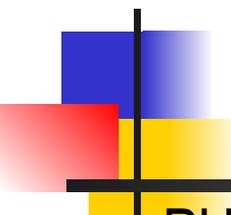
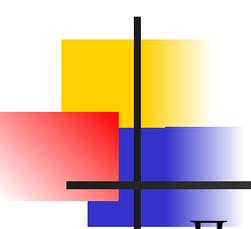


PHP. Функции. ОСНОВЫ РАБОТЫ



PHP – процедурный язык программирования. Подпрограммы в PHP представлены функциями. Умение грамотно работать с ними – основа эффективного программирования на языке

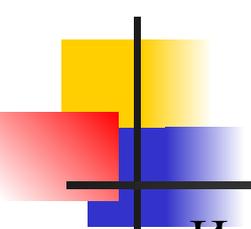


Понятие подпрограммы

- Подпрограмма – поименованная часть компьютерной программы, содержащая описание определённого набора действий
- Преимущества использования подпрограмм:
 - Сокращение объема кода программы (устаревш.)
 - Структуризация программы с целью удобства ее понимания и сопровождения
 - Обеспечение возможности повторного использования кода через создание библиотек
- Подпрограмма выполняет определенную последовательность действий (алгоритм) над данными, переданными в параметрах
- Подпрограммы во многих языках делятся на процедуры и функции
- В ряде языков (в т.ч. PHP) процедуры заменены функциями не возвращающими значения

Функции в PHP

- В языке имеются развитые синтаксические средства для работы с функциями
- Существует значительное количество определенных функций, которыми может пользоваться прикладной программист
- Часть таких функций являются встроенными функциями и языковыми конструкциями и доступны в языке всегда
- Часть функций находится в модулях и расширениях и включаются в сборку PHP по умолчанию
- В сборку могут также быть включены любые другие модули и расширения
- Функции `phpinfo()` и `get_loaded_extensions()` позволяют узнать какие модули подключены в данной сборке
- Пользователю предоставляются средства определения новых функций и включения их в программные проекты



Особенности работы с функциями в PHP

- Идентификаторы функций регистронезависимы
- В PHP функции определяются однократно, не удаляются, не переопределяются и не перегружаются!
- Все функции имеют глобальную область видимости
- Функции не обязаны быть определены до их использования
- Рекурсия допустима, но может вести к переполнению стека

Описание функций

- Описание функций с помощью прототипов
- Примеры прототипов функций
 - `int Sum (int $Par1, int $Par2)`
 - `void Func1 (bool $Left [, bool $Right = TRUE])`
 - `mixed Oper (mixed Par1, mixed Par2, [mixed Par N, ...])`
- Сведения, необходимые наряду с прототипом
 - Ссылка на версии
 - Общее описание действия функции
 - Список формальных параметров
 - Возвращаемое значение
 - Замечание
- Возвращаемые значения ¶

Псевдотипы

- **mixed** – псевдотип, указывающий на то, что параметр может принимать много типов (но необязательно все):
например, функция **gettype()** принимает все типы PHP, тогда как **str_replace()** принимает только типы `string` и `array`.
- **number** - `number` говорит о том, что параметр может быть либо **integer**, либо **float**.
- **void** - в качестве типа результата означает, что возвращенное значение бесполезно, в списке параметров означает, что функция не принимает параметров

Пример описания функции в справочнике функций на php.net

strlen Change language: [Edit](#) [Report a Bug](#)

(PHP 4, PHP 5)
strlen — Возвращает длину строки

Описание

```
int strlen ( string $string )
```

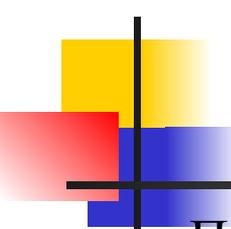
Возвращает длину строки **string**.

Список параметров

string
Строка (string), для которой измеряется длина.

Возвращаемые значения

Длина строки **string** в случае успеха, и 0, если **string** пуста.



Пройденный материал: Функции. Основы работы

- Процедурное программирование и подпрограммы
- Функции в РНР и особенности работы с ними
- Описание функций
- Использование псевдотипов для описания функций
- Примеры описания функций