вид оператора

if (условие)

```
{...команды при верном условии...;}
```

```
else {...команды при неверном условии...;}
```

```
if ($a=>0)
```

```
{$s=$a-$a/3;}
```

```
else {$s=$a+$a*3;}
```

Общий вид функции:

```
в_1 ? в_2 : в_3;
```

```
$x<0 ? -$x : $x;
```

ОПЕРАТОР ВЫБОРА SWITCH

Общий вид:

switch (выражение)

```
{case значение: ... команды...; break;
```

```
case другое значение: ... команды...; break;}
```

```
switch ($s=$a/2)
```

```
{case 10: echo 's=10';
```

```
break;
```

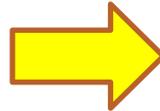
```
case 5: echo 's=5'; break;}
```



ОПЕРАТОРЫ ЦИКЛА FOR, WHILE, DO..WHILE

Общий вид

```
for (Нзсч, усл, измсч)
{ . . . команды . . . ;}
```



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

Общий вид:

```
while (условие)
{ . . .команды . . . ;}
```



```
while ($i<10)
{ $s=$s+$i;
echo'$s';
$i++;}
```

Общий вид:

```
do { . . .команды . . . ;}
while (условие);
```



```
do
{ $s=$s+$i;
echo'$s';
$i++;}
while ($i<10);
```

