

# Membuat Album Foto Digital Sederhana

Kontribusi: Taufik Abidin  
Tuesday, 26 December 2006

Di era seperti sekarang ini, popularitas kamera non-digital semakin menurun. Orang semakin gemar menggunakan kamera digital bukan saja karena harganya yang semakin terjangkau, tetapi juga karena hasilnya dapat dengan mudah dishare dengan kerabat dan sahabat melalui Internet.

Menampilkan foto-foto digital (images) di website bukanlah pekerjaan yang sulit belakangan ini. Bagi yang memiliki website pribadi dan sudah terbiasa dengan simbol-simbol html (html tag), foto-foto digital dapat ditata dengan baik dalam halaman website. Bagi yang tidak ingin dipusingkan dengan simbol-simbol html, foto-foto digitalnya bisa diupload dalam album foto digital gratis yang belakangan ini cukup banyak tersedia, seperti Yahoo photos, Smugmug, dan Fotki. Namun, bagi 'computer geek' yang gemar menulis program, foto-foto digital biasanya ditampilkan secara otomatis menggunakan script. Pengotomatisan proses 'upload' dan 'display' ini juga didorong oleh faktor jumlah foto digital yang dimiliki (yang biasa berjumlah sangat besar), sehingga mendefinisikan simbol-simbol html satu per satu untuk setiap foto menjadi sangat tidak efisien. Tulisan ini membahas cara membuat album foto digital secara sederhana menggunakan Perl.

## Asumsi dan Deskripsi Program

Dalam tulisan ini, foto-foto digital diasumsikan disimpan dalam sebuah direktori dan database tidak digunakan untuk mengelola foto-foto tersebut. Diasumsikan pula dalam direktori tersebut hanya ada file-file foto digital (dalam format JPG, GIF, atau PNG). Program yang dibahas dalam tulisan ini akan membaca nama-nama file yang ada dalam direktori tersebut secara otomatis, kemudian mengidentifikasi ukuran dari file (dalam kilobyte), dan menampilkan foto-foto digital tersebut satu per satu dalam halaman website. Album foto digital, seperti Yahoo photos dan Smugmug, sudah pasti menggunakan database untuk mengelola data-data foto digital, membangkitkan primary key dan foreign key untuk setiap foto-foto yang dimiliki oleh user tertentu, dan menggunakan indeks untuk mencari dan menampilkan foto secara cepat.

## Implementasi

Berikut adalah kode program yang ditulis menggunakan Perl. Pembahasan secara mendetail dari setiap pernyataan akan dibahas pada bagian pembahasan di bawah ini. Dengan cara ini diharapkan program dapat dengan mudah dipahami.

```
#!/usr/bin/perl -w
#copyright 2006 Keudekupi.com

use strict;
use CGI;
my $cgi = new CGI;
print $cgi->header({-type=>'text/html'});

my $pos = defined($cgi->param('pos')) ? $cgi->param('pos') : 0;
my $imageDir = "../photos";
my @imageSet = ();

collectImages(\@imageSet);

$pos = 0 if $pos < 0;
$pos = @imageSet - 1 if $pos >= @imageSet;

print $cgi->start_html({-title=>"Photos Album Demo"});
displayImage($imageSet[$pos]);
print $cgi->end_html;

sub collectImages{
    my $imageSetRef = shift;
    my $cmd = "ls -Ash1 --reverse $imageDir";
```



membaca isi direktori, mengurutkan file dari file teranyar ke file terusan, dan menyimpannya dalam array `imageSet`. Pembacaan isi direktori dan pengurutan dilakukan menggunakan perintah `ls` yang merupakan perintah menampilkan file dan isi direktori pada sistem operasi Linux atau Unix. Untuk mengetahui secara lebih detail tentang perintah `ls`, ketik man `ls` di shell Linux.

```
sub collectImages{
  my $imageSetRef = shift;
  my $cmd = "ls -Ash1 --reverse $imageDir";
  my $res = ` $cmd `;
  my @entry = split('\n', $res);
  foreach (@entry){
    next if /total/;
    push(@$imageSetRef, $_);
  }
}
```

Pernyataan pertama (pernyataan `shift`) dalam subroutine ini menampung nilai parameter yang dikirim saat subroutine ini dipanggil. Selanjutnya pada baris ketiga dari subroutine ini, skalar `$res` akan menampung hasil dari eksekusi skalar `$cmd`, atau perintah `"ls -Ash1 --reverse $imageDir"`. Berikut adalah salah satu contoh bila perintah `ls` di atas dieksekusi.

```
total 168k
4.0k toplogo.jpg
4.0k right.gif
112k mainlogo.jpg
4.0k left.gif
20k 3za-top-logo.jpg
4.0k 3za-redbody.jpg
12k 3za-logo-with-top.jpg
8.0k 3za-bottom-shadow-curve.jpg
```

Baris pertama menyatakan total ukuran dari file-file yang tersimpan dalam direktori tersebut (dalam kilobytes). Baris kedua dan seterusnya, yang terdiri dari 2 kolom menyatakan ukuran dan nama dari file. Setiap kolom dipisahkan oleh spasi. Pernyataan `foreach` akan menelusuri nilai array `@entry`, dari indeks 0 sampai dengan ukuran array - 1. Setiap nilai yang ditelusuri akan disimpan dalam variabel khusus perl yaitu `$_`. Selanjutnya, setiap nilai dalam skalar `$_` disimpan dalam array `imageSet` yang dikirim ke dalam subroutine ini melalui referensinya (parameter passed by reference), yang ditandai dengan penggunaan backslash (`\`) pada saat parameter ini dikirim ke dalam subroutine:

```
collectImages(\@imageSet);
```

Subroutine `displayImage` akan menampilkan image pada posisi 'pos' di browser. Penulis menyerahkan sepenuhnya kepada pembaca untuk mencoba memahami setiap pernyataan dalam subroutine ini.

## Demo

Demo dari program yang dibahas dalam tulisan ini dapat dilihat disini.

## Penutup

Banyak orang beranggapan kalau pemrograman itu sulit dan mengerikan. Namun dari pengalaman yang penulis peroleh justru pemrograman itu fun! Melalui tulisan ini penulis ingin mengajak pembaca semua yang berkecimpung dalam bidang ilmu komputer agar terus berupaya mengasah kemampuan pemrograman sehingga kita dapat menjadi kelompok masyarakat pembuat aplikasi, bukan hanya menjadi masyarakat pengguna aplikasi. Selamat berkarya!