



yiiframework

50 Yii Framework Tip

50 เทคนิคพิเศษสำหรับ Yii Framework ที่คุณอาจไม่เคยเห็นมาก่อนที่จะช่วยให้พัฒนา Application อย่างมืออาชีพมากยิ่งขึ้น

คำนำ

หนังสือเล่มนี้เป็นหนังสืออีกเล่มของ Yii Framework โดยเป็นหนังสือที่รวบรวมเอาเทคนิคต่างๆ เพื่อใช้ในการสร้าง Application ให้มีความรวดเร็วและสมบูรณ์แบบมากยิ่งขึ้น โดยกำหนดตั้งแต่พื้นฐานไปจนถึงระดับสูง โดยแบ่งการนำเสนอออกเป็นบท ซึ่งในแต่ละบทก็จะมีเทคนิคต่างๆ กันไป

หนังสือเล่มนี้เป็นหนังสือที่ผมและทีมงานตั้งใจเขียนและถ่ายทอดเป็นอย่างมาก โดยเราทำงานอย่างหนักหลายเดือนเพื่อให้หนังสือเล่มนี้มีความครบถ้วนสมบูรณ์มากที่สุด ซึ่งก็หวังเป็นอย่างยิ่งว่าจะเป็นหนังสือใกล้มือเพื่อต่อเติม Application ของท่านให้มีความสมบูรณ์แบบมากยิ่งขึ้น

ท้ายที่สุดต้องขอขอบคุณทุกท่านที่ทำให้หนังสือเล่มนี้เกิดขึ้น ขอคุณน้องวีระยุทธ บุญอ่อน ที่ช่วยเรื่อง Code Action Proof ขอคุณน้องณัฐวุฒิ เกษศรีรัตน์ ที่ช่วยในเรื่อง Graphic และที่สำคัญต้องขอขอบคุณเพื่อนๆ ชาว Programmer Thailand ทุกๆ ท่านที่ช่วยสนับสนุนพวกเรา ด้วยดีเสมอมา ขอคุณมากๆ ครับ

มานพ กองอุ่น
และทีมงาน Programmer Thailand

สารบัญ

คำนำ.....	1
สารบัญ	2
บทที่ 1 ขั้นพื้นฐาน.....	6
การติดตั้ง Yii Framework	6
การดาวน์โหลด.....	6
การติดตั้ง.....	6
การสร้าง Project Skeleton.....	10
การสร้าง Project ใน Netbeans IDE.....	11
การติดตั้ง Yii Snippets ใน Sublime Text 3	13
การติดตั้ง Yii Eclipse PlugIn	15
การออกแบบ ER Diagram และการ Forward Engineering	20
การออกแบบ ER Diagram.....	20
การสร้างฐานข้อมูล.....	23
การ Forward Engineering.....	24
พจนานุกรมข้อมูล (Data Dictionary).....	28
การสร้าง Attribute Label ให้เป็นภาษาไทยโดยอัตโนมัติ.....	30
การกำหนด Yii Message ให้เป็นภาษาไทย	33
การใช้งานรูปภาพ	34
ตัวอย่างการใช้งาน	34
การใช้งาน Link.....	36

ตัวอย่างการใช้งาน	36
บทที่ 2 ว่าด้วยเรื่อง Extension	38
การตัดคำด้วย XReadmore	38
การติดตั้ง Yii Booster Extension.....	40
การติดตั้ง User Extension และ Rights Extension	44
การติดตั้ง User Extension	44
การติดตั้ง Rights Extension.....	47
การใช้งาน User และ Rights Extension เบื้องต้น	51
การส่ง Email ด้วย SMTP Extension	55
การติดตั้ง.....	55
การใช้งาน.....	56
บทที่ 3 ว่าด้วยเรื่องของ Model.....	57
Database Active Record.....	57
Create.....	57
Read.....	57
SLUG	59
TAGABLE.....	61
บทที่ 4 ว่าด้วยเรื่องของ Controller.....	70
การสร้าง Criteria ในการ Select ข้อมูล.....	70
การสร้างระบบ User Login จากฐานข้อมูล	72
บทที่ 5 ว่าด้วยเรื่องของ View	73
การ Register ไล่วาง JavaScript และ CSS	73
การ Register Core Script.....	73
การ Register JavaScript ไล่วาง.....	73

การ Register ໄວ້ CSS	74
การ Register Script ที่เป็ JavaScript แอ Inline	75
การเปลี่ยนข้อความ “Home” ใน Breadcrumbs เป็น “หน้าหลัก”	76
การปรับ Breadcrumbs ให้อยู่ในรูปแบบของ Tag ul และ li	77
การสร้าง Drop Down List สำหรับการเก็บข้อมูลแบบ ENUM	78
การสร้าง Carousel (Carousel) ใน Yii Booster Extension	79
การสร้าง CKEditor สำหรับ TextArea	80
การดึงข้อมูลจากตารางอื่นมาแสดงเป็น Drop Down List	82
การเปลี่ยน site/index เป็น home ใน URL	83
การซ่อน ໄວ້ index.php เมื่อเรียก URL	84
การสร้าง Captcha ใหักับฟอร์มด้วย CCaptcha	86
การสร้างฟอร์มแบบ Horizontal สำหรับ Yii Booster Extension	88
การแสดงผลจากตารางอื่นในความสัมพันธ์แบบ BelongsTo	89
การสร้าง Date Picker สำหรับเก็บค่า Field DATE ใน MySQL	92
การแสดงผลวันเวลาภาษาไทยจากฐานข้อมูล	93
การแก้ไขหน้า Error สำหรับ Yi Framework	94
การสร้างและแสดง Flash Message สำหรับ Yii Booster Extension	95
การสร้าง AJAX Button	97
การสร้าง AJAX Link	99
การสร้าง Facebook Like Button	101
สร้าง Application ใน Facebook	101
สร้าง Facebook Like Button	102
บทที่ 6 ว่าด้วยเรื่องของการงาน	105
การแสดงผลข้อมูลแบบ JSON	105

การสร้างกราฟด้วย jqWidgets.....	106
การแสดงผลกราฟด้วย Highcharts JS.....	109
การ Export ข้อมูลแบบ CSV.....	112
การสร้างกราฟด้วย RGraph Extension	114
การสร้างกราฟด้วย Google Chart Extension	116
Pie Chart.....	116
Gauge.....	117
Line Chart	117
บทที่ 7 กรณีตัวอย่าง.....	119
การสร้าง Drop Down List เลือกจังหวัดโชว์อำเภอ เลือกอำเภอโชว์ตำบล แบบ Dependence (AJAX).....	119
provinces.....	119
districts.....	119
tambons	120
การสร้าง sitemap.xml ใน Yii Framework ส่งให้ Google Webmaster Tool และ Search Engine.....	124
การทำเว็บไซต์หลายภาษาใน Yii Framework.....	128
การ Upload ไฟล์รูปภาพ.....	132
การสร้างเมนูจากตารางในฐานข้อมูล MySQL.....	134
การสร้าง Webservice ด้วย SOAP.....	136
บรรณานุกรม.....	139

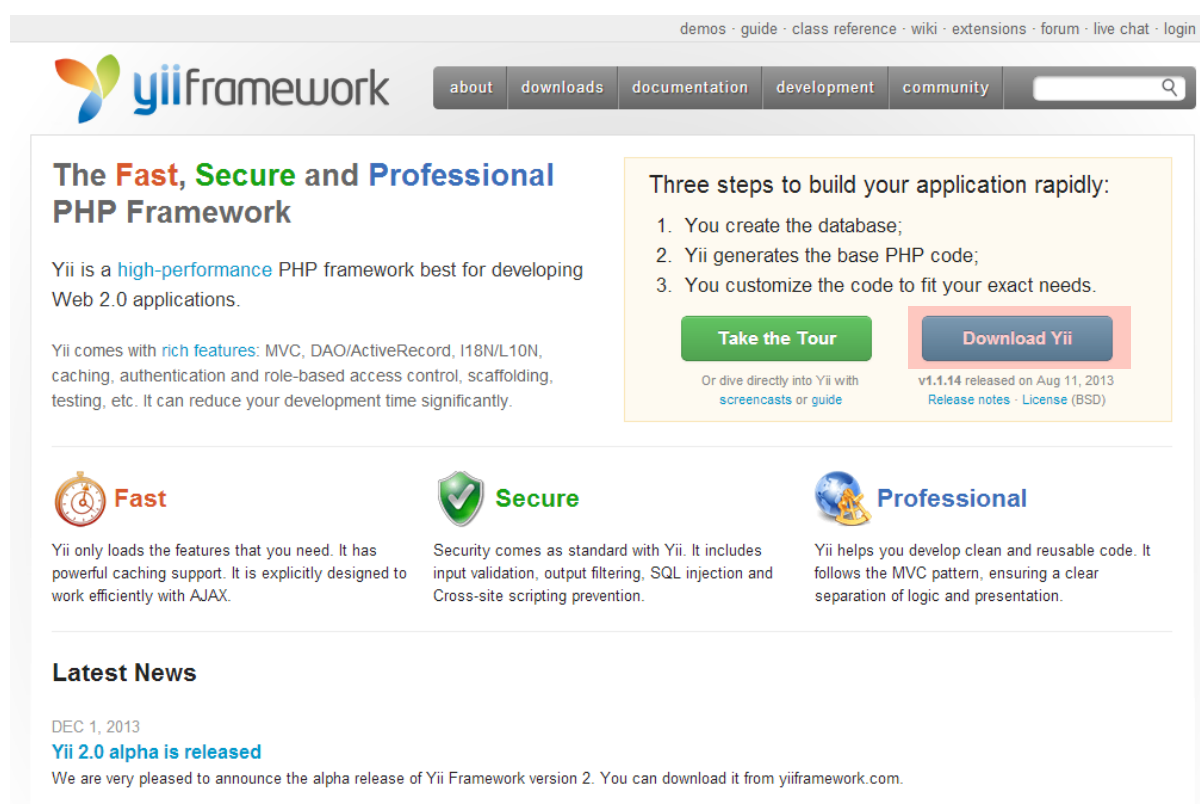
บทที่ 1 ขั้นพื้นฐาน

การติดตั้ง Yii Framework

ก่อนอื่นขอแนะนำเรื่องของการติดตั้ง Yii Framework กันก่อนนะครับ เพื่อให้สามารถเรียนรู้และเข้าใจได้มากยิ่งขึ้น

การดาวน์โหลด

ในการดาวน์โหลดนั้นไปที่เว็บไซต์ <http://www.yiiframework.com>



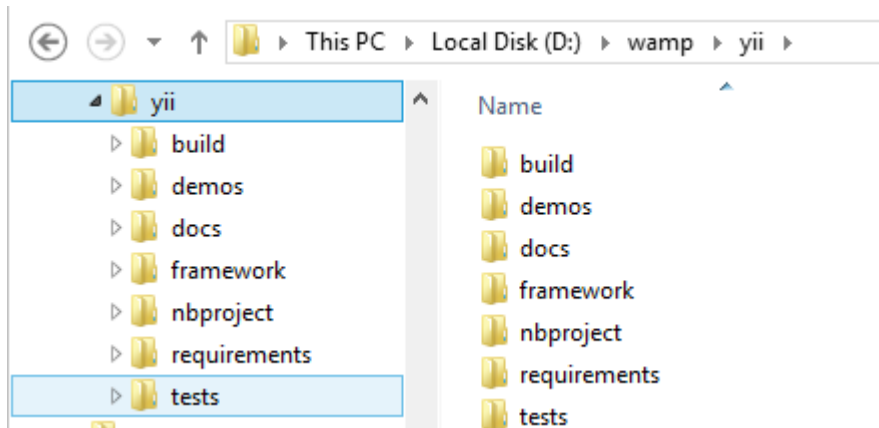
The screenshot shows the official Yii Framework website. At the top, there's a navigation bar with links: demos, guide, class reference, wiki, extensions, forum, live chat, and login. Below this is the Yii Framework logo and a search bar. The main content area features the headline "The Fast, Secure and Professional PHP Framework". It describes Yii as a high-performance PHP framework for Web 2.0 applications. A list of features includes MVC, DAO/ActiveRecord, I18N/L10N, caching, authentication, and role-based access control. To the right, a box titled "Three steps to build your application rapidly:" lists: 1. You create the database; 2. Yii generates the base PHP code; 3. You customize the code to fit your exact needs. Below this are buttons for "Take the Tour" and "Download Yii". The "Download Yii" button is highlighted with a red border. Below the buttons, it says "v1.1.14 released on Aug 11, 2013" with links for "Release notes" and "License (BSD)". Further down, there are three columns: "Fast" (with a clock icon), "Secure" (with a shield icon), and "Professional" (with a globe icon). Each column has a brief description of the feature. At the bottom, there's a "Latest News" section with a date "DEC 1, 2013" and a headline "Yii 2.0 alpha is released". The text below the headline says: "We are very pleased to announce the alpha release of Yii Framework version 2. You can download it from yiiframework.com."

