

Ćw.1. Skrypt do przesyłania obrazów jpg i tworzenia miniatur zdjęć

Na początku potrzebny jest formularz html (np. *zdjecia.html*), który umożliwi użytkownikowi przesyłanie zdjęć na serwer. Przy tworzeniu formularza przechwytyującego pliki należy ustawić parametr *enctype="multipart/form-data"*. Ważne jest również, by pamiętać o ustawieniu parametru *name* dla pola typu *file*. Przykładowy kod formularza znajduje się na Listingu 1.

Listing 1.

```
<form method="post" action="zdjecia.php" enctype="multipart/form-data" >
  Max. wys. <input type="text" name="wys"><br/>
  Max. szer. <input type="text" name="szer"><br/>
  Plik .jpg <input type="file" name="zdjecie"><br/>
  <input type="submit" value="Zapisz" name="zapisz" >
</form>
```

W celu obsługi przechwycenia zdjęcia należy utworzyć oddzielny plik php np. *zdjecia.php*. Pominiemy tu walidację danych w polach **Max. wys.** i **Max. Szer** (dotyczy wymiarów **miniatur w pikselach**), ale w ostatecznej wersji skryptu należy to uzupełnić.

Następnie:

1. Sprawdzamy, czy formularz został wysłany. Używamy do tego funkcji **isset**. Jako argument podajemy atrybut *name* przycisku do wysłania formularza - w naszym przypadku jest to *'zapisz'*. Sprawdzamy również, czy nie przesłano parametru o nazwie *pic* metodą **get**. Będzie to nam potrzebne później, przy generowaniu linków do naszych zdjęć. Następnie korzystamy z funkcji *is_uploaded_file*, by sprawdzić, czy przesyłanie pliku na serwer zakończyło się powodzeniem. Jeśli tak, potwierdzamy, czy typ pliku jest zgodny z naszymi założeniami (tj. jpeg). Jeśli wszystko jest w porządku, zapisujemy wgrane zdjęcie. Działania te realizuje kod:

```
if (isset($_POST['zapisz']) && $_POST['zapisz'] == 'Zapisz' && !isset($_GET['pic'])) {
    if (is_uploaded_file($_FILES['zdjecie']['tmp_name'])) {
        $typ = $_FILES['zdjecie']['type'];
        if ($typ == 'image/jpeg') {
            move_uploaded_file($_FILES['zdjecie']['tmp_name'], './' . $_FILES['zdjecie']['name']);
        }
    }
}
```

2. Po wykonaniu tych czynności tworzymy kopię zdjęcia, żeby wygodniej było pracować nad miniaturą. Następnie ustawiamy nagłówek na **Content-Type: image/jpeg**, by zasygnalizować przeglądarce, że na wyjściu będzie obraz. Pobieramy informacje o wymiarach przesłanego zdjęcia i porównujemy je z maksymalnymi wymiarami zalecanymi przez użytkownika, by ustawić poprawną skalę i wymiary miniatury:

```
$link = $_FILES['zdjecie']['name'];
$random = uniqid(rand);
$zdj = $random . '.jpg';
copy($link, './' . $zdj);

header('Content-Type: image/jpeg');
list($width, $height) = getimagesize($zdj);

$wys = $_POST['wys'];
$szer = $_POST['szer'];
$skalaWys = 1;
$skalaSzer = 1;
$skala = 1;
if ($width > $szer)
    $skalaSzer = $szer/$width;
```

```

if ($height > $wys)
$skalaWys = $wys/$height;
if ($skalaWys <= $skalaSzer)      $skala = $skalaWys;
else      $skala = $skalaSzer;

$newH = $height*$skala;
$newW = $width*$skala;

```

3. Tworzymy plik graficzny wypełniony kolorem czarnym o wymiarach miniatury, a także pobieramy zawartość oryginalnego obrazu przy użyciu funkcji *imagecreatefromjpeg*. Następnie kopiujemy pobraną zawartość w całości do przeskalowanego pliku i zapisujemy go na serwerze korzystając z funkcji *imagejpeg*. Po zakończeniu tych czynności usuwamy wcześniej stworzony czarny obraz oraz zawartość oryginalnego zdjęcia, a także kopię oryginalnego obrazu, którą stworzyliśmy na początku:

```

$nowe = imagecreatetruecolor($newW, $newH);
$obraz = imagecreatefromjpeg($zdj);

imagecopyresampled($nowe, $obraz, 0, 0, 0, 0, $newW, $newH, $width, $height);
imagejpeg($nowe, './mini-'. $link, 100);

imagedestroy($nowe);
imagedestroy($obraz);
unlink($zdj);

