



PHP Features you didn't know existed

Elliott White III - *Eli*

What is this talk?

Good Question

A random list of 'neat things' that you can do in PHP,
that perhaps you didn't know about.

At least hopefully there are a few that you haven't
heard of.

http_build_query()

```
$vals = array('search' => 'PHP 5 in Practice',  
             'site' => 'digg.com');  
$query = http_build_query($vals);  
  
$resend = http_build_query($_GET);
```

ignore_user_abort()

```
ignore_user_abort(true);
```

```
$bool = connection_aborted();
```

register_shutdown_function()

```
function shutdown() {}  
register_shutdown_function('shutdown');
```

set_time_limit()

```
set_time_limit(600);
```

Use with `register_shutdown_function` to handle timeouts.

range()

```
$array = range(10,20);
```

```
$reversed = range(20,10);
```

```
$stepped = range(10,20,3);
```

```
$alpha = range('a','f');
```

readfile() as a passthrough

```
readfile('http://example.com/data.html');  
  
file_get_contents('http://example.com/data.html');
```

strip_tags()

```
$stripped = strip_tags($input);
```

```
$semi = strip_tags($input, '<p><b>');
```

Spellchecking

```
$dict = pspell_new('en');
$words = str_word_count($str, 2);
foreach ($words as $word) {
    if (!(pspell_check($dict, $word))) {
        $suggest = pspell_suggest($dict, $word);
        echo "{$word}: ", implode($suggest, ' '), "<br />";
    }
}
```

Similar Text / Levenshtein

levenshtein() returns the number of changes needed:

```
$diff = levenshtein($name, 'Eli White');
```

similar_text() can give you a percentage of similarity:

```
$num = similar_text($name, 'Eli White', $percent);
```

Soundex / Metaphone

```
$key = metaphone($word);
```

```
$key = soundex($word);
```

Metaphone is 'supposed' to be more accurate since it understands basic rules of English pronunciation, and generates keys of variable length.

serialize() & unserialize()

```
$var = array('digg' => 1, 'life' => 42);  
$v = serialize($var);
```

```
$var = unserialize("a:2:{s:4:\"digg\";i:1;s:4:\"life\";i:42;}");
```

Header buffering & headers_sent()

Ini File:

```
output_buffering = 4096
```

```
if (!(headers_sent())) {  
    setcookie('what', 'It Works');  
}
```

@ symbol

```
$result = @function();
```

```
@$val += 1;
```

```
$whatever = @$_GET['v'] ? $_GET['v'] : 'default';
```

@ vs. *isset()* vs. *array_key_exists()*

`@$var['a']['b']`

`isset($var['a']['b'])`

`array_key_exists('b', $var['a'])`

trigger_error()

```
trigger_error("Whoops", E_USER_ERROR);
```

Direct access of globals - \$GLOBALS

`$GLOBALS['myvar']`

fscanf() using character sets

```
fscanf($file, "%3s*%40[^\n] %f\n", $a, $b, $c);
```

```
fscanf($file, "%3[a-c]\t%6[0-9]\n", $a3, $a6);
```

```
fscanf($file, ' "[^"]" "[^"]" ');
```

do { while (false);

```
do {  
    if (something()) {  
        break;  
    }  
    somethingelse();  
} while (false);
```

php://input & php://stdin

Command line input:

```
file_get_contents('php://stdin');
```

Raw Post data:

```
file_get_contents('php://input');
```

php://filter

```
file_put_contents('php://filter/write=string.toupper|  
string.rot13/resource=myfile.rot');
```

```
readfile('php://filter/read=string.strip_tags  
/resource=http://digg.com');
```

```
var_dump(stream_get_filters());
```

compress.zlib://

```
file_put_contents('compress.zlib://gzipfile.gz', 'DATA');  
readfile('compress.zlib://gzipfile.gz');
```

Generating lists of months/weekdays

```
foreach (range(1,12) as $month) {  
    $daystamp = mktime(0, 0, 0, $month);  
    $monthnames[] = date('F', $daystamp);  
}
```

```
foreach(range(1,7) as $day) {  
    $stamp = mktime(0, 0, 0, 1, $day);  
    $days[date('N', $stamp)] = date('l', $stamp);  
}
```

Including code from other servers

```
include 'http://example.com/file.inc';
```

Possible Other Topics

- Output Buffering
- SPL Iterators
- Assertions
- Exif

Digg is Hiring!

Digg.com is hiring experienced PHP programmers!

<http://digg.com/jobs>
jobs@digg.com

Any Questions?

For this presentation and more:

<http://eliw.com/>

Visit <http://digg.com/>

The Digg logo consists of the word "digg" in a white, lowercase, sans-serif font, centered on a solid blue rectangular background.