แล้วคลิกที่ปุ่ม Download Yii เพื่อทำการดาวน์โหลด

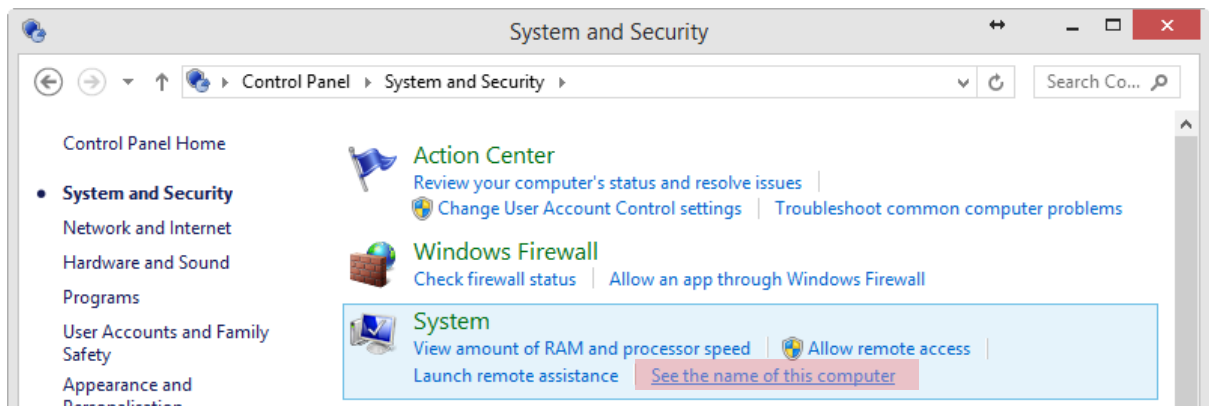
การติดตั้ง

ในการติดตั้งนั้นในที่นี้เราจะติดตั้ง Yii Framework ให้เข้ากับ WAMP Server โดยสามารถดาวน์โหลดและติดตั้ง WAMP Server ได้ที่เว็บไซต์ www.wampserver.com ในที่นี้ติดตั้งไว้ที่ d:\wamp

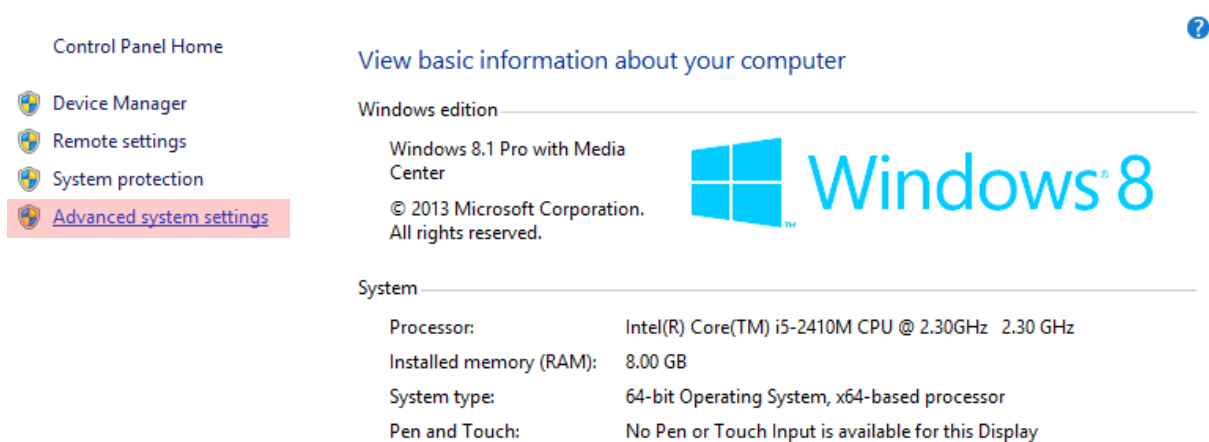
หลังจากที่ดาวน์โหลด Yii Framework เรียบร้อยแล้ว ให้แตก Zip ไฟล์ไปยัง d:\wamp\yii



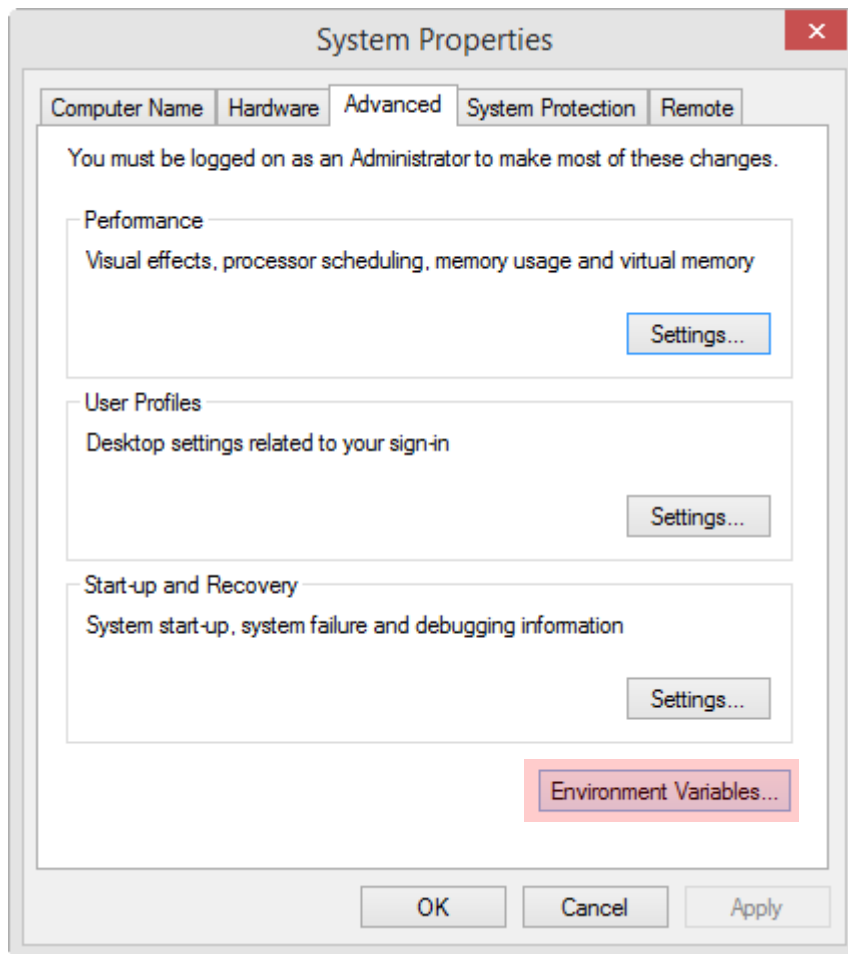
จากนั้นไป Set System Variable ในส่วนของ Path โดยไปที่ Control Panel -> System and Security



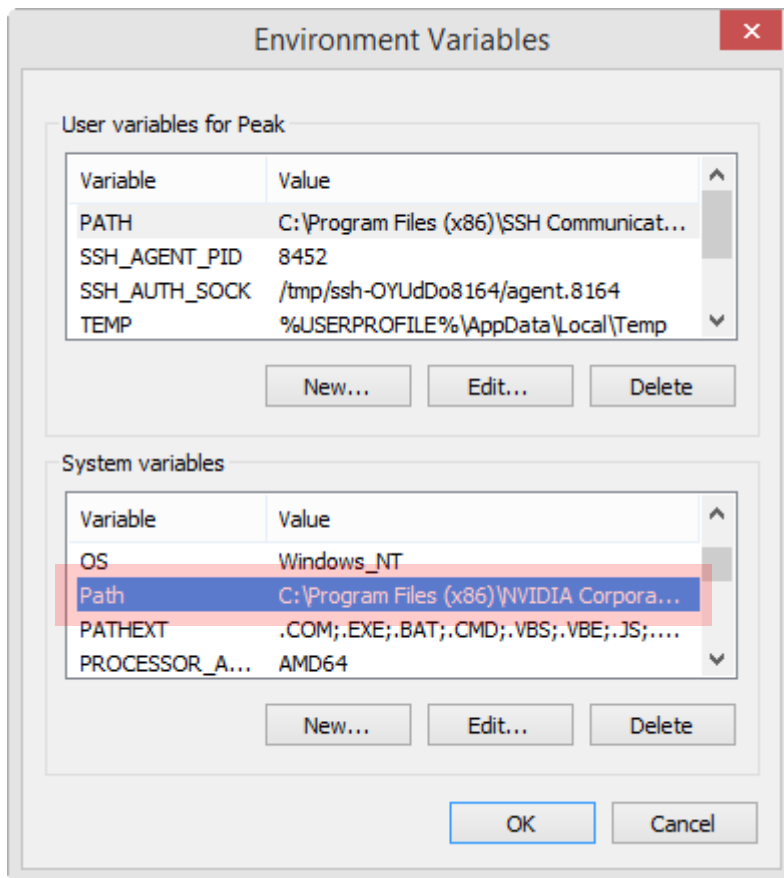
จากนั้นกดที่ See the name of this computer



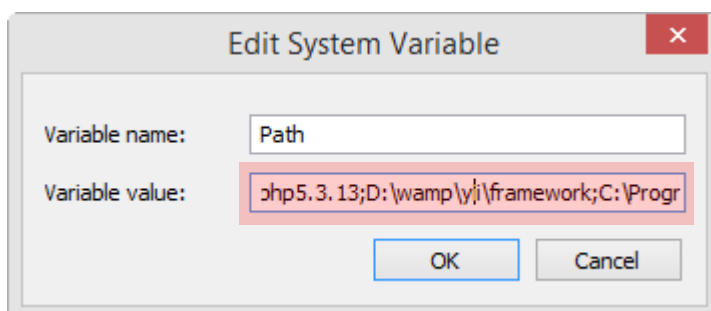
เลือก Advanced system settings



กดเลือก Environment Variables...



เลือก Path ในส่วนของ System variables แล้วกด Edit...



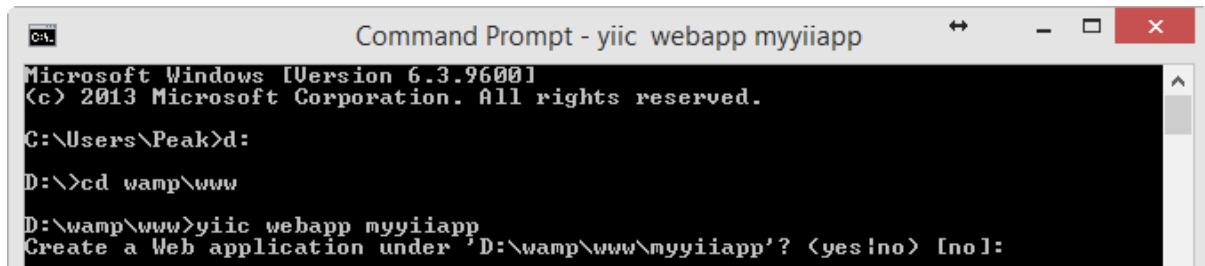
จากนั้นนำ Path ของ PHP และ Framework มาใส่ ในที่นี้คือ

D:\wamp\bin\php\php5.3.13; D:\wamp\yii\framework;

หมายเหตุ Version ของ PHP นั้นอาจแตกต่างกันได้ และหลังจากตั้งค่าเสร็จเรียบร้อยแล้วอาจจะต้อง Restart ก่อนจึงจะสามารถใช้คำสั่ง yiii ได้

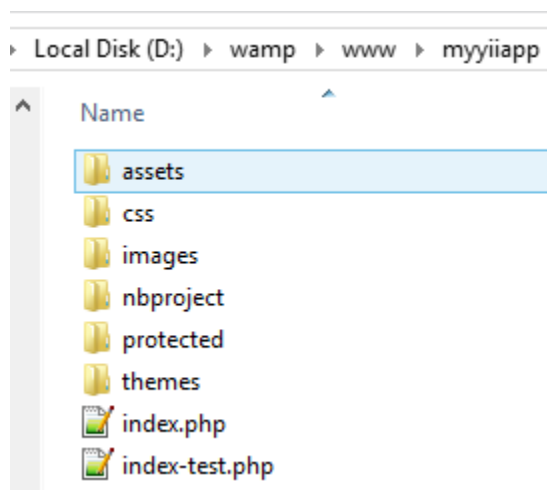
การสร้าง Project Skeleton

การสร้าง Project Skeleton หรือการสร้างโครงสร้างไฟล์ต่างๆ ของ Application ใหม่ ก่อนการเขียนโปรแกรม โดยเปิดโปรแกรม Command Prompt แล้ว CD ไปที่ D:\wamp\www จากนั้นพิมพ์คำสั่งสำหรับการสร้าง Project Skeleton ดังนี้ yiic webapp myyiiapp สำหรับ myyiiapp คือชื่อ Project เรายื่นเอง



```
Command Prompt - yiic webapp myyiiapp
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
C:\Users\Peak>d:
D:\>cd wamp\www
D:\wamp\www>yiic webapp myyiiapp
Create a Web application under 'D:\wamp\www\myyiiapp'? <yes|no> [nol:]
```

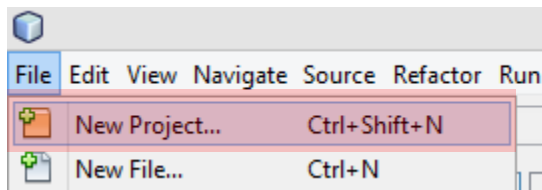
จากนั้นกดปุ่ม Enter แล้วกด yes เพื่อสร้าง ก็จะได้โครงสร้างไฟล์ต่างๆ ดังนี้



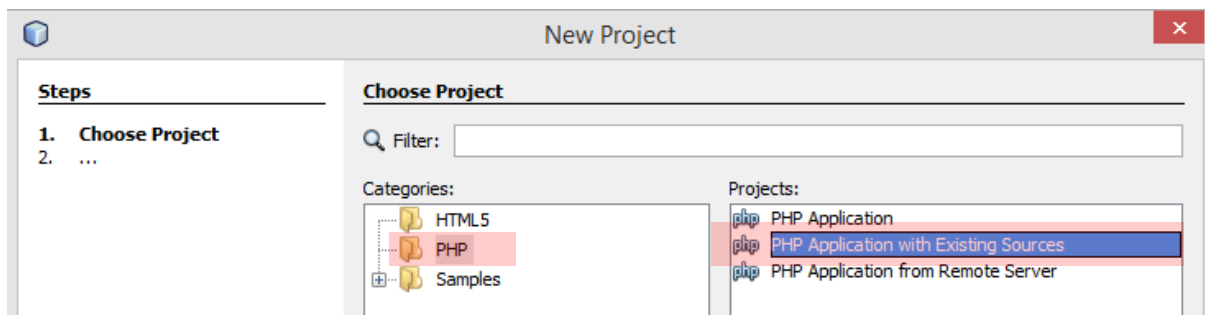
การสร้าง Project ใน Netbeans IDE

Netbean IDE เป็น Software ที่เป็น Integrated Development Environment หรือสภาพแวดล้อมของการพัฒนา Application เราโดยสามารถดาวน์โหลดได้ที่

