

Modul 2: jQuery, MySQL dan PHP Data Object

Materi Training

Pengenalan jQuery

jQuery adalah sebuah library yang berisi JavaScript, yang menggunakan penelusuran DOM yang sederhana, event handling, animating dan interaksi Ajax untuk pengembangan web dengan cepat. jQuery dirancang untuk mengubah cara seseorang menuliskan kode JavaScript—lihat <http://jquery.com>. Berikut ini beberapa alasan mengapa kita harus menggunakan jQuery:

- Free dan Open source software
- Lightweight footprint
- CSS3-compliant
- Cross-browser
- Minimal code
- Off-the-shelf plugins

Pengenalan Ajax – Asynchronous JavaScript and XML

Asynchronous JavaScript and XML (AJAX) adalah sebuah teknik pemrograman atau pendekatan yang dijalankan di sisi client untuk menerima data dari server secara asynchronous dalam proses background, tanpa mempengaruhi tampilan dan perilaku halaman web saat ini. Data biasanya diterima menggunakan object XMLHttpRequest. Library jQuery memiliki dukungan penuh untuk AJAX. Fungsi atau method dalam jQuery memungkinkan kita untuk menampilkan data dari server tanpa mengharuskan web browser merefresh halaman webnya.

Berkenalan dengan jQuery.ajax()

Mari kita lihat sebuah contoh dari API jQuery.ajax():

```
$.ajax({
  url: "my_ajax_responder.php",
  type: "POST",
  data: {'name': 'Joni'}, //pasangan kunci dan nilai
  "call=login&name=Joni"
  success: function(xh){
```

```

    //success handler or callback
  },
  error: function(){
    //error handler
  }
});

```

Di dalam fungsi \$.ajax() kita dapat melihat bahwa objek konfigurasi (dibuat menggunakan objek literal dari JavaScript) adalah dilewatkan ke dalamnya, dan konfigurasi dapat didefinisikan sebagai berikut:

- url berisi URL dari server script yang akan digunakan dalam komunikasi
- type berisi tipe HTTP request, GET atau POST
- data berisi data yang akan dikirim ke server dalam sebuah form atau dalam pasangan kunci dan nilai atau sebagai parameter URL.
- Error berisi keterangan jika ada kesalahan dalam pemanggilan AJAX

Mari kita lihat contoh AJAX dari jQuery

```

$.ajax({
  type: "GET",
  url: "test.js",
  dataType: "script"
});

```

Pengantar PHP Data Object (PDO)

PHP Data Object (PDO) adalah pendefinisian ekstention sederhana dan interface yang konsisten untuk mengakses database dalam PHP. PDO menyediakan sebuah layer data-access abstraction, yang berisi database yang akan anda gunakan dan pendefinisian function untuk menjalankan query baca dan tulis ke table.

Membuat Aplikasi Web Poster Status

Sebagian besar platform social networking , seperti Facebook, Twitter dan Google Plus, menyediakan fitur posting status antar teman-teman pengguna, dan memungkinkan pengguna melihat status teman-teman mereka. Pada training kali ini kita akan membuat project yang mirip dengan cara kerja jaringan sosial tersebut. Mari kita mengimplementasikan sedikit fitur yang mirip dengan sebuah platform social networking, katakanlah Facebook.


Perancangan Aplikasi Web Poster Status

Berikut ini adalah tampilan web yang akan dibuat dalam training PHP kali ini.