```

4. Miniatura przesłanego przez użytkownika zdjęcia została utworzona, więc pozostaje już tylko udostępnić linki do niej i do oryginału. W tym celu wykorzystamy wcześniej wspomniany parametr zapisany pod nazwą *pic* i przesyłany metodą *get*. Ponieważ wcześniej zadeklarowaliśmy w nagłówku, że wyjście jest obrazem, więc przeglądarka nie wyświetli żadnego tekstu tylko obraz o nazwie przekazanej za pomocą *pic*. Po zakończeniu wykonywania skryptu tworzenia miniatury, przeniesiemy użytkownika ponownie do tego samego pliku (**zdjecia.php**), ale z dodatkową informacją. Strona, do której zostanie przeniesiony użytkownik będzie miała link: **zdjecia.php?pic=nazwa_zdjecia**. W naszym pliku php dopisujemy linię sprawdzającą, czy metodą *get* przesłano parametr o nazwie *pic* i czy nie jest on pusty. Jeśli wszystko jest w porządku, na wyjście wyprowadzane są linki do zdjęcia, które przesłał użytkownik i do jego miniatury, oraz link do powrotu na stronę formularza. Fragment ... else header('location: index.html'); odnosi się do wcześniejszego warunku *if*, który sprawdzał, czy typ przesłanego pliku był poprawny:

```

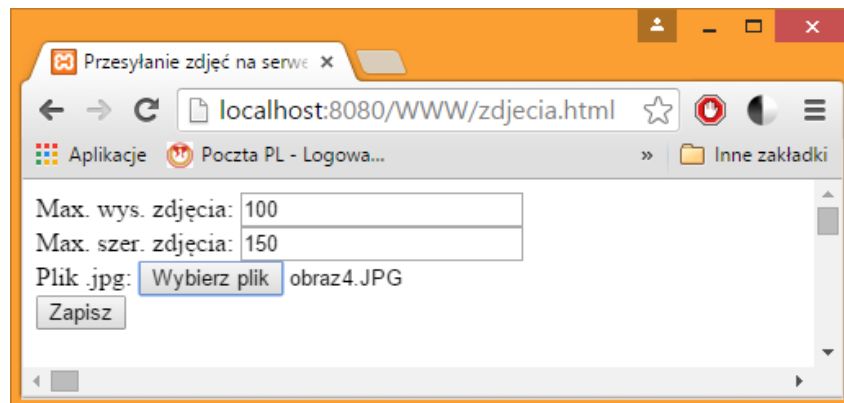
    $dlugosc = strlen($link);
    $dlugosc -= 4;
    $link = substr($link, 0, $dlugosc);

    header('location: ?pic='.$link);
}
else {
    header('location: index.html');
}
}
}

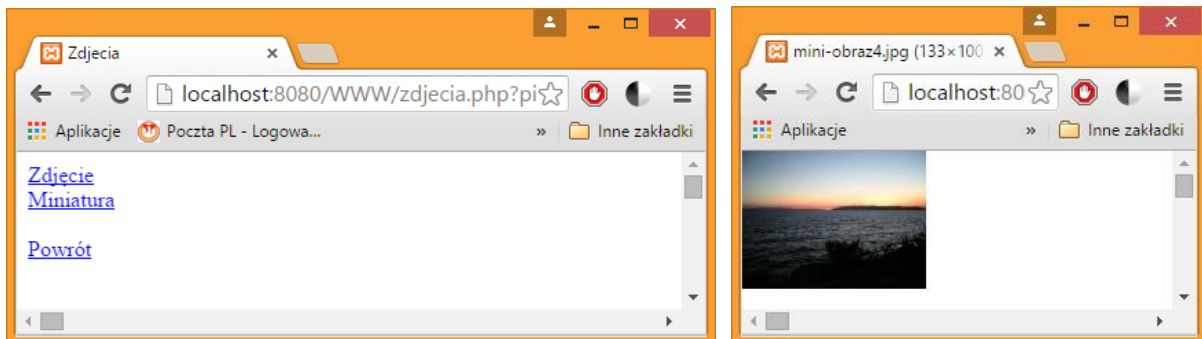
if (isset($_GET['pic']) && !empty($_GET['pic'])) {
    echo '<a href="'. $_GET['pic']. '.jpg">Zdjęcie</a><br/>';
    echo '<a href="mini-'. $_GET['pic']. '.jpg">Miniatura</a><br/><br/>';
    echo '<a href="zdjecia.html">Powrót</a>';
}

```

Ostateczny efekt działania skryptu z Ćw.1. przedstawiają rysunki 1-2.



