

Kurs WWW

Paweł Rajba

pawel@ii.uni.wroc.pl
<http://pawel.ii.uni.wroc.pl/>

Spis treści

- Obsługa obrazków w PHP
 - Wprowadzenie
 - Tworzenie i niszczenie obrazka
 - Kolory
 - Funkcje od obsługi obrazków

Wprowadzenie

- Przygotowanie php do obsługi obrazków:
 - w pliku php.ini usuwamy komentarz w wierszu
`;extension=php_gd2.dll`
 - restart Apache'a
 - W WAMP'ie dodajemy rozszerzenie **php_gd2**
- Na marginesie
 - żeby pojawiały się komunikaty o błędach wprowadzamy wiersz
`display_errors = On`
zamiast
`display_errors = Off`

Tworzenie i niszczenie obrazka

- Do utworzenia obrazka mamy funkcje
 - resource imagecreate(int x_size,int y_size)
 - tworzy zupełnie pusty obrazek (bez tła)
 - resource imagecreatetruecolor(int x_size,int y_size)
 - tworzy czarny obrazek o podanych rozmiarach
 - resource imagecreatefromjpeg(string filename)
 - resource imagecreatefrompng(string filename)
 - tworzą obrazek na podstawie pliku
- Do usunięcia obrazka z pamięci mamy funkcję
 - bool imagedestroy (resource image)

Kolory

- Do tworzenia koloru używamy funkcji
 - `int imagecolorallocate (resource image, int red, int green, int blue)`
 - znaczenie red, green, blue
 - image określa, dla którego obrazka tworzymy kolor
 - ma znaczenie, gdy tworzymy obrazek poprzez `imagecreate`
 - pierwszy utworzony kolor będzie kolorem tła
 - dostępne będziemy mieli tylko utworzone dla obrazka kolory
 - powyższe nie dotyczy funkcji `createimatetruecolor`
 - od razu mamy dostępne wszystkie kolory

Kolory

- Do usunięcia koloru z obrazka mamy funkcję
 - `int imagecolordeallocate (resource image, int color)`

Funkcje do obsługi obrazków

- `int imagerectangle (resource image, int x1, int y1, int x2, int y2, int c)`
 - rysuje prostokąt
 - `x1, y1` – lewy górny róg; `x2, y2` – prawy dolny róg; `c` – kolor
- `int imagefilledrectangle (resource image, int x1, int y1, int x2, int y2, int c)`
 - rysuje wypełniony prostokąt
 - `x1, y1` – lewy górny róg, `x2, y2` – prawy dolny róg; `c` – kolor

Funkcje do obsługi obrazków

- `int imageellipse (resource image, int cx, int cy, int w, int h, int color)`
 - rysuje elipsę
 - `cx, cy` – środek, `w` – szerokość, `h` – wysokość; `color` – kolor
- `int imagefilledellipse (resource image, int cx, int cy, int w, int h, int color)`
 - rysuje wypełnioną elipsę
 - znaczenie parametrów jak powyżej

Funkcje do obsługi obrazków

- `int imagearc (resource image, int cx, int cy, int w, int h, int s, int e, int color)`
 - rysuje wycinek koła
 - `cx, cy` – środek, `w, h` – szerokość, wysokość; `s, e` – kąty startowy, końcowy, `color` – kolor
- `bool imagefilledarc (resource image, int cx, int cy, int w, int h, int s, int e, int color, int style)`
 - rysuje wypełniony wycinek koła
 - `style` przyjmuje wartości: `IMG_ARC_PIE`, `IMG_ARC_CHORD`, `IMG_ARC_NOFILL`, `IMG_ARC_EDGED`; pozostałe parametry jak powyżej

Funkcje do obsługi obrazków

- `bool imagefilter (resource src_im, int filtertype
[, int arg1 [, int arg2 [, int arg3]]])`
 - nałożenie filtra na obrazek
 - dostępne filtry:
 - `IMG_FILTER_NEGATE`, `IMG_FILTER_GRAYSCALE`
 - `IMG_FILTER_BRIGHTNESS`, `IMG_FILTER_CONTRAST`
 - `IMG_FILTER_COLORIZE`, `IMG_FILTER_EDGEDETECT`
 - `IMG_FILTER_EMOSS`, `IMG_FILTER_GAUSSIAN_BLUR`
 - `IMG_FILTER_SELECTIVE_BLUR`
 - `IMG_FILTER_MEAN_REMOVAL`
 - `IMG_FILTER_SMOOTH`

Funkcje do obsługi obrazków

- `int imageline (resource image, int x1, int y1, int x2, int y2, int color)`
 - rysuje linie
 - `x1, y1` – jeden koniec kreski; `x2, y2` – drugi koniec kreski; `color` – kolor lub styl
- `bool imagesetstyle (resource image, array style)`
 - drugi argument jest tablicą pikseli
 - żeby zadziałało trzeba funkcję `imageline()` wywołać z ostatnim parametrem `IMG_COLOR_STYLED`
- `int imagesetpixel (resource img, int x, int y, int color)`
 - rysuje pojedynczy pixel o podanym kolorze

Funkcje do obsługi obrazków

- array `imageTTFtext` (resource image, float size, float angle, int x, int y, int color, string fontfile, string text)
 - size – rozmiar czcionki, angle – kąt obrotu, x,y – pozycja, color – kolorek, fontfile – plik czcionki, text – tekst napisu
- int `imageString` (resource image, int font, int x, int y, string s, int color)
 - font – liczba 1-5 określa jeden z typów wbudowanych; x, y – pozycja; s – napis; color – kolorek
- array `getImageSize` (string filename
[, array &imageinfo])
 - pobiera rozmiar obrazka

Obrazki

- Przykłady
 - zestaw.php
 - wykres-kolowy.php
 - znak-wodny.php
 - skalowanie.php
 - licznik.php