

Урок №1



```
<?php
echo
    "All PHP programmers should know...
    We are privileged to have known...
    In a language that only grow...
    Each day more...
    I love PHP.";
?>
```

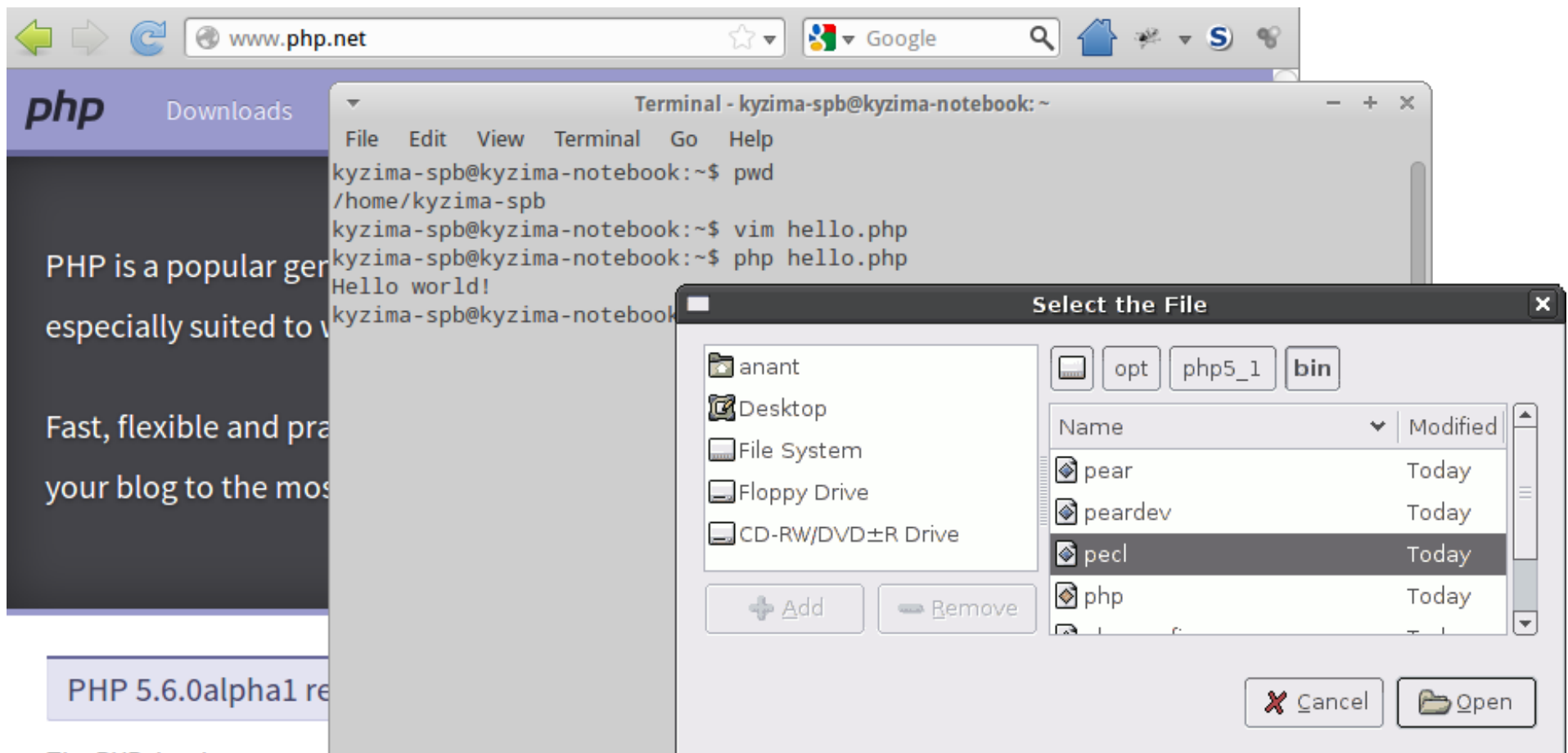
PHP: Hypertext Preprocessor

- Автор: Rasmus Lerdorf
- Семантика: мультипарадигменный
- Тип исполнения: интерпретатор компилирующего типа
- Появился: в 1994 году
- Последняя версия: 5.5.9 (06.02.2014)
- Типизация данных: динамическая
- Регистр символов: регистрозависимый
- Сайт: <http://php.net>



Какие области применения PHP?

