

Slide 3.14: PHP built-in functions
Slide 4.1: PHP forms and user input
[Home](#)



PHP User-Defined Functions

Other than the built-in functions, users can create their own functions. A user function is a block of code that can be executed whenever we need it:

- All functions start with the word `function`.
- Give the function a name that reflects what the function does.
- Parameters are specified after the function name, inside the parentheses. In PHP, call-by-value is by default. If an `&` is added before the variable name in the function definition, it passes parameters by name instead.

In a call-by-name system, a variable within your named block of code points to the same piece of computer memory as the variable in the calling code.

- Add a `{`—The function code starts after the opening curly brace.
- Insert the function code. To let a function return a value, use the [return](#) statement.
- Add a `}`—The function is finished by a closing curly brace.

This program reads a list of integers and returns the sum of even values by

- Implementing a user-defined function `even`,

```
<html><body>
<?php
function even( $x ) {
    if ( $x%2 == 0 ) return( $x );
    else return( 0 );
}
$list   = " 10 -2 0   3 4 1 -14 ";
$sep    = " ";
$number = strtok( $list, $sep );
```

which uses the modulus operator % to decide whether a value is even, and

- Using a built-in function [strtok\(string,split\)](#), which returns a sequence of integers one by one.

```
$acc    = 0;
while ( is_numeric( $number ) ) {
    $acc  += even( $number );
    $number = strtok( $sep );
}
echo  $acc;
?>
</body></html>
```

Output =

[Slide 3.14: PHP built-in functions](#)
[Slide 4.1: PHP forms and user input](#)
[Home](#)