Status Poster

Write your status here


[Share](#)



Joko Dolog
Belajar PHP...
at 3:47 AM on May 8th, 2015



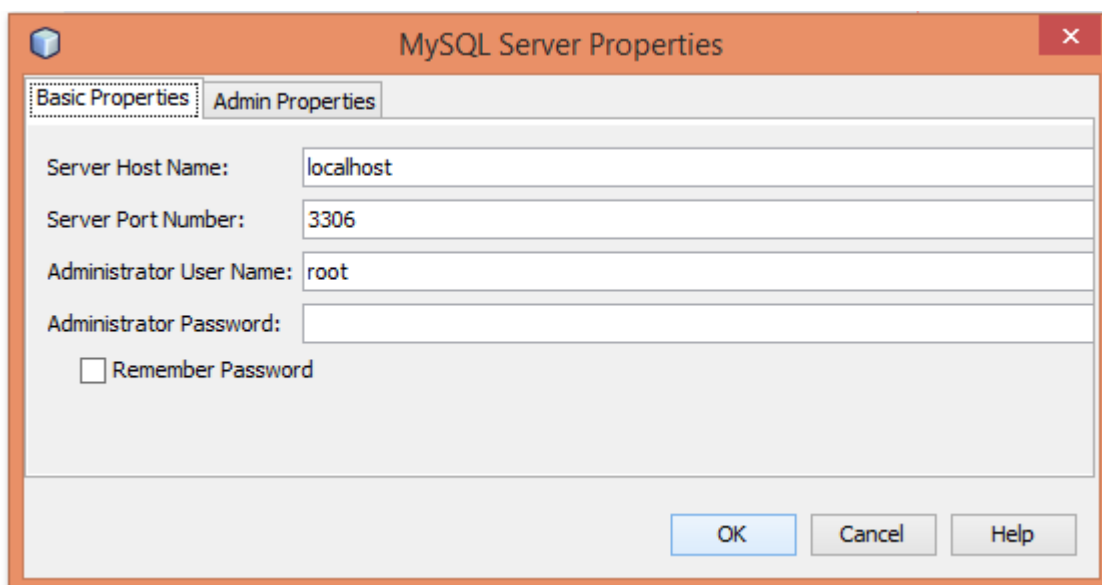
John Megantara
Belajar AJAX...
at 3:46 AM on May 8th, 2015



Teguh Sutanto
Yuk bikin program Sosmed sendiri
at 3:41 AM on May 8th, 2015

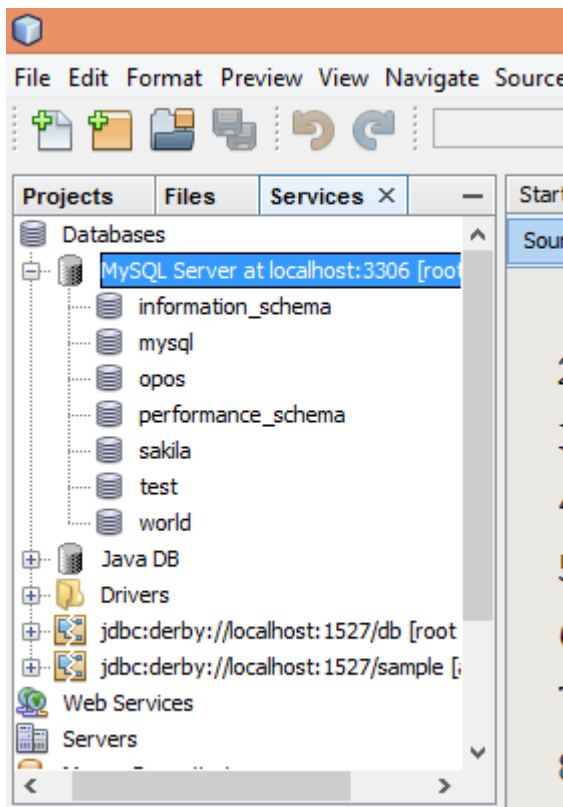
Lab 0: Persiapan—Setting Database

- Untuk pertama kali, kita akan membuat koneksi ke database MySQL di dalam IDE Netbeans; tekan Ctrl+5 untuk memasuki window Services, kemudian pilih node Database, klik kanan pada MySQL Database Server dan pilih Properties untuk membuka window MySQL Server Properties, lihat tampilan berikut:

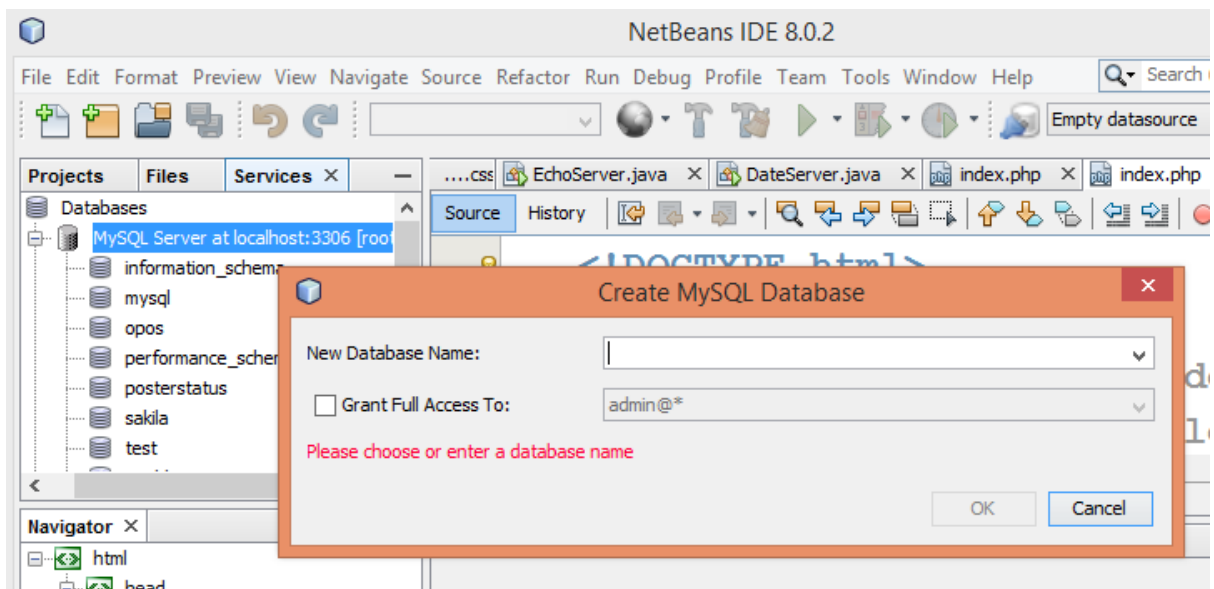


- Masukan setting alamat host, port, user name dan password kemudian pilih tombol OK

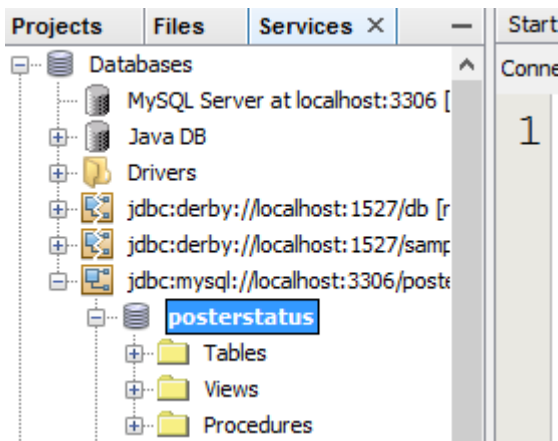
3. Kalau konfigurasi yang anda tuliskan pada langkah di atas tidak ada kesalahan maka akan bisa dilihat daftar database yang ada di server MySQL, lihat gambar berikut:



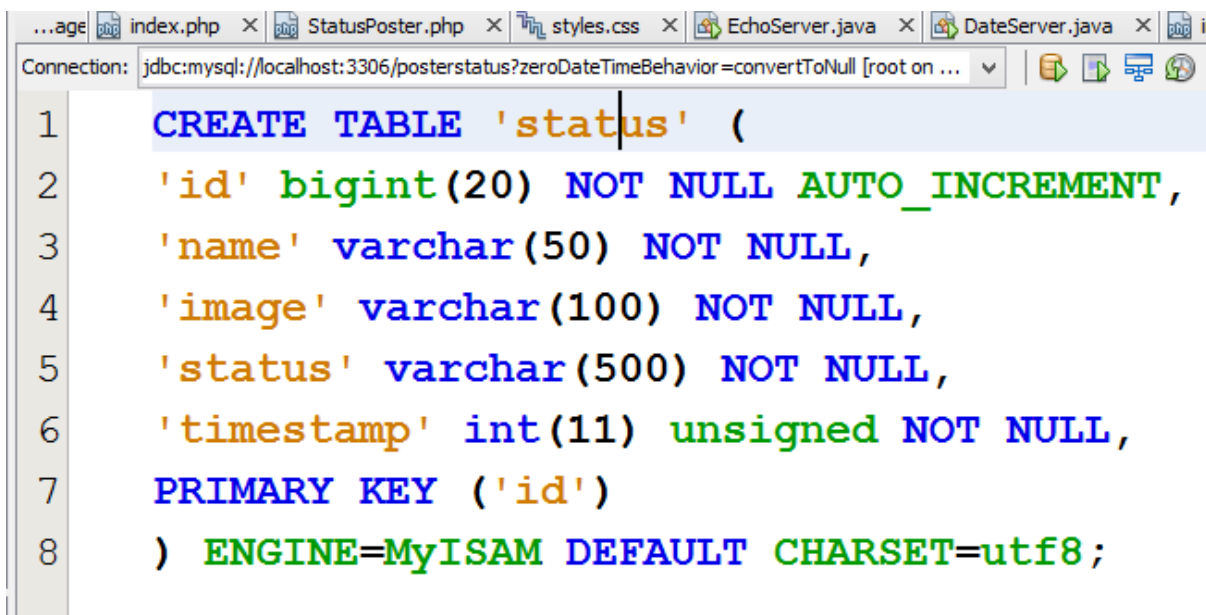
4. Masih di dalam window Service, pilih node MySQL kemudian klik kanan dan pilih Create Database, lihat gambar berikut:



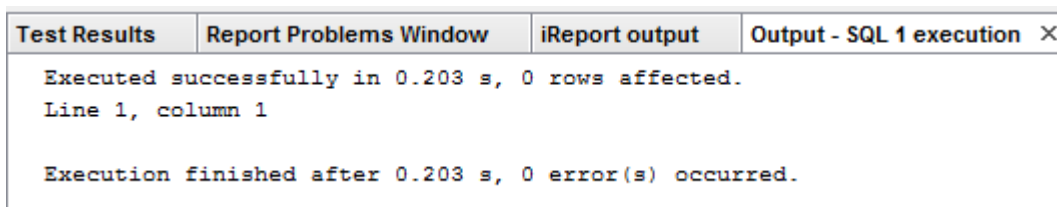
5. Masukkan nama database posterstatus (Field New Database Name) kemudian pilih OK
6. Pilih database posterstatus sehingga muncul tampilan sebagai berikut:



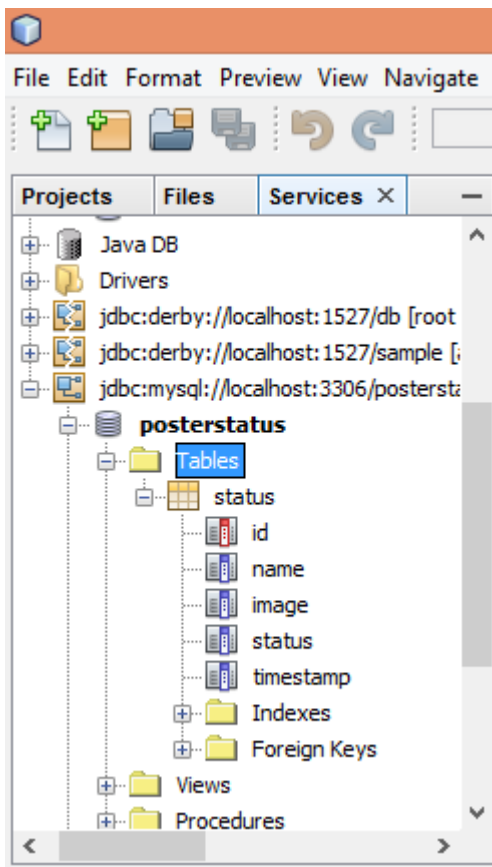
7. Tambahkan sebuah tabel dalam database posterstatus dengan cara: pilih Tables, kemudian Execute Command ... untuk menjalankan perintah SQL:



8. Untuk menjalankan query gunakan tombol Execute SQL pada toolbar yang telah disediakan atau dengan menekan tombol keyboard Ctrl+Shift+E. Jika perintah SQL berhasil dijalankan maka akan menampilkan keterangan seperti berikut:



9. Setelah berhasil menjalankan query create table maka akan terbentuk tabel status dengan properti sebagai berikut:



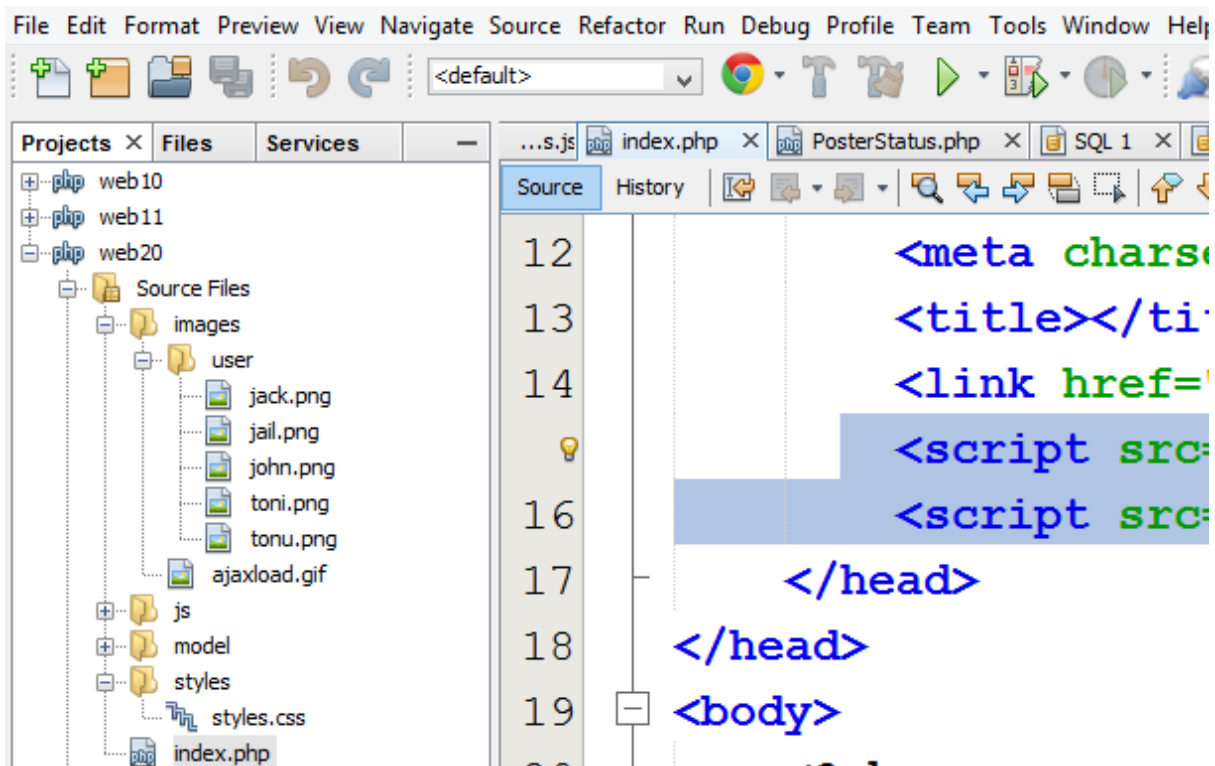
10. Tambahkan beberapa baris data ke dalam tabel status, lihat gambar berikut:

select * from status ×

Page Size: 20 | Total Rows: 3 | Page: 1 of 1 | Matching Rows:

#	id	name	image	status	timestamp
1	50	Teguh Sutanto	tonu.png	Yuk bikin program Sosmed sendiri	1431031277
2	51	John Megantara	toni.png	Belajar AJAX...	1431031599
3	52	Joko Dolog	jack.png	Belajar PHP...	1431031661

Lab 1: Membuat PHP Application Project dengan Netbeans



1. Buat project PHP baru dengan nama web20
2. Tambahkan folder images, model, js dan styles.
 - a. Folder images digunakan untuk meletakkan gambar yang akan digunakan dalam aplikasi web
 - b. Folder model berisi deklrarsi class atau file PHP yang akan disertakan dalam halaman web yang berfungsi sebagai model dari aplikasi
 - c. Folder js digunakan untuk menyimpan file JavaScript
 - d. Folder styles digunakan menyimpan file-file StyleSheet
3. Tambahkan sebuah PHP Class dengan nama PosterStatus di dalam folder model
4. Tambahkan constanta dan function berikut ini ke dalam class PosterStatus

```

class PosterStatus {
    private $db = NULL;
    const DB_SERVER = "localhost";
    const DB_USER = "su";
    const DB_PASSWORD = "supersuper";
    const DB_NAME = "posterstatus";
    public function __construct() {
        ;
    }
}

```

- Untuk melakukan koneksi ke database melalui PDO lakukan implementasi terhadap konstruktort dengan menuliskan kode seperti berikut:

```
public function __construct() {
    $dsn =
    'mysql:dbname='.self::DB_NAME.';host='.self::DB_SERVER;
    try {
        $this->db = new PDO($dsn, self::DB_USER, self::DB_
        PASSWORD);
    } catch (PDOException $e) {
        throw new Exception('Connection failed: ' .
        $e->getMessage());
    }
    return $this->db;
}
```

- Untuk menampilkan status yang sudah tersimpan dalam tabel status maka kita akan membuat sebuah method yang didalamnya akan menjalankan perintah query select.

```
public function getStatusPosts() {
    $statement = $this->db->prepare("SELECT name, image,
    status,timestamp FROM status ORDER BY timestamp DESC,id");
    $statement->execute();
    if ($statement->rowCount() > 0) {
        return $statement->fetchAll();
    }
    return false;
}
```

- Pada bagian akhir class tambahkan satu baris perintah untuk membuat instance dari class posterstatus, lihat potongan program berikut:

```
$status = new StatusPoster();
```

Lab 2: Menampilkan isi tabel status dalam halaman web

- Tambahkan sebuah file CSS dalam folder styles dengan nama style.css
- Tuliskan kustomisasi tampilan HTML sebagai berikut dalam file style.css

```
body {
    font-family:Arial,Helvetica,sans-serif;
    font-size:12px;
}
h1,input {
    color:#fff;
    background-color:#1A3C6C;
}
h1,input,textarea,.inputbox,.postStatus {
    padding:5px;
}
```



```
input,textarea,ul li img,.inputbox {
border:1px solid #ccc;
}
ul li {
width:100%;
display:block;
border-bottom:1px solid #ccc;
padding:10px 0;
}
ul li img {
padding:2px;
}
.container {
width:60%;
float:none;
margin:auto;
}
.content {
padding-left:15px;
}
.content a {
font-weight:700;
color:#3B5998;
text-decoration:none;
}
.clearer {
clear:both;
}
.hidden {
display:none;
}
.left {
float:left;
}
.right {
float:right;
}
.localtime {
color:#999;
}
.inputbox {
height:70px;
margin:15px 0;
}
.inputbox textarea {
width:450px;
height:50px;
overflow:hidden;
}
.inputbox input {
margin-right:30px;
width:50px;
}
```