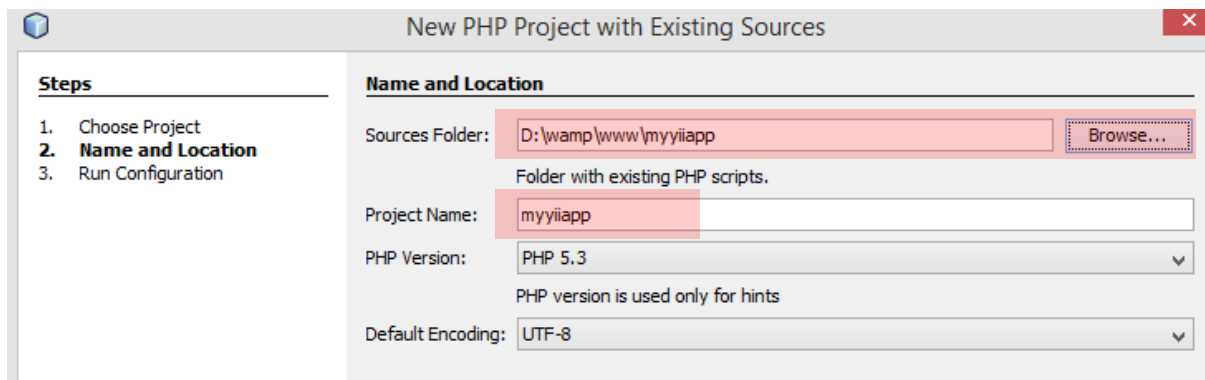
<http://www.netbeans.org> การสร้าง Project ใน Netbean นั้นไปที่ File->New Project



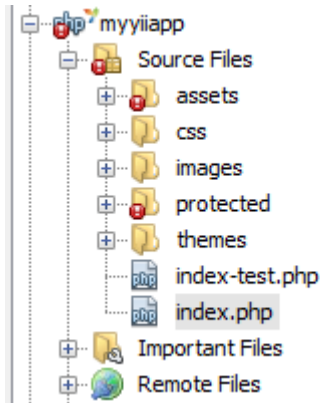
จากนั้นเลือก PHP Application with Existing Sources



จากนั้น Browse ไปยัง Folder myyiiapp



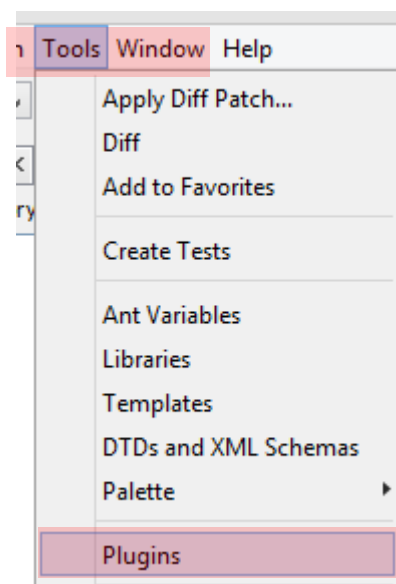
แล้วกดปุ่ม Finish จะเห็นว่าด้านซ้ายมือมีรายการไฟล์ต่างๆ ใน Project เรียบร้อยแล้ว



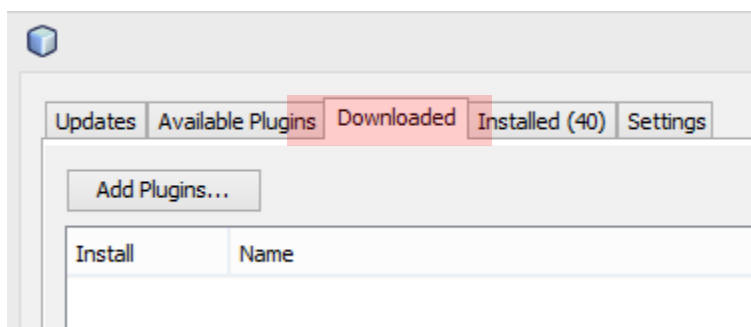
หมายเหตุ สำหรับ Yii Framework จะมี PlugIn ใน Netbeans IDE ซึ่งสามารถดาวน์โหลดมาติดตั้งได้ฟรี จะช่วยให้สามารถเรียกใช้งาน Library ของ Yii Framework ได้ง่ายยิ่งขึ้น ดาวน์โหลดได้ที่

<http://plugins.netbeans.org/plugin/47246/php-yii-framework-netbeans-phpcc>

การติดตั้ง ไปที่ Tools->PlugIns



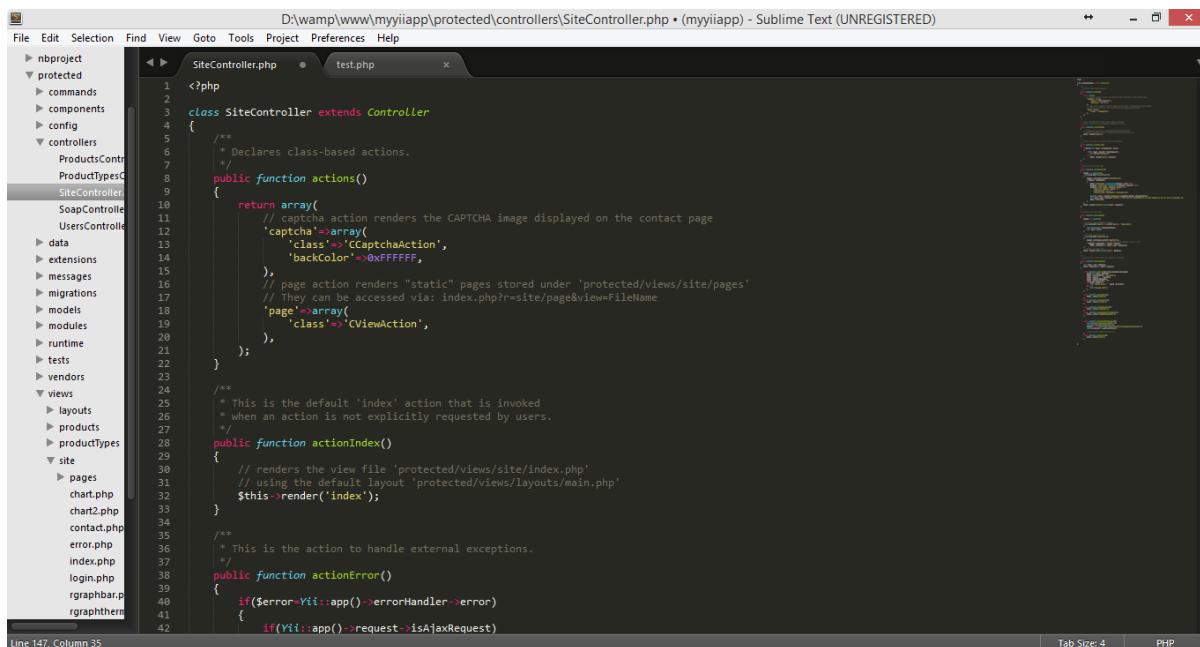
จากนั้นไปที่ Tab Downloaded



แล้วกด Add Plugins... เลือก PlugIn ที่ดาวน์โหลด แล้วกด Install

การติดตั้ง Yii Snippets ใน Sublime Text 3

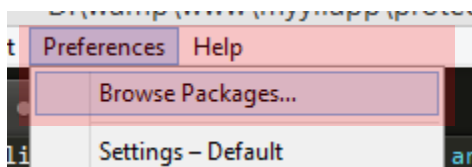
Sublime Text 3 เป็น Editor อีกตัวหนึ่งที่น่าสนใจ เนื่องจากสามารถใช้ได้ฟรี และมีความรวดเร็วในการเปิดใช้งาน












ใน Sublime Text 3 จะมี Code ลักษณะ Auto Complete ที่เรียกว่า Snippet อยู่สำหรับวิธีการใช้งาน เช่นพิมพ์ hlink แล้วกด tab ก็จะมีปรากฏ Code `<?php echo CHtml::link('text', array(), array('optionName'=>optionValue)); ?>` ขึ้นมาให้ทันที

สำหรับ Snippet สำหรับ Yii Framework นั้นสามารถดาวน์โหลดได้ที่ <https://github.com/dexnode/sublime-yii-snippets>

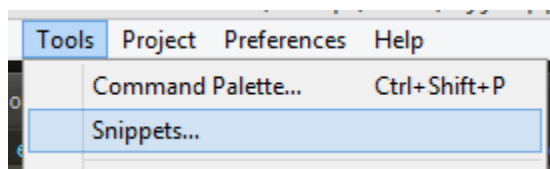
สำหรับการติดตั้งให้ไปที่เมนู Preferences->Browse Packages...



จากนั้นนำ snippets ที่ดาวน์โหลดมาวางใน Folder User

AppData ▸ Roaming ▸ Sublime Text 3 ▸ Packages ▸ User	
Name	Date modified
 yii-activeCheckBox.sublime-snippet	3/2/2012 2:26 PM
 yii-activeCheckBoxList.sublime-snippet	3/2/2012 2:26 PM
 yii-activeDataProvider.sublime-snippet	3/2/2012 2:26 PM
 yii-activeDropDownList.sublime-snippet	3/2/2012 2:26 PM
 yii-activeFileField.sublime-snippet	3/2/2012 2:26 PM
 yii-activeHiddenField.sublime-snippet	3/2/2012 2:26 PM
 yii-activeId.sublime-snippet	3/2/2012 2:26 PM
 yii-activeInputField.sublime-snippet	3/2/2012 2:26 PM
 yii-activeLabel.sublime-snippet	3/2/2012 2:26 PM

สำหรับการเรียกใช้งานให้ดูคำสั่งได้ที่เมนู Tools->Snippets... ก็จะปรากฏคำสั่งต่างๆ ขึ้นมา



Snippet:	
Snippet: Yii CHtml::asset	hass,tab
Snippet: Yii CHtml::button	hb,tab
Snippet: Yii CHtml::cdata	hc,tab
Snippet: Yii CHtml::css	hcss,tab
Snippet: Yii CHtml::encode	hec,tab
Snippet: Yii CHtml::error	herr,tab
Snippet: Yii CHtml::form	hform,tab
Snippet: Yii CHtml::image	himg,tab
Snippet: Yii CHtml::label	hlabel,tab
Snippet: Yii CHtml::link	hlink,tab
Snippet: Yii CHtml::mailto	hmt,tab
Snippet: Yii CHtml::script	hjs,tab
Snippet: Yii CHtml::tag	ht,tab

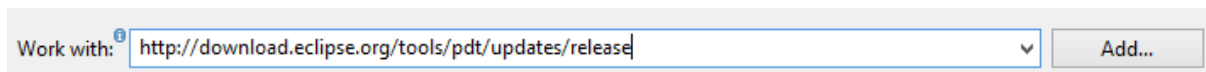
สำหรับการสร้าง Project ใน Sublime Text3 นั้นง่ายๆ เพียงแค่เลือก File->Open Folder ก็
สามารถเปิด Project ได้ทันที

การติดตั้ง Yii Eclipse Plugin

Eclipse ก็เป็น IDE ตัวหนึ่งที่เราสามารถนำมาเขียน PHP ได้ แต่ต้องติดตั้ง Eclipse PDT Plugin เข้าไปด้วย โดยสามารถดาวน์โหลดได้จากเว็บไซต์ www.eclipse.org

สำหรับการติดตั้ง Eclipse PDT Plugin นั้นเมื่อติดตั้ง Eclipse เสร็จเรียบร้อยแล้ว เปิด Eclipse แล้วไปที่เมนู Help->Install New Software จากนั้นใส่ URL

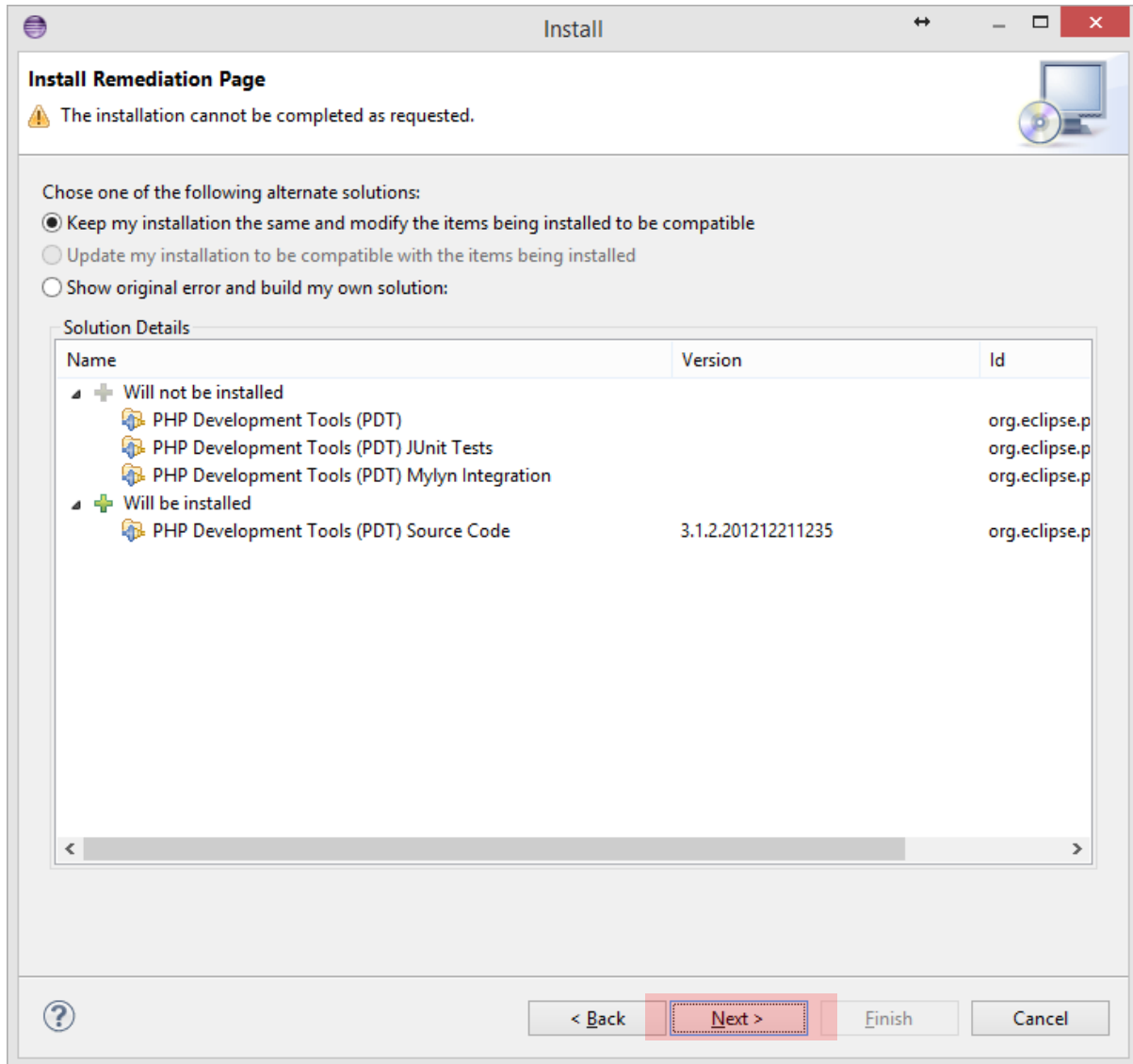
<http://download.eclipse.org/tools/pdt/updates/release> แล้วกดปุ่ม Add...