1. Создание скриптов для выполнения на стороне сервера.
2. Создание скриптов для выполнения в консоли (командной строке).
3. Создание GUI приложений.



Какие теги использовать для ограничения сценария PHP?

```
<?php ?> # распространенная и правильная форма  
<? ?>    # сокращенная форма  
<% %>    # в стиле ASP  
<script language="php">...</script> # в стиле HTML
```

Разделитель инструкций PHP

Точка с запятой - использовать всегда!

Как добавить комментарий в код PHP?

```
// однострочный комментарий  
  
/* Это много- /* вложенные комментарии не допустимы */  
   строчный комментарий */  
  
# Комментарии в стиле UNIX-оболочек
```

Задача №1.1

Занести строку Hello World! в переменную и вывести на экран.

```
<?php  
    echo "Hello World!"; #вывод на экран  
?>
```

Как объявить переменную в PHP?

Переменная – поименованная область оперативной памяти, в которую в процессе работы программы записываются данные или из которой они извлекаются.

Переменные помечаются знаком доллара, за которым следует имя переменной.

На имена переменных накладываются следующие ограничения:

1. первым символом имени должна быть буква или знак подчеркивания, за которым следует любая комбинация буквенно-цифровых символов или знаков подчеркивания;
2. **имя не может содержать зарезервированные слова или специальные символы;**
3. имена чувствительны к регистру, поэтому `$a` и `$A` являются разными переменными.

Задача №1.2

Занести строку Hello World! в переменную и вывести на экран.

```
<?php
    $str    "Hello World!"; #присваиваем значение переменной
    echo $str;               #вывод на экран
?>
```

Какие типы данных существуют в PHP?

Тип переменной – является ее характеристикой и отражает:

1. количество выделяемой памяти;
2. формат представления данных;
3. диапазон допустимых значений;
4. операции, допустимые для применения к переменной.

PHP поддерживает 8 основных типов.

Скалярные (простые) типы

Содержат значения атомарного типа

Логические флаги

boolean (bool)

Значения – true (истина) и false (ложь).

Иногда значение также называют булевыми.

```
<?php
    $flag1 = true;
    $flag2 = false;
?>
```

Целые числа

integer (int)

Значения в диапазоне: -2147483648 – 2147483647.

Максимальное значение:

- в восьмеричной СС: 017777777777
- в шестнадцатеричной СС: 0x7FFFFFFF

```
<?php
```

```
$a = 1234; # десятичное число
```

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

```
$a = 0123; # восьмеричное число (83)
```

```
$a = 0495; # $a = 4!!!
```

```
$a = 0x6F; # шестнадцатеричное число (111)
```

```
?>
```

Числа с плавающей точкой

float (double)

Дробные числа или числа с плавающей точкой.

В PHP имена типов float и double являются синонимами.

По историческим причинам они означают одно и то же.

```
<?php
    $a = 1.234;
    $a = 1.2e3; # == 1200
    $a = 1E-3;  # == 0.001
?>
```

Строки

string

Строки могут быть обозначены одинарными и двойными кавычками или терминаторами «here doc».

При использовании одинарных кавычек строка воспринимается как есть. Использование двойных кавычек позволяет интерпретатору заменить указанную в строке переменную на ее значение.

```
$n = 1;  
echo "n = $n"; # напечатает "n = 1"  
echo "$n = $n"; # напечатает "$n = 1"  
echo '$n';      # напечатает "$n"  
  
echo "$0000"; # после $ недопустимое имя переменной  
               # поэтому строка будет выводиться как есть: "$0000"
```

Экранирование спецсимволов – обратный слеш.

В отдельных случаях строка может восприниматься как массив символов.

Структурированные типы

Могут одновременно содержать более одного значения:
простого или сложного

Массивы

array

Массив в PHP – это упорядоченное отображение, которое устанавливает соответствие между значением и ключом (индексом).

Массивы будут рассмотрены подробно в третьем уроке.

```
<?php
# нумерованный массив
$n = array(1, 2, 3);

# ассоциативный массив
$n = array(
    'key1' => 'value1',
    'key2' => 'value2',
);
?>
```

Объекты

object

Объекты – создаются на основе заранее определенных классов при помощи оператора **new**.