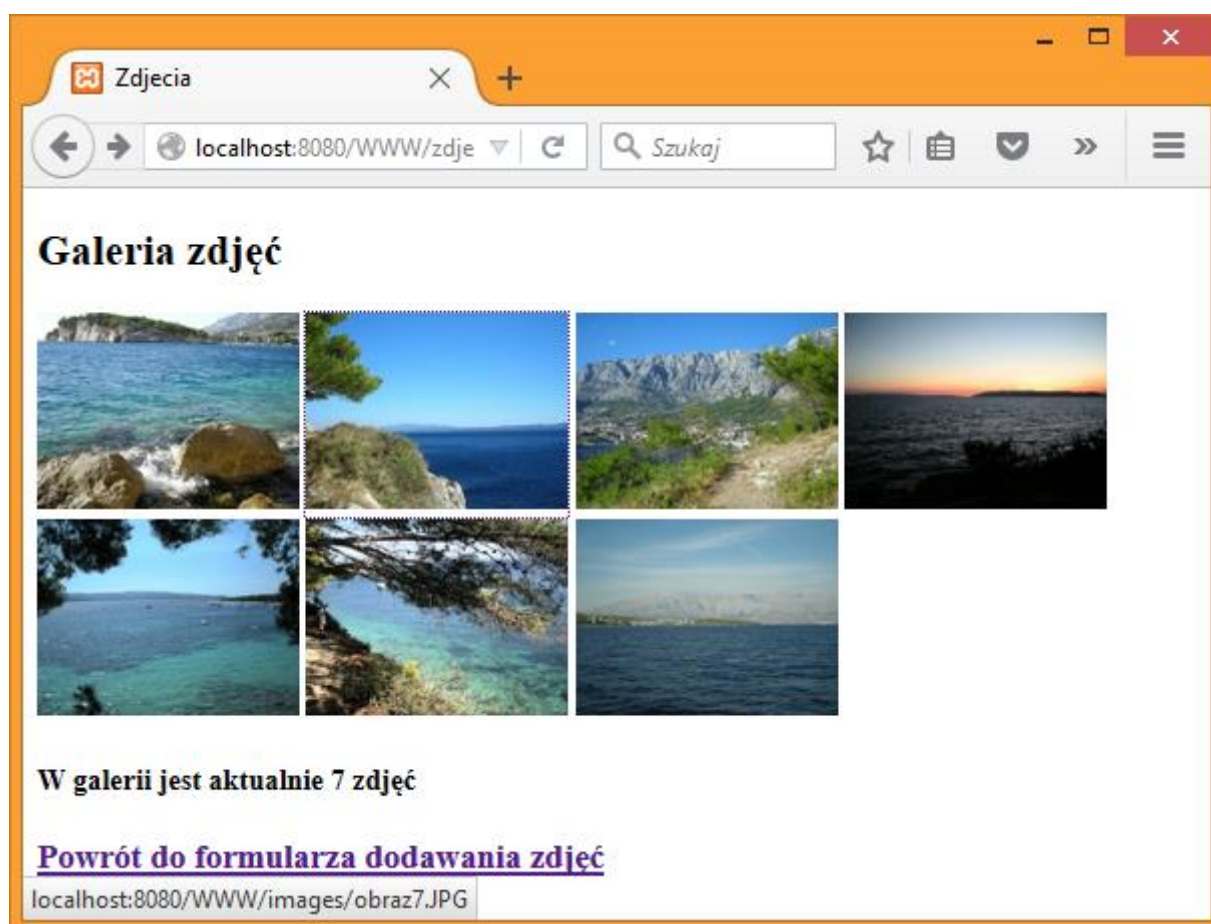
Rys.1. Formularz do przesyłania zdjęć (plik *zdjecia.html*)



Rys.2. Efekt pracy skryptu *zdjecia.php* i widok miniatury zdjęcia przesłanego na serwer

Ćw.2. Tworzenie galerii z wykorzystaniem skryptu z ćw.1.

1. Zmodyfikuj poprzedni skrypt *zdjecia.php* tak aby miniaturki były zapisywane w folderze *miniatura* a zdjęcia w folderze *zdjecia*. Najpierw utwórz te foldery w głównym katalogu projektu. Sprawdź czy wszystko działa poprawnie.
2. Do skryptu *zdjecia.php* dodaj instrukcje wyświetlające miniatury wszystkich zdjęć, które już zostały przesłane na serwer (Rys. 3). W celu realizacji tego zadania skorzystaj z wykładu 4 – praca z plikami i katalogami, przykłady 6 i 6a. Ponadto każda miniatura ma stanowić jednocześnie hiperłącze do zdjęcia oryginalnych rozmiarów.



Rys.3. Ostateczny efekt pracy skryptu `zdjecia.php`- każda miniaturka jest linkiem do oryginalnego zdjęcia