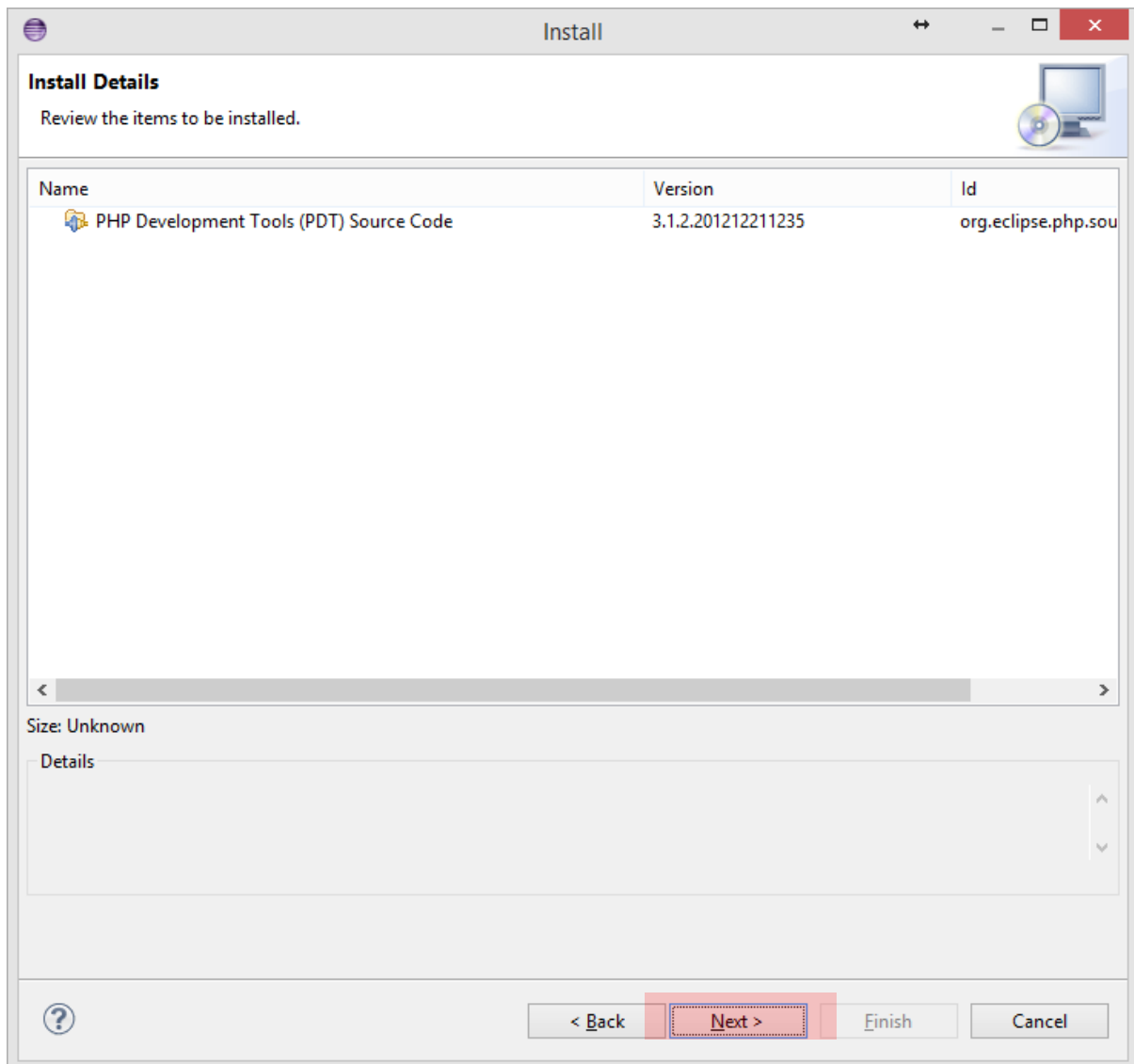
จากนั้นจะปรากฏข้อมูลดังภาพ

- ▶ ☒ PHP Development Tools
- ▶ ☒ Uncategorized

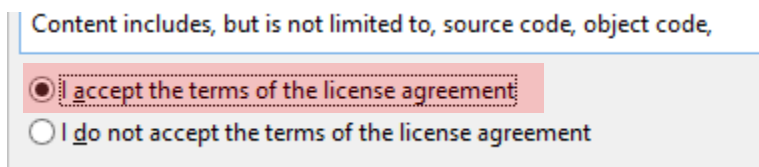
กดปุ่ม Next... ได้เลย



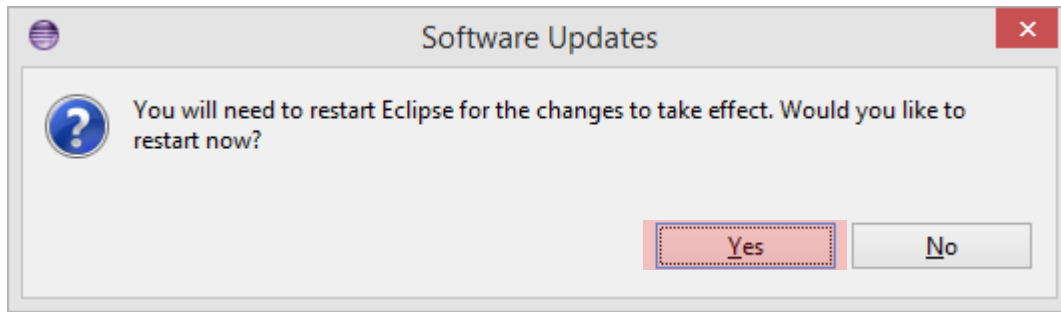
จะปรากฏรายละเอียดการติดตั้ง กดปุ่ม Next...



กดปุ่มยอมรับการใช้งาน แล้วกดปุ่ม Next...

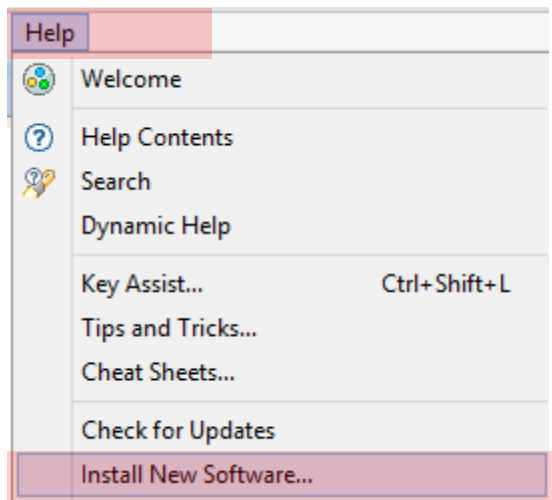


แล้วกดปุ่ม Finish แล้วกดปุ่ม Yes เพื่อ Restart Eclipse

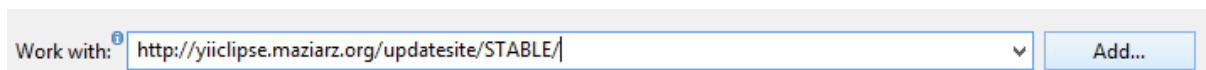


สำหรับ Eclipse IDE นั้นก็มี PlugIn สำหรับ Yii Framework เช่นกันโดยมีเว็บไซต์ <http://yiiclipse.maziarz.org/> เป็นเว็บไซต์ผู้พัฒนา

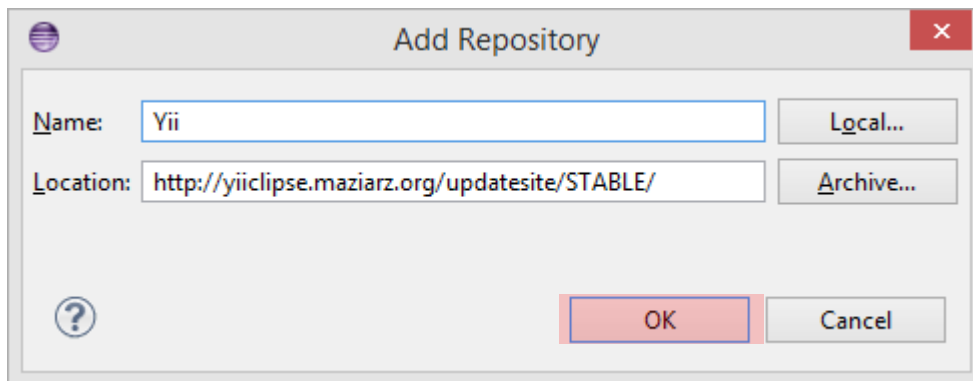
สำหรับการติดตั้ง Yii Eclipse PlugIn นั้นไปที่เมนู Help->Install New Software...



วาง Link <http://yiiclipse.maziarz.org/updatesite/STABLE/> ที่ช่อง Work with: จากนั้นกดปุ่ม Add...



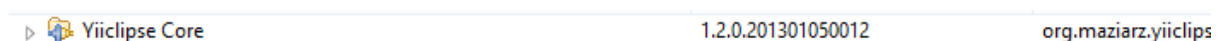
จากนั้นตั้งชื่อ Yii (ที่จริงชื่ออะไรก็ได้ครับ) แล้วกดปุ่ม OK



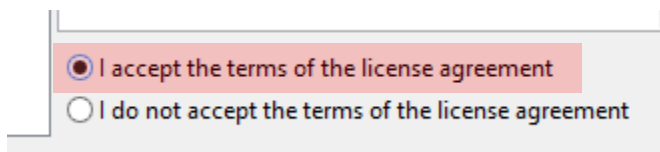
จะปรากฏข้อมูลดังภาพ

▶ ☒  Yiiclipse

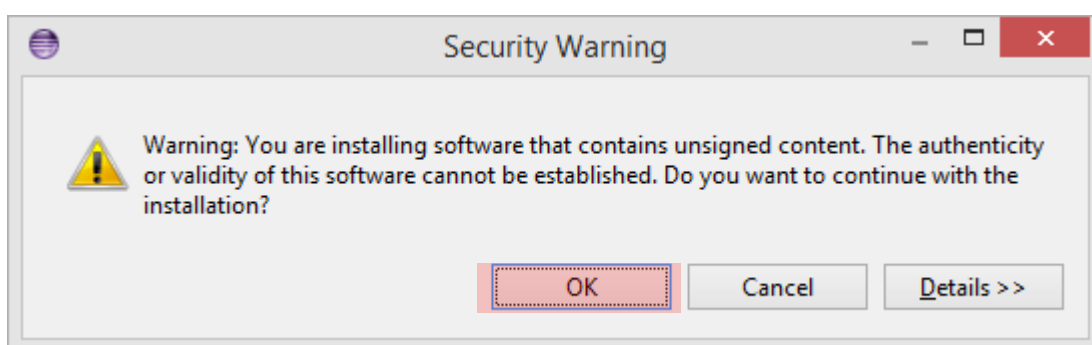
จากนั้นกดปุ่ม Next... จะปรากฏรายละเอียด



กดเลือกยอมรับข้อตกลงแล้วกดปุ่ม Next...