3. Buka file index.php kemudian tambahkan kode program PHP berikut di bagian atas program index.php

```
<?php
define('BASE_URL', 'http://localhost/chapter3/');
?>
```

4. Kemudian tambahkan seting untuk link CSS pada bagian <head>:

```
<link href="<?=BASE_URL?>styles/styles.css" media="screen"
rel="stylesheet" type="text/css" />
```

5. Tambahkan juga konfigurasi untuk JQuery:

```
<script src="js/jquery.js"></script>
<script src="<?= BASE_URL ?>js/status.js"></script>
```

6. Tambahkab sebuah file JavaScript dengan nama status.js pada foldel js, tuliskan kerangka program sebagai berikut:

```
$(document).ready(function ($)
{
var Status = {
};
});
```

7. Mari menuliskan sebuah fungsi dalam objek status untuk menampilkan currenttime seperti berikut:

```
currentTime: function (timestamp) {
if (typeof timestamp !== 'undefined' && timestamp !== '')
var currentTime = new Date(timestamp * 1000);
else
var currentTime = new Date();
var hours = currentTime.getHours();
var minutes = currentTime.getMinutes();
var timeStr = '';
if (minutes < 10) {
minutes = "0" + minutes
}
timeStr = ((hours > 12) ? (hours - 12) : hours) + ":" +
minutes + ' ';
if (hours > 11) {
timeStr += "PM";
} else {
timeStr += "AM";
}
return timeStr;
},
```

8. Tambahkan method currentDate() dalam objek status:

```
currentDate: function (timestamp) {
var m_names = ["January", "February", "March", "April",
"May",
"June", "July", "August", "September", "October", "November",
"December"];
if (typeof timestamp !== 'undefined' && timestamp !== '')
var d = new Date(timestamp * 1000);
else
var d = new Date();
var curr_date = d.getDate();
var curr_month = d.getMonth();
var curr_year = d.getFullYear();
var sup = "";
if (curr_date === 1 || curr_date === 21 || curr_date === 31)
{
sup = "st";
}
else if (curr_date === 2 || curr_date === 22)
{
sup = "nd";
}
else if (curr_date === 3 || curr_date === 23)
{
sup = "rd";
}
else
{
sup = "th";
}
return m_names[curr_month] + ' ' + curr_date + sup + ', ' +
curr_year;
},
```

9. Tambahkan getLocalTimerStr()

```
getLocalTimeStr: function (gmtTimestampInSec) {
return 'at ' + this.currentTime(gmtTimestampInSec)
+ ' on ' + this.currentDate(gmtTimestampInSec);
}
```

10. Buka file index.php kemudian tambahkan kode berikut:

```
<body>
<div id="container" class="container">
<h1>Status Poster</h1>
<ul>
</ul>
</div>
</body>
```

11. Dibawah tag <!DOCTYPE html> tambahkan kode berikut:

```
<?php
require_once 'StatusPoster.php';
$result = $status->getStatusPosts();
define('BASE_URL', 'http://localhost/chapter3/');
?>
```

12. Untuk menampilkan semua status maka akan tambahkan proses perulangan untuk semua record, seperti berikut:

```
<?php
if (is_array($result))
foreach ($result as $row) {
echo '
<li>
<a href="#">

</a>
<div class="content left">
<a href="#">' . $row['name'] . '</a>
<div class="status">' . $row['status'] . '</div>
<span class="localtime" data-timestamp="' .
$row['timestamp'] . '"></span>
</div>
<div class="clearer"></div>
</li>
';
}
?>
```

13. Untuk menampilkan `Status.showLocalTimer()` secara cepat, maka harus dilakukan pemanggilan method tersebut pada bagian akhir dari method `ready()`.

```
$(document).ready(function ($) {
var Status = {
//whole library methods...
};
Status.showLocalTime();
});
```

14. Sekarang waktunya mencoba aplikasi, silakan dijalankan dan lihat hasilnya.