

# APLIKACJE INTERNETOWE PHP



**TEMAT 5-08:** Generowanie grafiki w praktyce.

Autor dokumentu: Wojciech Galiński

środa, 7 września 2016 r.

351203 Technik informatyk

ŹRÓDŁA WIEDZY: <http://www.php.net/manual/pl>, <http://webmaster.helion.pl/index.php/kurs-php>, <http://pl.wikipedia.org/>,  
<http://pl.wikibooks.org/wiki/PHP>, <http://phpkurs.pl/>, <http://kursphp.com/>.

## Zagadnienia obowiązkowe

1. **Zastosowanie filtra oraz innych efektów do obrazu** – służy do tego funkcja „**imagefilter**”.

### PRZYKŁAD:

```
// ***** Wczytanie obrazu w wybranym formacie *****
$obraz = imagecreatefromjpeg('obraz.jpg');
// $obraz = imagecreatefrompng('obraz.png');
// $obraz = imagecreatefromgif('obraz.gif');
// ***** Wybór jednego lub więcej filtrów *****
// imagefilter($obraz, IMG_FILTER_NEGATE);
// imagefilter($obraz, IMG_FILTER_GRAYSCALE);
// imagefilter($obraz, IMG_FILTER_CONTRAST, 50);
// imagefilter($obraz, IMG_FILTER_BRIGHTNESS, 50);
// imagefilter($obraz, IMG_FILTER_COLORIZE, 0, 255, 0, 48);
// imagefilter($obraz, IMG_FILTER_EDGEDETECT);
imagefilter($obraz, IMG_FILTER_EMOSS);
// ***** Inne efekty zastosowane do obrazu *****
// imagegammaadjust($obraz, 1.0, 3.0);
// ***** Wyświetlenie obrazu w formacie JPEG *****
header('Content-type: image/jpeg'); imagejpeg($obraz); imagedestroy($obraz);
```

2. **Generowanie obróconego obrazka z przeźroczystym tłem** – po obróceniu na obrazku pojawiają się jednokolorowe obszary, które przeszkadzają. Najlepszym rozwiązaniem jest ustawienie tych obszarów jako przeźroczystych.

### PRZYKŁAD:

```
<?php
$nazwa_pliku = 'obraz.jpg'; $kat = -5;
// Obrót obrazka z zachowaniem przeźroczystości
$zrodlo = imagecreatefromjpeg($nazwa_pliku);
$obraz = imagecreatetruecolor(imagesx($zrodlo), imagesy($zrodlo));
imagecopy($obraz, $zrodlo, 0, 0, 0, 0, imagesx($zrodlo), imagesy($zrodlo));
$skolor_alfa = imagecolorallocatealpha($obraz, 0, 0, 0, 127); // 127 - przeźroczystość
$obraz = imagerotate($obraz, -5, $skolor_alfa); imagesavealpha($obraz, true);
imagepng($obraz, 'obrocony.png', 9);
?>
```

3. **Tworzenie miniatury „w locie”** – może w tym pomóc funkcja „**imagecopyresampled**” (funkcja „**imagecopyresized**” robi to samo, ale bez interpolacji).

### PRZYKŁAD:

```
<?php
$nazwa_pliku = 'obrocony.png';
// Utworzenie w pamięci oryginalnego obrazka oraz obrazka dla miniatury
$obraz = imagecreatefrompng($nazwa_pliku);
// Rozmiary miniatury: ustalona szerokość i proporcjonalna wysokość albo na odwrót
list($szerokosc, $wysokosc) = getimagesize($nazwa_pliku);
$nowa_szerokosc = 256; $nowa_wysokosc = $nowa_szerokosc*$wysokosc/$szerokosc;
// $nowa_wysokosc = 256; $nowa_szerokosc = $nowa_wysokosc*$szerokosc/$wysokosc;
// Wygenerowanie miniatury
$miniatura = imagecreatetruecolor($nowa_szerokosc, $nowa_wysokosc);
imagealphablending($miniatura, false);
imagecopyresampled($miniatura, $obraz, 0, 0, 0, 0,
                    $nowa_szerokosc, $nowa_wysokosc, $szerokosc, $wysokosc);
imagecolorallocatealpha($miniatura, 0, 0, 0, 127); // 127 - przeźroczystość
imagesavealpha($miniatura, true);
imagepng($miniatura, 'kafelek.png', 9); // wyświetlenie pomniejszonego obrazka
?> 
```

4. **Wypełnianie wybranego obszaru obrazka deseniem (kafelkami)** – można do tego wykorzystać funkcje: „**imagecreatefrompng**” oraz „**imagefilledrectangle**” z opcją „**IMG\_COLOR\_TILED**”.