จากนั้นจะปรากฏข้อความเตือนกด OK

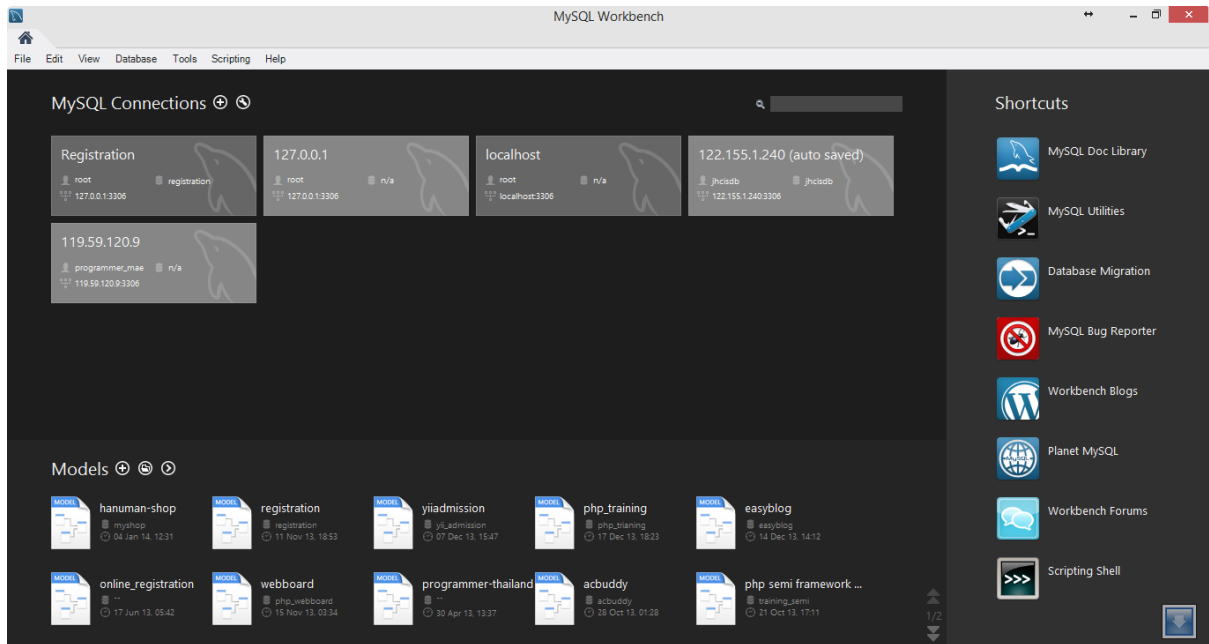


แล้วก็ทำการ Restart Eclipse จากนั้นก็สามารถใช้ Eclipse กับ PHP และ Yii Framework ได้แล้วครับ

การออกแบบ ER Diagram และการ Forward Engineering

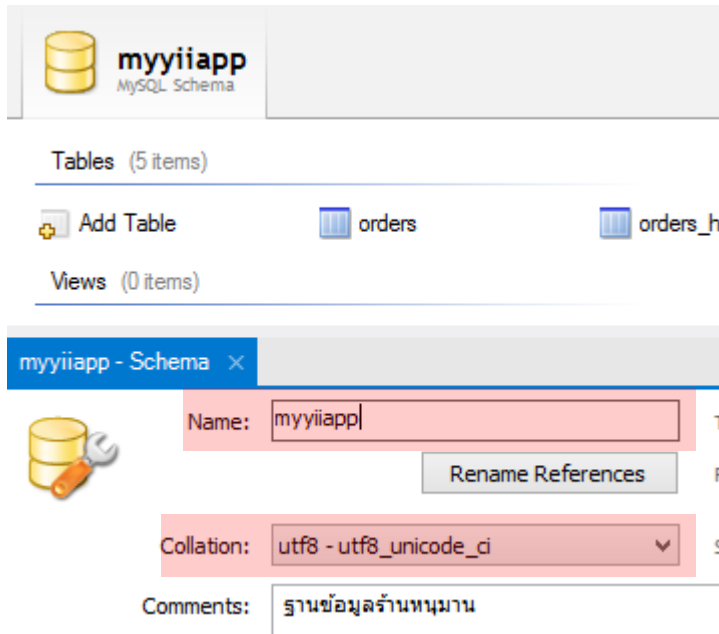
สำหรับการออกแบบ ER Diagram หรือ Entity Relationship Diagram นั้นเป็นการออกแบบ Entity หรือตารางในการเก็บข้อมูลในฐานข้อมูล MySQL นั้นเองโดยจะใช้โปรแกรมของ MySQL ที่มีชื่อว่า MySQL Workbench ซึ่งสามารถดาวน์โหลดได้ที่

<http://www.mysql.com/products/workbench/> โดยสามารถดาวน์โหลดและติดตั้งได้ฟรี



การออกแบบ ER Diagram

สร้าง Project ใหม่ ไปที่เมนู File->New Model แล้วเปลี่ยนชื่อ Schema หรือชื่อฐานข้อมูลตามที่ต้องการ



แล้วคลิกที่ Add Diagram



แล้วใช้เครื่องมือ Create New Table เพื่อสร้างตารางหรือ Entity ใหม่



โดยกำหนดให้มีลักษณะของตารางดังนี้ คือ

- Field แรกมีชื่อว่า id เป็น Primary Key และเป็น Auto Increment
- ใส่รายละเอียดทั้งหมด เช่น ชนิด ความยาว และใส่ความเห็น (Comment) ของแต่ละ Field ด้วย *** สำคัญมาก ตอนที่ใช้ Gii Generator ***

users - Table x

Table Name: Schema: **myyiiapp**

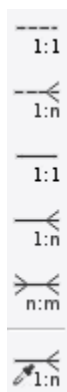
Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
firstname	VARCHAR(100)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
lastname	VARCHAR(100)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
username	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
password	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
address	TEXT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
province	ENUM('กรุงเทพมหานคร...', 'นนทบุรี', 'ปทุมธานี', 'นนทบุรี')	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
zipcode	VARCHAR(5)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
created	DATETIME	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
user_type	ENUM('admin','user')	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'user'

Collation: Table Default

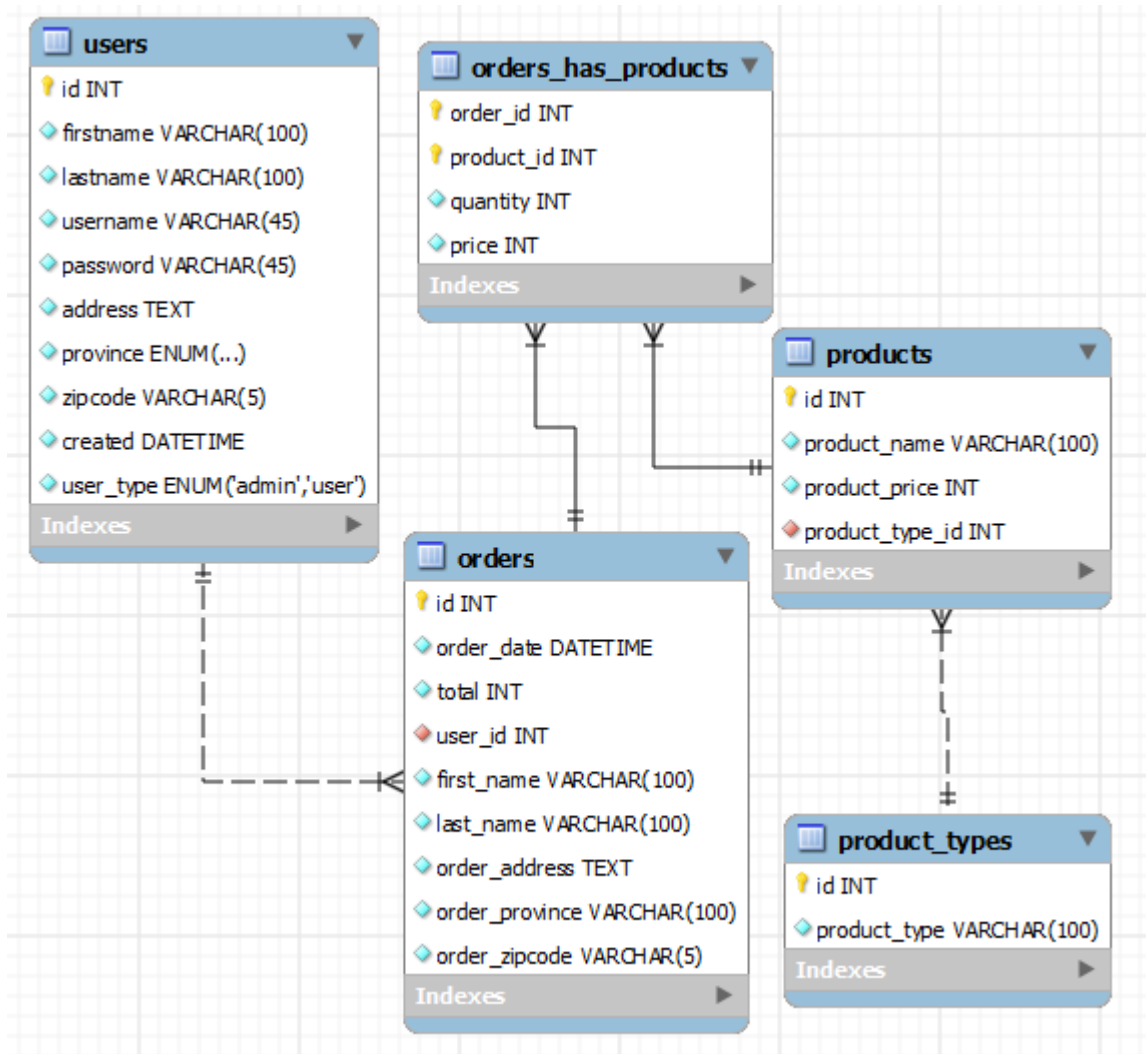
Comments: ชื่อ

Columns Indexes Foreign Keys Triggers Partitioning Options Inserts Privileges

ใช้เครื่องมือความสัมพันธ์ (Relation) เพื่อสร้างความสัมพันธ์แบบต่างๆ

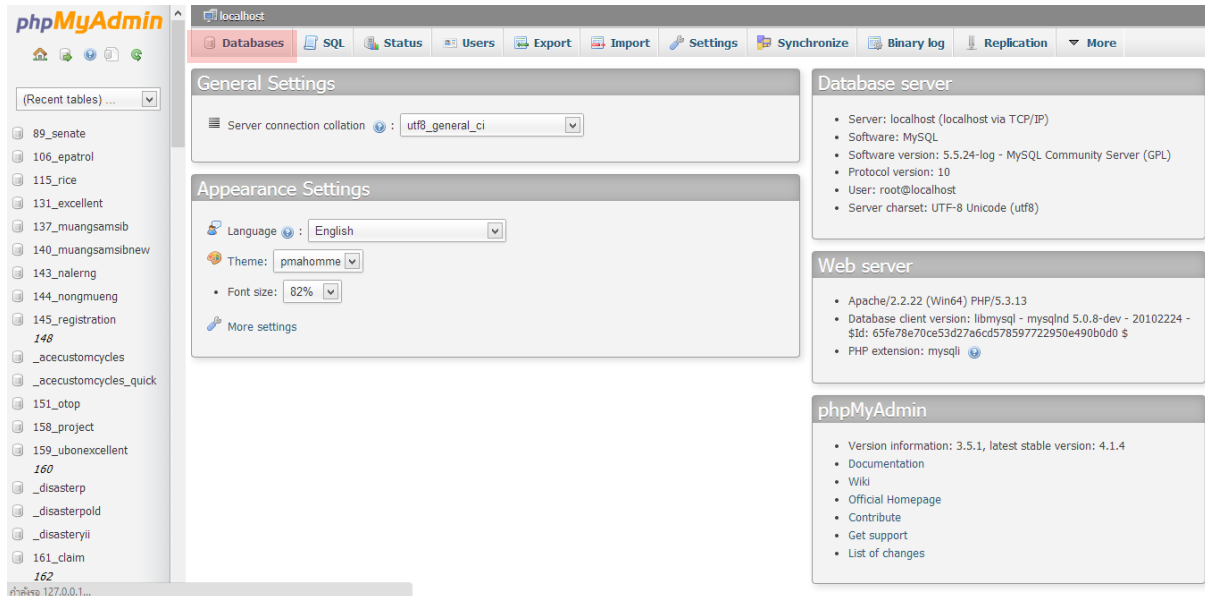


เมื่อได้ Diagram ที่ต้องการแล้วจะมีลักษณะดังนี้



การสร้างฐานข้อมูล

ก่อนการ Forward Engineering นั้นให้ทำการสร้างฐานข้อมูลก่อน โดยเข้า
<http://127.0.0.1/phpmyadmin> หรือ <http://localhost/phpmyadmin>

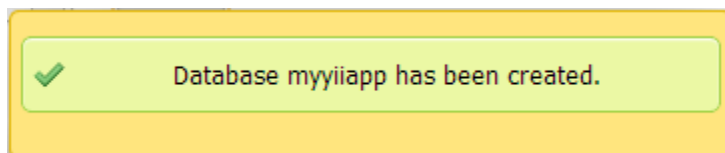


จากนั้นคลิกที่ Database แล้วใส่ชื่อฐานข้อมูลตามต้องการ ในที่นี้คือ myyiiapp แล้วเลือกการเข้ารหัสตัวอักษรให้เป็น utf8_unicode_ci แล้วกด Create

Databases

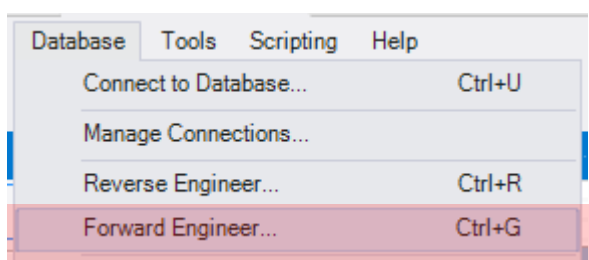


จะปรากฏกล่องข้อความสีเขียวบอกว่าได้สร้างฐานข้อมูล myyiiapp เรียบร้อยแล้ว



การ Forward Engineering

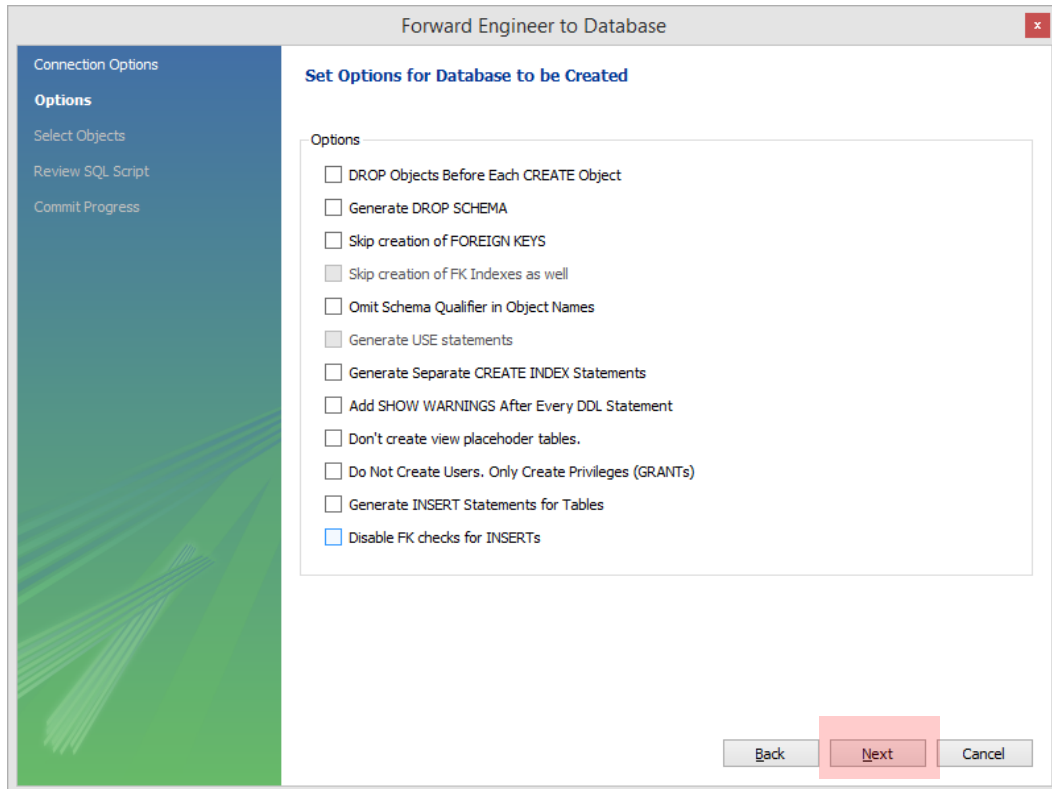
การ Forward Engineering เป็นการนำโครงสร้างจาก ER Diagram ไป Create ในฐานข้อมูลจริง โดยไปที่เมนู Database->Forward Engineer



