

```
<?php
/**
 * Analyses speed gains for different "for" statements
 *
 * Once one can understand which statements and functions are
 * executed faster, one can optimize one's projects.
 *
 * @name      "For" Statement Analyzer
 * @author    David Branco <David@NeoeliteUSA.com>
 * @link      http://www.NeoeliteUSA.com/
 * @version   1.05
 * @since     Oct 21, 2006
 * @license   http://creativecommons.org/licenses/by/2.5/
 */

/* Timer Class to measure the speed of PHP */
class Timer {
    private static $page_generate_start;
    private static $page_generate_end;
    public static $page_generate_time;

    function measure()
    {
        list($usec, $sec) = explode(" ", microtime());
        return ((float)$usec + (float)$sec);
    }

    function start()
    {
        self::$page_generate_start = self::measure();
        return true;
    }

    function end()
    {
        self::$page_generate_end = self::measure();
        self::$page_generate_time = self::$page_generate_end - self::$page_generate_start;
        return true;
    }
}

$timer = new Timer();
timer::start();

for ($i = 0; $i < 10000000; $i++) {}

timer::end();
echo 'for ($i = 0; $i < 10000000; $i++)'. '<br />'. timer::$page_generate_time. '<br /><br />'. "\n";

timer::start();

for ($i = 0; $i < 10000000; ++$i) {}

timer::end();
echo 'for ($i = 0; $i < 10000000; ++$i)'. '<br />'. timer::$page_generate_time. '<br /><br />'. "\n";

timer::start();

for ($i = 10000000-1; $i >= 0; --$i) {}

timer::end();
echo 'for ($i = 10000000-1; $i >= 0; --$i)'. '<br />'. timer::$page_generate_time. '<br /><br />'. "\n";

timer::start();

for ($i = 10000000-1; $i >= 0; $i--) {}
```

```

timer::end();
echo 'for ($i = 10000000-1; $i >= 0; $i--)'.<br />.timer::$page_generate_time.<br /><br />.'.<n";

timer::start();

for ($i = 10000000; $i > 0; --$i) {}

timer::end();
echo 'for ($i = 10000000; $i > 0; --$i)'.<br />.timer::$page_generate_time.<br /><br />.'.<n";

timer::start();

for ($i = 10000000; $i > 0; $i--) {}

timer::end();
echo 'for ($i = 10000000; $i > 0; $i--)'.<br />.timer::$page_generate_time.<br /><br />.'.<n";

/*****/
/*****/
/*****/
/*****/
/*****/
/*****/
function printr ( $object , $name = '' ) {
    if ( is_array ( $object ) ) {
        print ( '<pre>' ) ;
        print_r ( $object ) ;
        print ( '</pre>' ) ;
    } else {
        var_dump ( $object ) ;
    }
}
$array = array('a','b','c','d','e','f','g','u','k','n','o','p');

echo printr($array).<n";

timer::start();

$total=count($array);
for ($i=0; $i<$total; $i++) {}

timer::end();
echo 'Traditional for statment: for ($i=0; $i<$total; $i++)'.<br />.timer::$page_generate_time.<br /><br />.'.<n";

timer::start();

for ($i=0, $total=count($array); $i<$total; $i++) {}

timer::end();
echo 'Wei Style for statment: for ($i=0, $total=count($array); $i<$total; $i++)'.<br />.timer::$page_generate_time.<br /><br />.'.<n";

timer::start();

for ($i=count($array); $i > 0; --$i) {}

timer::end();
echo 'Fastest for statment: for ($i=count($array); $i > 0; --$i)'.<br />.timer::$page_generate_time.<br /><br />.'.<n";

timer::start();

foreach ($array as $arr) {}

```

```
timer::end();  
echo 'foreach statment: foreach ($array as $arr)'. '<br />'. timer::$page_generate_time. '<br /><br />'. "\n";  
  
?>
```