Классы и объекты будут рассмотрены подробно в восьмом уроке.

```
<?php
    $obj = new stdClass();

    $obj->property1 = 1;
    $obj->property2 = 'Hello';
?>
```

Специальные типы

Не предназначены для выполнения с ними
математических и других операций

Дескриптор ресурсов

resource

Дескриптор ресурсов хранит указатели на внешние ресурсы, с которыми можно работать в PHP.

Никаких операций, включая преобразование типа, с этими значениями выполнять нельзя, так как они используются только специальными функциями.

Создаются эти значения также специальными функциями при открытии доступа к определенному ресурсу (БД, файлы и т.д.).

#создает файловый дескриптор

```
$fp = fopen('/etc/passwd', 'r');
```

#создает дескриптор соединения с БД

```
$pdo = new PDO('mysql:host=localhost;dbname=t', 'root', 'root');
```

#создает дескриптор нового палитрового изображения

```
$img = imagecreate(200, 100);
```

null

Специальный тип для пустых¹ либо не инициализированных значений.

Переменная будет иметь значение **NULL** в трех случаях:

1. если переменная не была инициализирована — ей не было присвоено никакого значения;
2. если переменной явно присвоено значение **NULL**;
3. если значение переменной было явно уничтожено функцией **unset()**.

```
$a;  
$b = null;  
  
$img = imagecreate(200, 100);  
unset($img);
```

¹ Имеется ввиду отсутствие какого-либо значения, так как пустая строка все же имеет определенное значение.

Задача №1.3

Создать несколько переменных.

Каждой переменной присвоить значение простого типа.

Вывести переменные на экран используя функцию `var_dump`

`void var_dump (mixed expression [, mixed ...])`

Отображает структурированную информацию об одном или нескольких выражениях, включая их тип и значение.

```
<?php
    $a    123;

    #вывод структурированной информации о переменной
    var_dump($a);
?>
```


Как определить тип переменной PHP?

Название	Синтаксис
is_null	<code>bool is_null (mixed var)</code>
is_bool	<code>bool is_bool (mixed var)</code>
is_int	<code>bool is_int (mixed var)</code>
is_float	<code>bool is_float (mixed var)</code>
is_string	<code>bool is_string (mixed var)</code>
is_array	<code>bool is_array (mixed var)</code>
is_object	<code>bool is_object (mixed var)</code>
is_resource	<code>bool is_resource (mixed var)</code>

Как выполнить явное приведение переменной к определенному типу?

Сходно с C-стилем, когда тип указывается перед исходной переменной.

Тип	Описание
(int), (integer)	для целых чисел
(bool), (boolean)	для логических значений
(float), (double), (real)	для дробей
(string)	для строк
(array)	для массивов
(object)	для объектов

Задача №1.4

Используя `var_dump` и явное приведение переменной к типу `Boolean` проверить в каких случаях переменная `true`, а в каких `false`:

непосредственное указано значение `false`;

целое или дробное число имеет нулевое значение;

строка является пустой (`""`) или содержит один ноль (`"0"`);

массив не содержит ни одного элемента;

указана не существующая или не имеющая значения переменная, либо явно указан `NULL`.

Какие операторы существуют в PHP?

1. Операторы присваивания;
2. Арифметические операторы;
3. Инкремент и декремент;
4. Конкатенация – слияние строк;
5. Операторы сравнения;
6. Оператор проверки класса объекта;
7. Логические операторы;
8. Бинарные операторы;
9. Операторы контроля ошибок;
10. Операторы запуска внешней программы.

Задача №1.5

Задача на мышление.

Используя полученные знания о приоритетах операторов, решить устно следующие выражения:

$i += ++i + i++$, если $i=6$

$++i+i+++i$, если $i=1$

$b = a++ + (--a * ++a)$, чему равно b , если $a=2$

Как объявить константу в PHP?

Константа – не изменяемое значение.

Имена констант подчиняются тем же правил, что и имена переменных, НО перед именем константы не ставится знак доллара.

Могут иметь только константные значения логического, числового или строкового типа.

```
<?php
    define('HELLO_WORLD', 'Hello World!');
    echo HELLO_WORLD;
?>
```


Конец