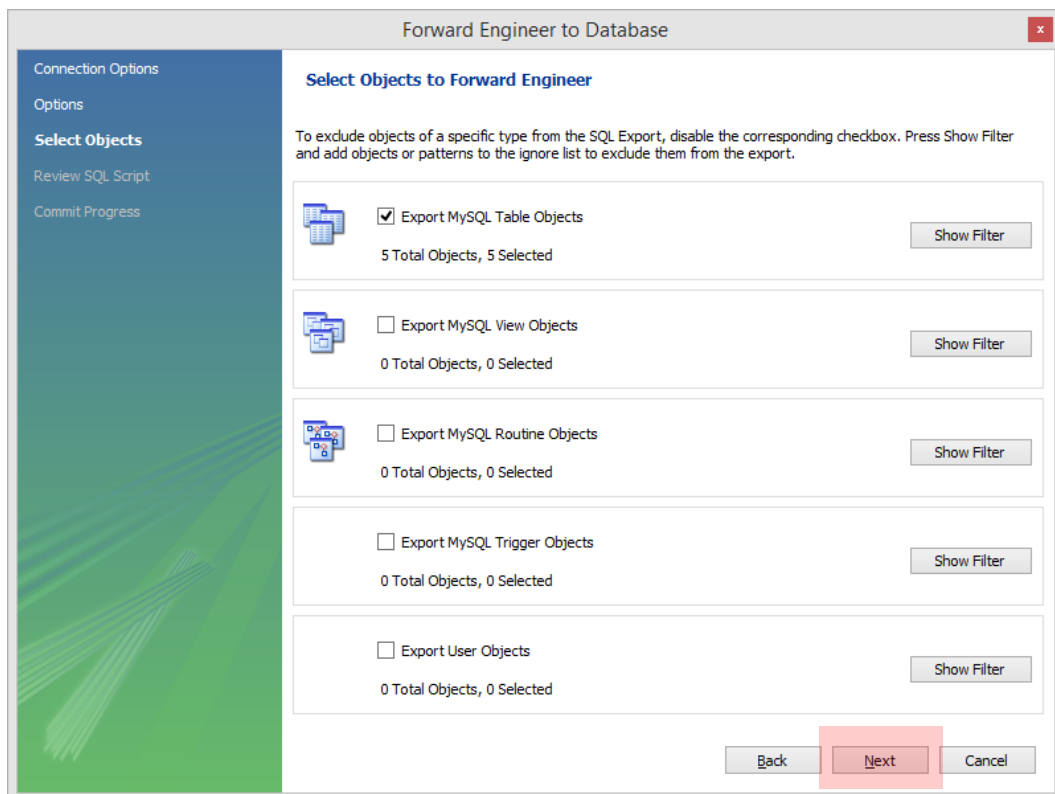
จะปรากฏขั้นตอนของการเชื่อมต่อ ให้คลิก Next ไปได้เลย

The screenshot shows a window titled "Forward Engineer to Database" with a sidebar on the left containing "Connection Options", "Options", "Select Objects", "Review SQL Script", and "Commit Progress". The main area is titled "Set Parameters for Connecting to a DBMS". It includes a "Stored Connection" dropdown set to "127.0.0.1" with a tooltip "Select from saved connection settings", and a "Connection Method" dropdown set to "Standard (TCP/IP)" with a tooltip "Method to use to connect to the RDBMS". Below these are three tabs: "Parameters", "SSL", and "Advanced". The "Parameters" tab is active, showing fields for "Hostname" (127.0.0.1), "Port" (3306), "Username" (root), "Password" (with "Store in Vault ..." and "Clear" buttons), and "Default Schema". To the right of these fields are descriptive tooltips. At the bottom right are "Back", "Next", and "Cancel" buttons, with the "Next" button highlighted by a red rectangle.

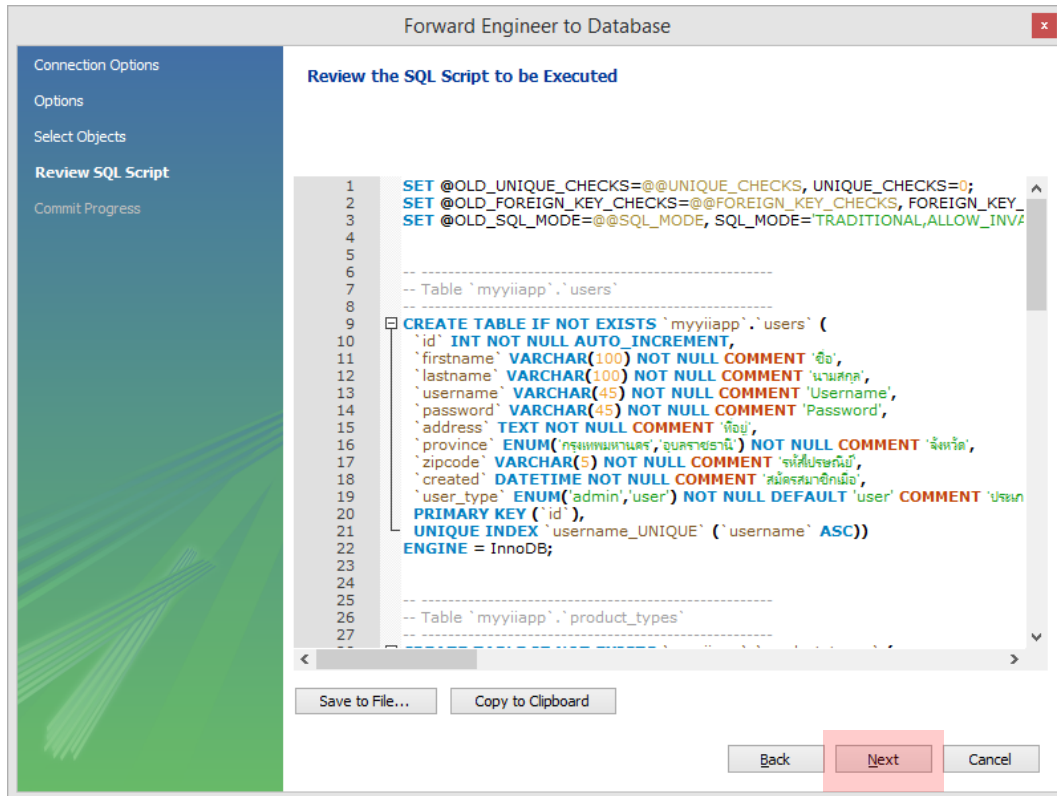
จะปรากฏ Option เสริม เพื่อเลือกที่จะลบตารางเก่าออกจากฐานข้อมูลหรือเปล่า ขั้นตอนนี้คลิก Next ได้เลย



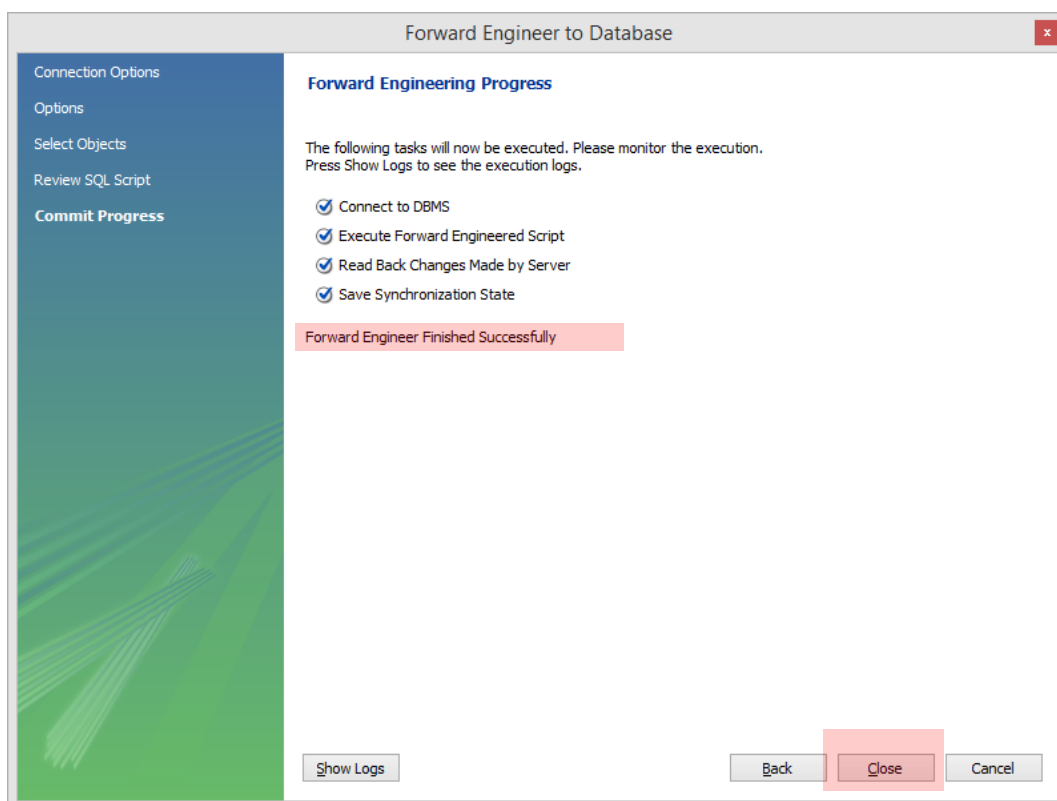
จากนั้นจะปรากฏการกรอง ว่าจะ Forward อะไรมไปบ้างในที่นี้ใช้หมดทุกอย่างให้คลิก Next ไปได้เลย



จะปรากฏรายการคำสั่ง SQL ซึ่งสามารถ Copy ไว้ได้ ในที่นี้กด Next ไปได้เลย



จะปรากฏผลลัพธ์ของการ Forward Engineering เรียบร้อยแล้ว โดยจะสังเกตเห็นคำว่า Forward Engineer Finished Successfully



ก็เป็นอันเสร็จเรียบร้อยสำหรับการสร้างฐานข้อมูลและการ Forward Engineering

พจนานุกรมข้อมูล (Data Dictionary)

orders

Table comments: orders

Column	Type	Null	Default	Comments
id	int(11)	No		
order_date	datetime	No		วันที่สั่งซื้อ
total	int(11)	No		รวมเป็นเงิน
user_id	int(11)	No		ผู้สั่งซื้อ
first_name	varchar(100)	No		ชื่อผู้รับ
last_name	varchar(100)	No		นามสกุลผู้รับ
order_address	text	No		ที่อยู่ผู้รับ
order_province	varchar(100)	No		จังหวัดผู้รับ
order_zipcode	varchar(5)	No		รหัสไปรษณีย์ผู้รับ

orders_has_products

Table comments: orders_has_products

Column	Type	Null	Default	Comments
order_id	int(11)	No		ใบสั่งซื้อ
product_id	int(11)	No		สินค้า
quantity	int(11)	No		จำนวน
price	int(11)	No		ราคา

product_types

Table comments: product_types

Column	Type	Null	Default	Comments
id	int(11)	No		

product_type	varchar(100)	No		ประเภทสินค้า
--------------	--------------	----	--	--------------

products

Table comments: products

Column	Type	Null	Default	Comments
id	int(11)	No		
product_name	varchar(100)	No		ชื่อสินค้า
product_price	int(11)	No		ราคาสินค้า
product_type_id	int(11)	No		ประเภทสินค้า

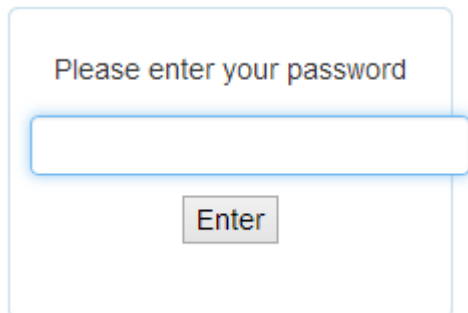
users

Table comments: users

Column	Type	Null	Default	Comments
id	int(11)	No		
firstname	varchar(100)	No		ชื่อ
lastname	varchar(100)	No		นามสกุล
username	varchar(45)	No		Username
password	varchar(45)	No		Password
address	text	No		ที่อยู่
province	enum('กรุงเทพมหานคร', 'อุบลราชธานี')	No		จังหวัด
zipcode	varchar(5)	No		รหัสไปรษณีย์
created	datetime	No		สมัครสมาชิกเมื่อ
user_type	enum('admin', 'user')	No	user	ประเภทผู้ใช้

การสร้าง Attribute Label ให้เป็นภาษาไทยโดยอัตโนมัติ

หลังจากที่มีการออกแบบ ER Diagram และทำการ Forward Engineering เรียบร้อยแล้ว ขั้นตอนถัดมาจะเป็นเรื่องของ การ Generate Code ด้วย Gii Generator โดยเข้า URL `index.php?r=gii` จะปรากฏช่องสำหรับกรอก Password ให้ทำการกรอก Password ตามที่ได้ตั้งไว้ใน `protected/config/main.php`

A screenshot of a web form with a light blue border. Inside, the text "Please enter your password" is displayed in a dark blue font. Below the text is a white rectangular input field with a blue border. Underneath the input field is a small grey button with the word "Enter" in black text.