PRZYKŁAD:

```
<?php
    $szerokosc = 1000;    $wysokosc = 500;    $nazwa_pliku = 'kafelek.png';
    $kafelek = imagecreatefrompng($nazwa_pliku);
    $obraz = imagecreatetruecolor($szerokosc, $wysokosc);
    imagealphablending($obraz, false);
    imagecopy($obraz, $kafelek);
    imagefilledrectangle($obraz, 0, 0, $szerokosc-1, $wysokosc-1, IMG_COLOR_TILED);
    imagecolorallocatealpha($obraz, 0, 0, 0, 127); // 127 - przezroczystość
    imagesavealpha($obraz, true);
    imagepng($obraz, 'kafelki.png', 9);
?>
```

5. **Kopiowanie obrazu** – służy do tego funkcja „**imagecopy**”.

PRZYKŁAD:

```
// Wczytanie obrazu i nakładki na obraz
$obraz = imagecreatefromjpeg('obraz.jpg');
$kafelek = imagecreatefrompng('kafelek.png');
$margines_prawa = 10;    $margines_gora = 10; // margines dla nakładki
// Wstawienie nakładki na obrazek
$sz = imagesx($kafelek);    $ws = imagesy($kafelek);
$x = imagesx($obraz) - $sz - $margines_prawa;    $y = $margines_gora;
imagecopy($obraz, $kafelek, $x, $y, 0, 0, $sz, $ws);
// Wyświetlenie obrazka
header('Content-type: image/jpeg');    imagejpeg($obraz);    imagedestroy($obraz);
```

6. **Tworzenie wykresu „w locie”** – wykorzystamy poznane wcześniej funkcje do wygenerowania wykresu słupkowego oraz kołowego.

PRZYKŁAD:

```
<?php $szerokosc = 700;    $wysokosc = 300;    $ramka = 10;    $grubosc = 18;    $odstep = 8;
$dane = array(0.5, 1, 2, 5, 10, 20, 50, 100, 50, 20, 10, 5, 2, 1, 0.5); // % wysokości
// Wykres słupkowy
$wykres = imagecreatetruecolor($szerokosc, $wysokosc);
imagealphablending($wykres, false);    imagesavealpha($wykres, true);
$przezroczysty = imagecolorallocatealpha($wykres, 0, 0, 0, 127); // 127 - przezroczystość
imagefilledrectangle($wykres, 0, 0, $szerokosc, $wysokosc, $przezroczysty);
for ($i=0; $i<count($dane); ++$i)
    imagefilledrectangle($wykres, $ramka+$i*($grubosc+$odstep),
        ($wysokosc-$ramka)-($wysokosc-2*$ramka)*$dane[$i]/100,
        $ramka+$grubosc+$i*($grubosc+$odstep), $wysokosc-$ramka, 0xFFFFFF);
// Wykres kołowy
$sum = array_sum($dane);    $sum_biez = 0;    $s = $wysokosc;
$sx = $szerokosc - $s/2;    $sy = $wysokosc/2;    $s -= 2*$ramka;
imagesetthickness($wykres, 3);
$kolory = array(0x1abc9c, 0x2ecc71, 0x3498db, 0x9b59b6, 0x34495e, 0xf1c40f, 0xe67e22,
    0xe74c3c, 0xecf0f1, 0x95a5a6, 0x16a085, 0x27ae60, 0x2980b9, 0x8e44ad,
    0x2c3e50, 0xf39c12, 0xd35400, 0xc0392b, 0xbdc3c7, 0x7f8c8d);
for ($i=0; $i<count($dane); ++$i)
{
    imagefilledarc($wykres, $sx, $sy, $s, $s, $sum_biez, $sum_biez+360*$dane[$i]/$sum,
        $kolory[$i % count($kolory)], IMG_ARC_EDGED);
    $sum_biez += 360*$dane[$i]/$sum;
}
imagefilledarc($wykres, $sx, $sy, $s/3, $s/3, 0, 360, 0x000000, IMG_ARC_EDGED);
imagesetthickness($wykres, 2);
imagearc($wykres, $sx, $sy, $s/3+2, $s/3+2, 0, 360, 0xaaaaaa); // obramowanie wew.
imagearc($wykres, $sx, $sy, $s+2, $s+2, 0, 360, 0xaaaaaa); // obramowanie zew.
imagepng($wykres, 'wykres.png');    imagedestroy($wykres);
?><p style="background: yellow; padding: 20px;"></p>
```

## Zadania

1. Zapoznaj się z przedstawionymi przykładami. Wykonaj je i przeanalizuj ich działanie.
2. Wypróbuj inne (zakomentowane) filtry w przykładzie z pierwszego punktu.
3. Wygeneruj miniatury plików z wybranego katalogu i zapisz je dodając do nazwy „\_small” (nie zmieniając rozszerzenia pliku).
4. Do wykresu z ostatniego zadania wstaw wartości na obu wykresach.