ไปที่เมนู Model Generator

Welcome to Yii Code Generator!

You may use the following generators to quickly build up your Yii application:

- [Controller Generator](#)
- [Crud Generator](#)
- [Form Generator](#)
- [Model Generator](#)
- [Module Generator](#)
- [Bootstrap Generator](#)

แล้วใส่เครื่องหมาย * เพื่อสร้างทุกๆ Model

Generators
Controller Generator
Crud Generator
Form Generator
Model Generator
Module Generator
Bootstrap Generator

Model Generator

This generator generates a model class for the specified database table.

Fields with * are required. Click on the highlighted fields to edit them.

Database Connection *

db

Table Prefix

[empty]

Table Name *

*

Base Class *

CActiveRecord

Model Path *

application.models

Build Relations

☒

Use Column Comments as Attribute Labels

☒

Code Template *

default (D:\wamp\yii\framework\yii\generators\model\templates\default)

Preview

This refers to the table name that a new model class should be generated for (e.g. `tbl_user`). It can contain schema name, if needed (e.g. `public.tbl_post`). You may also enter * (or `schemaName.*` for a particular DB schema) to generate a model class for EVERY table.

จากนั้นกดปุ่ม Preview และ Generate ต่อไป โดยต้องให้แน่ใจว่า CheckBox Use Column Comments as Attribute Labels ถูกเลือกอยู่

Preview
Generate

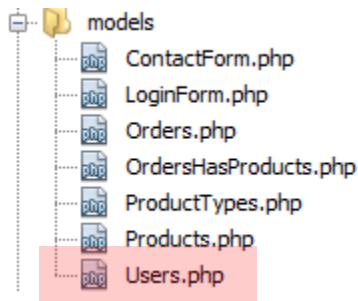
Code File	Generate <input checked="" type="checkbox"/>
models\Orders.php	new <input checked="" type="checkbox"/>
models\OrdersHasProducts.php	new <input checked="" type="checkbox"/>
models\ProductTypes.php	new <input checked="" type="checkbox"/>
models\Products.php	new <input checked="" type="checkbox"/>
models\Users.php	new <input checked="" type="checkbox"/>

ก็จะปรากฏข้อความแจ้งเตือนว่าได้สร้างเสร็จเรียบร้อยแล้ว

The code has been generated successfully.

```
Generating code using template "D:\wamp\yii\framework\yii\generators\model\templates\default"...
generated models\Orders.php
generated models\OrdersHasProducts.php
generated models\ProductTypes.php
generated models\Products.php
generated models\Users.php
done!
```

เปิดดู Model เพื่อทดสอบ



เลื่อนดูในส่วนของ Attribute Label จะเห็นว่ามีข้อความภาษาไทยเกิดขึ้นแล้ว โดยที่เราไม่ต้องมาแก้ไขอีก หรือหากต้องการแก้ไขก็สามารถแก้ไขได้เลย

```
public function attributeLabels()
{
    return array(
        'id' => 'ID',
        'firstname' => 'ชื่อ',
        'lastname' => 'นามสกุล',
        'username' => 'Username',
        'password' => 'Password',
        'address' => 'ที่อยู่',
        'province' => 'จังหวัด',
        'zipcode' => 'รหัสไปรษณีย์',
        'created' => 'สมัครสมาชิกเมื่อ',
        'user_type' => 'ประเภทผู้ใช้',
    );
}
```

การกำหนด Yii Message ให้เป็นภาษาไทย

ในการพัฒนา Application นั้นคงหลีกเลี่ยงไม่พ้นเรื่องของ Localization หรือการใช้งานภาษาในแต่ละท้องถิ่นนั่นเอง ส่วนหนึ่ง Yii Framework ก็ได้รวบรวมและมีการแปลไว้แล้วหลายภาษาหนึ่งในนั้นก็คือภาษาไทยของเรานั่นเอง

การตั้งค่าภาษานั้นทำได้ง่ายตายมากเพียงเปิดไฟล์ protected/config/main.php แล้วกำหนดค่าดังนี้

ใส่ค่า 'language'=>'th',

```
8     return array(  
9         'basePath'=>dirname(__FILE__).DIRECTORY_SEPARATOR.'..',  
10        'name'=>'My Web Application',  
11        'language'=>'th',
```

เพียงเท่านี้เวลาที่มีการแสดงผลเช่น การ validate ก็จะเป็นภาษาไทยเรียบร้อยแล้ว ทดลองโดยไปที่เมนู Contact แล้วกดปุ่ม Submit ก็จะปรากฏ Error ตัวหนังสือสีแดงที่เป็นภาษาไทยเรียบร้อยแล้ว

Contact Us

If you have business inquiries or other questions, please fill out the following form to contact us. Thank you.

*Fields with * are required.*

โปรดแก้ไขข้อผิดพลาดทางด้านล่าง:

- Name ไม่ควรเป็นค่าว่าง
- Email ไม่ควรเป็นค่าว่าง
- Subject ไม่ควรเป็นค่าว่าง
- Body ไม่ควรเป็นค่าว่าง
- รหัสป้องกันไม่ถูกต้อง

Name *

Name ไม่ควรเป็นค่าว่าง

Email *

Email ไม่ควรเป็นค่าว่าง

Subject *

Subject ไม่ควรเป็นค่าว่าง



การใช้งานรูปภาพ

บ่อยครั้งใน Application เราได้มีการใช้งานรูปภาพดังนั้นเราควรที่จะมาเรียนรู้ในเรื่องของการใช้งานรูปภาพกันนะครับ ซึ่งรูปภาพนั้นเป็นส่วนหนึ่งของ Helper CHtml::image() โดยเป็นตัวช่วยที่จะ Generate ออกมาให้อยู่ในรูป HTML ที่เป็น Tag img โดยสามารถดูเพิ่มเติมได้ที่ <http://www.yiiframework.com/doc/api/1.1/CHtml#image-detail>

สำหรับการใช้งานนั้นจะมี Class Reference ดังนี้ public static string image(string \$src, string \$alt="", array \$htmlOptions=array ()) โดยมี \$src คือ ที่อยู่ของรูปภาพ \$alt คือ ข้อความเมื่อนำ Mouse ไปวาง หรือเมื่อไม่มีรูปภาพก็จะแสดงข้อความนี้ \$htmlOptions คือ การกำหนด Attribute อื่นๆ เช่น class, width, height เป็นต้น

ใน Root ของเว็บไซต์เรามี Folder เก็บรูปภาพชื่อ myyiiapp/img/ และมีไฟล์รูปภาพชื่อ image.png ใน Yii Framework เราสามารถอ้างอิงไปยัง Application Base Path โดยใช้คำสั่งดังนี้
Yii::app()->baseUrl

ตัวอย่างการใช้งาน

การใช้งานพื้นฐาน

```
echo CHtml::image(Yii::app()->baseUrl.'/img/image.png');
```

รูปภาพกับ ALT

หากมีการใส่ Attribute alt ก็จะกำหนดดังนี้

```
echo CHtml::image(Yii::app()->baseUrl.'/img/image.png','ทดสอบ ALT');
```

จะได้ผลลัพธ์ดังนี้

```

```

รูปภาพกับขนาด

ทดลองเพิ่มความกว้างและความสูงของรูปภาพ

```
echo CHtml::image(Yii::app()->baseUrl.'/img/image.png','ทดสอบ ALT',array('width'=>'100px','height'=>'80px'));
```

จะได้ผลลัพธ์ดังนี้

```

```

การใช้งาน Link

อีกส่วนที่สำคัญนั้นก็คือการทำ Link โดยใน Application ค่อนข้างมีการใช้งานบ่อยซึ่ง Link นั้นก็มี Helper ที่ทำงานเฉพาะ นั่นก็คือ CHtml::link() สามารถดูเพิ่มเติมได้ที่

<http://www.yiiframework.com/doc/api/1.1/CHtml#link-detail>

สำหรับการใช้งานนั้นจะมี Class Reference ดังนี้ public static string link(string \$text, mixed \$url='#', array \$htmlOptions=array ()) โดยมี

\$text คือ ข้อความ Link

\$url คือ Link

\$htmlOptions คือ การกำหนด Attribute อื่นๆ เช่น class, title เป็นต้น

ตัวอย่างการใช้งาน

การใช้งานพื้นฐาน

```
echo CHtml::link('ข้อความ',array('controller/action'));
```

จะได้ผลลัพธ์ดังนี้

```
<a href="index.php?r=controller/action">ข้อความ</a>
```

หรือ

```
<a href="controller/action">ข้อความ</a> (กรณีเปิดการใช้งาน urlManager)
```

การสร้าง Link แบบส่งค่า

```
echo CHtml::link('ข้อความ',array('controller/action','param1'=>'value1','param2'=>'value2'));
```

จะได้ผลลัพธ์ดังนี้

```
<a href="index.php?r=controller/action&param1=value1&param2=value2">ข้อความ</a>
```

การสร้าง Link ใน Controller เดียวกัน

```
echo CHtml::link('ข้อความ',array('action'));
```

การสร้าง Link ไปที่ Module

```
echo CHtml::link('ข้อความ',array('/module/controller/action'));
```

การเปิด Link ใน Tab ใหม่

echo

```
CHtml::link('ข้อความ',array('controller/action','param1'=>'value1'),array('target'=>'_blank'));
```

การใส่ Class และกำหนด Id

```
echo CHtml::link('ข้อความ',array('controller/action','param1'=>'value1'),array('class'=>'ชื่อ  
class','id'=>'ชื่อ id'));
```

กำหนดให้ Confirm ก่อนเปิด Link

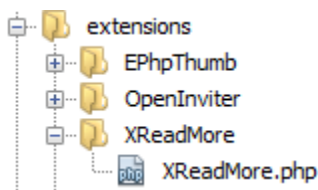
```
echo CHtml::link('ข้อความ','#',array('submit'=>array('delete','id'=>$data->ID),'confirm'=>  
แน่ใจว่าต้องการลบ?));
```

บทที่ 2 ว่าด้วยเรื่อง Extension

การตัดคำด้วย XReadmore

บ่อยครั้งเวลาที่เราเขียน Weblog นั้นการตัดคำเป็นเรื่องสำคัญมาก ในหน้า Application ที่รวม Blog ไว้ เช่นหน้าแรกซึ่งมี Blog รวมกันหลายๆ Blog หากมีการแสดงข้อมูล Blog ทั้งหน้า หน้า Application เราคงยืดเยื้อเป็นแน่ การตัดคำจึงเป็นส่วนสำคัญของทางแก้ของปัญหานี้ ซึ่งก็มีหลากหลายรูปแบบ แต่มีรูปแบบหนึ่งที่ทำให้เราสะดวกรวดเร็วในการใช้งานนั่นก็คือการตัดคำด้วย XReadmore Extension โดยสามารถดาวน์โหลดได้ที่ www.yiiframework.com/extension/xreadmore/

วิธีการติดตั้งหลังจากที่ดาวน์โหลดมาเรียบร้อยแล้วให้นำไปวางใน
protected/extensions/XReadMore/



วิธีการใช้งานนั้นให้เปิด View ที่ต้องการตัดคำแล้วเรียกใช้งานดังนี้ \$data คือ Data Model และระบุ Attribute หรือชื่อ Field ที่ต้องการให้ตัดคำในที่นี้คือ content แล้วกำหนด maxChar หรือ จำนวนคำสูงสุดที่แสดงผล หลังจากนั้นจะแสดง ...

```
$this->widget('ext.XReadMore.XReadMore', array(
    'showLink'=>false,
    'model'=>$data,
    'attribute'=>'content',
    'maxChar'=>500,
));
```

หลังจากเรียก URL จะแสดงผลดังนี้



December 17, 2013

หนังสือ Yii Framework Application Workshop #1 คลอดแล้วครับ

โดย [admin](#)

หลังจากที่ดั่งหน้าด่าเขียนหนังสือ Yii Framework Application Workshop เล่ม 1 มายาวนานตั้งแต่ วันที่ 15 เมษายน 2556 วันนี้ก็สำเร็จลงสำหรับการปรับแก้และทดสอบพร้อมที่จะให้เพื่อนๆ ชาว Programmer Thailand ได้นำไปประยุกต์ใช้ในงานของตัวเองแล้วครับ

วันแล้ววันเล่าที่ผมนั่งเขียน...

Tags: [yii](#), [yii framework](#), [book](#)

[Permalink](#) | [Comments \(1\)](#) | แก้ไขล่าสุดเมื่อ December 17, 2013

[อ่านต่อ...](#)

การติดตั้ง Yii Booster Extension

Bootstrap เป็น CSS Framework ที่พัฒนาโดย Twitter จากนั้นมีผู้นำมาพัฒนาต่อเป็น Extension ให้สามารถใช้ได้กับ Yii Framework โดยรวบรวมทำเป็น Widget ไว้เรียบร้อยแล้ว แล้วตั้งชื่อใหม่ว่า Yii Booster โดยสามารถดาวน์โหลดได้ที่เว็บไซต์ <http://yiibooster.clevertech.biz/> หรือ <https://github.com/clevertech/yiibooster> ก็ได้

ความสามารถที่โดดเด่นของ Bootstrap นั้นก็คือเรื่องของ CSS Framework โดยจะช่วยให้เราลดเวลาของการออกแบบ HTML และ CSS ลงไปโดยสิ้นเชิง ซึ่ง Bootstrap ได้รวบรวมและกำหนดรูปแบบต่างๆ ไว้ให้เราพร้อมใช้งานเรียบร้อยแล้ว โดยเพียงเรียกใช้งาน Class ของ CSS เช่น หากต้องการทำปุ่มก็เพียงเรียก `class="btn btn-primary"` ก็จะได้ปุ่มที่สวยงามแล้ว



Introducing YiiBooster

By Yii developers, for Yii developers: YiiBooster is Yii Bootstrap on 'roids.

นำ YiiBooster ที่ดาวน์โหลดมาเก็บไว้ใน `protected/extensions/bootstrap`

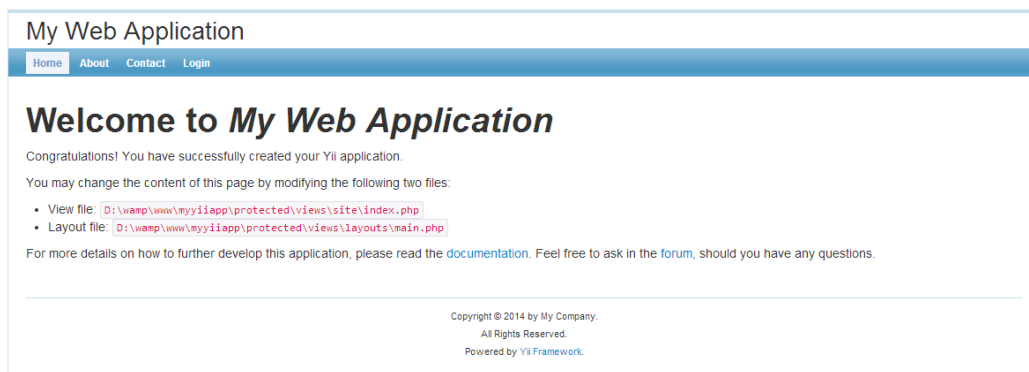
ส่วนของ modules

```
21         'modules'=>array(  
22             // uncomment the following to enable the Gii to  
23  
24             'gii'=>array(  
25                 'class'=>'system.gii.GiiModule',  
26                 'password'=>'12345',  
27                 // If removed, Gii defaults to localhost  
28                 'ipFilters'=>array('127.0.0.1','::1'),  
29                 'generatorPaths'=>array(  
30                     'bootstrap.gii',  
31                 ),  
32             ),
```

ส่วนของ preload

```
13         'preload'=>array('log','bootstrap'),
```

หลังจากนั้นทดลองเข้าสู่หน้าหลักของ Project จะเห็นว่าเว็บไซต์จะเป็นลักษณะ Responsive เรียบร้อยแล้ว



ทดลองเข้าสู่ระบบ Gii โดยพิมพ์ URL `index.php?r=gii` แล้วเข้าสู่ระบบโดยใช้รหัสผ่านที่ได้กำหนดในส่วนของ modules ในไฟล์ `protected/config/main.php` นั่นคือ 12345 จะปรากฏหน้าสำหรับ Generate Code ดังนี้



Welcome to Yii Code Generator!

You may use the following generators to quickly build up your Yii application:

- [Controller Generator](#)
- [Crud Generator](#)
- [Form Generator](#)
- [Model Generator](#)
- [Module Generator](#)
- [Bootstrap Generator](#)

โดยจะเห็นว่ามียกมา Bootstrap Generator ขึ้นมา แสดงว่าตั้งค่าถูกต้องเรียบร้อยแล้ว ก็พร้อมสำหรับการสร้าง Application โดยใช้ Bootstrap เป็น CSS Framework เรียบร้อยแล้ว

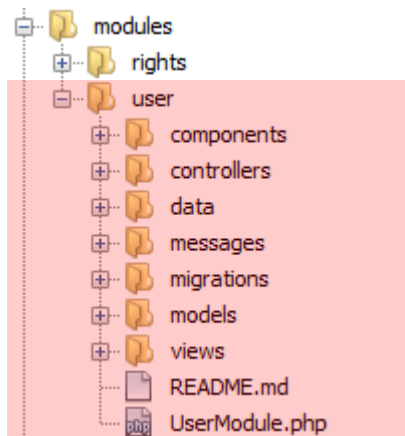
การติดตั้ง User Extension และ Rights Extension

การติดตั้ง User Extension

Yii User Extension เป็นระบบบริหารจัดการผู้ใช้งาน โดยมีระบบสมัครสมาชิก ระบบลิ้มรสผ่าน ระบบที่เกี่ยวข้องกับการจัดการสมาชิกซึ่งทำให้เรามีความสะดวกในการจัดการสมาชิกนั่นเอง

โดยสามารถดาวน์โหลดได้ที่เว็บไซต์ <http://www.yiiframework.com/extension/yii-user/>

หลังจากดาวน์โหลดเสร็จแล้วนำไปวางใน protected/modules/user



จากนั้นแก้ไขไฟล์ protected/config/main.php

ในส่วนของ import

```
'import'=>array(
    'application.models.*',
    'application.components.*',
    'application.modules.user.models.*',
    'application.modules.user.components.*',
    'application.modules.rights.*',
    'application.modules.rights.components.*',
),
```

ในส่วนของ modules

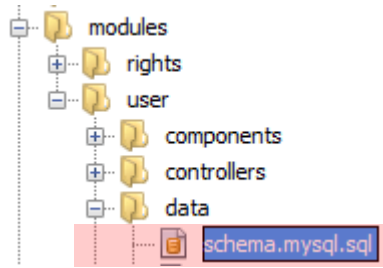
```
'modules'=>array(
    // uncomment the following to enable the Gii tool

    'gii'=>array(
        'class'=>'system.gii.GiiModule',
        'password'=>'12345',
        // If removed, Gii defaults to localhost only.
        'ipFilters'=>array('127.0.0.1','::1'),
        'generatorPaths'=>array(
            'bootstrap.gii',
        ),
    ),
    'user'=>array(
        'tableUsers' => 'users',
        'tableProfiles' => 'profiles',
        'tableProfileFields' => 'profiles_fields',
    ),
    'rights'=>array(
        'install'=>false,
    ),
),
```

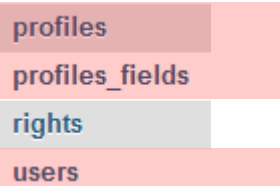
ในส่วนของ components

```
'user'=>array(
    'class'=>'RWebUser',
    // enable cookie-based authentication
    'allowAutoLogin'=>true,
    'loginUrl'=>array('/user/login'),
),
'authManager'=>array(
    'class'=>'RDbAuthManager',
    'connectionID'=>'db',
    'defaultRoles'=>array('Authenticated', 'Guest'),
),
```

ทำการ Dump ตารางลงในฐานข้อมูลโดยใช้ phpMyAdmin Import จากไฟล์
protected/modules/user/data/schema.mysql.sql



จะได้ตาราง 3 ตารางคือ profiles, profiles_fields และ users



สำหรับผู้ดูแลระบบจะได้ User และ Password คือ admin : admin และสำหรับผู้ทั่วไปจะได้ User และ Password คือ demo : demo

จากนั้นสร้างเมนูสำหรับเข้าสู่ระบบและสมัครสมาชิก และเมนูต่างๆ ที่เกี่ยวข้องกับ User Extension โดยเข้าไปแก้ไขที่ไฟล์ protected/views/layouts/main.php แล้วเพิ่ม Code ดังนี้ บริเวณ CMenu

```
array('url' => Yii::app()->getModule('user')->loginUrl, 'label' =>
Yii::app()->getModule('user')->t("Login"), 'visible' => Yii::app()->user-
>isGuest),
array('url' => Yii::app()->getModule('user')->registrationUrl, 'label' =>
Yii::app()->getModule('user')->t("Register"), 'visible' => Yii::app()-
>user->isGuest),
array('url' => Yii::app()->getModule('user')->profileUrl, 'label' =>
Yii::app()->getModule('user')->t("Profile"), 'visible' => !Yii::app()-
>user->isGuest),
array('url' => Yii::app()->getModule('user')->logoutUrl, 'label' =>
Yii::app()->getModule('user')->t("Logout") . ' (' . Yii::app()->user->name
. ')', 'visible' => !Yii::app()->user->isGuest),
```

จะได้ Code หน้าตาประมาณนี้ โดย Comment ส่วนที่เป็น Login และ Logout ของ Yii Framework ออก ให้ไปใช้ของ User Extension แทนครับ

```
<div id="mainmenu">
    <?php
    $this->widget('zii.widgets.CMenu', array(
        'items' => array(
            array('label' => 'Home', 'url' => array('/site/index')),
            array('label' => 'About', 'url' => array('/site/page', 'view' => 'about')),
            array('label' => 'Contact', 'url' => array('/site/contact')),
            //array('label' => 'Login', 'url' => array('/site/login'), 'visible' => Yii:
            //array('label' => 'Logout (' . Yii::app()->user->name . ')', 'url' => array
            array('label' => 'Rights', 'url' => array('/rights'), 'visible' => !Yii::app
            array('url' => Yii::app()->getModule('user')->loginUrl, 'label' => Yii::app
            array('url' => Yii::app()->getModule('user')->registrationUrl, 'label' => Yi
            array('url' => Yii::app()->getModule('user')->profileUrl, 'label' => Yii::a
            array('url' => Yii::app()->getModule('user')->logoutUrl, 'label' => Yii::app
        ),
    ));
    ?>
</div><!-- mainmenu -->
```

ถึงขั้นตอนนี้ก็สามารถเข้าระบบโดยใช้ Username และ Password จาก User Extension ได้แล้ว
ครับ ทดลองเข้าระบบ admin : admin และ demo : demo นะครับ

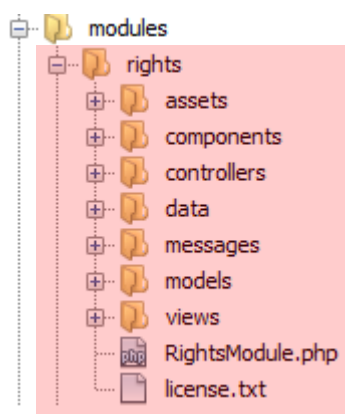
Login Register

การติดตั้ง Rights Extension

Yii Rights Extension เป็นระบบบริหารจัดการการเข้าถึงข้อมูลระดับ Action โดยกำหนดได้
ว่า Action ไหนให้ใครเข้าถึงได้

โดยสามารถดาวน์โหลดได้ที่เว็บไซต์ <http://www.yiiframework.com/extension/rights/>
หรือ <https://code.google.com/p/yii-rights/>

หลังจากดาวน์โหลดเรียบร้อยแล้วทำมาวางที่ protected/modules/right



จากนั้นเปิดไฟล์ protected/config/main.php เพื่อตั้งค่า

ในส่วนของ import

```
'import'=>array(
    'application.models.*',
    'application.components.*',

    'application.modules.user.models.*',
    'application.modules.user.components.*',
    'application.modules.rights.*',
    'application.modules.rights.components.*',
),
```

ในส่วนของ modules กำหนด install เป็น true ก่อน เมื่อติดตั้งเสร็จเรียบร้อยแล้วค่อยเปลี่ยนเป็น false ครับ

```
'modules'=>array(
    // uncomment the following to enable the Gii tool

    'gii'=>array(
        'class'=>'system.gii.GiiModule',
        'password'=>'12345',
        // If removed, Gii defaults to localhost only.
        'ipFilters'=>array('127.0.0.1','::1'),
        'generatorPaths'=>array(
            'bootstrap.gii',
        ),
    ),
    'user'=>array(
        'tableUsers' => 'users',
        'tableProfiles' => 'profiles',
        'tableProfileFields' => 'profiles_fields',
    ),
    'rights'=>array(
        'install'=>true,
    ),
),
```

การติดตั้งให้เรียก URL `index.php?r=rights/install` สำหรับการปิด urlManager หรือ `/rights/install` สำหรับการเปิด urlManger ครับ

จากนั้นกดปุ่ม Install ระบบก็จะทำการติดตั้งตารางให้ดังนี้ authassignment, authitem, authitemchild และ rights

authassignment
authitem
authitemchild
profiles
profiles_fields
rights

จากนั้นไปสร้างเมนูใน protected/views/layouts/main.php

```
array('label' => 'Rights', 'url' => array('/rights'), 'visible' =>
!Yii::app()->user->isGuest),
```

ในส่วนของ Widget CMenu ดังนี้

```
<div id="mainmenu">
    <?php
        $this->widget('zii.widgets.CMenu', array(
            'items' => array(
                array('label' => 'Home', 'url' => array('/site/index')),
                array('label' => 'About', 'url' => array('/site/page', 'view' => 'about')),
                array('label' => 'Contact', 'url' => array('/site/contact')),
                //array('label' => 'Login', 'url' => array('/site/login'), 'visible' => Yii
                //array('label' => 'Logout (' . Yii::app()->user->name . ')', 'url' => arra
                array('label' => 'Rights', 'url' => array('/rights'), 'visible' => !Yii::ap
                array('url' => Yii::app()->getModule('user')->loginUrl, 'label' => Yii::app
                array('url' => Yii::app()->getModule('user')->registrationUrl, 'label' => Y
                array('url' => Yii::app()->getModule('user')->profileUrl, 'label' => Yii::a
                array('url' => Yii::app()->getModule('user')->logoutUrl, 'label' => Yii::ap
            ),
        ));
    ?>
</div><!-- mainmenu -->
```

เมื่อถึงขั้นตอนนี้ก็สามารถใช้ Rights Extension ได้แล้วครับ โดยทดลองเข้าสู่ระบบเป็น admin จะเห็นเมนู Rights เกิดขึ้นครับ ถึงขั้นตอนนี้อย่าลืมปิด install เป็น false ใน protected/config/main.php ในส่วนของ modules นะครับ

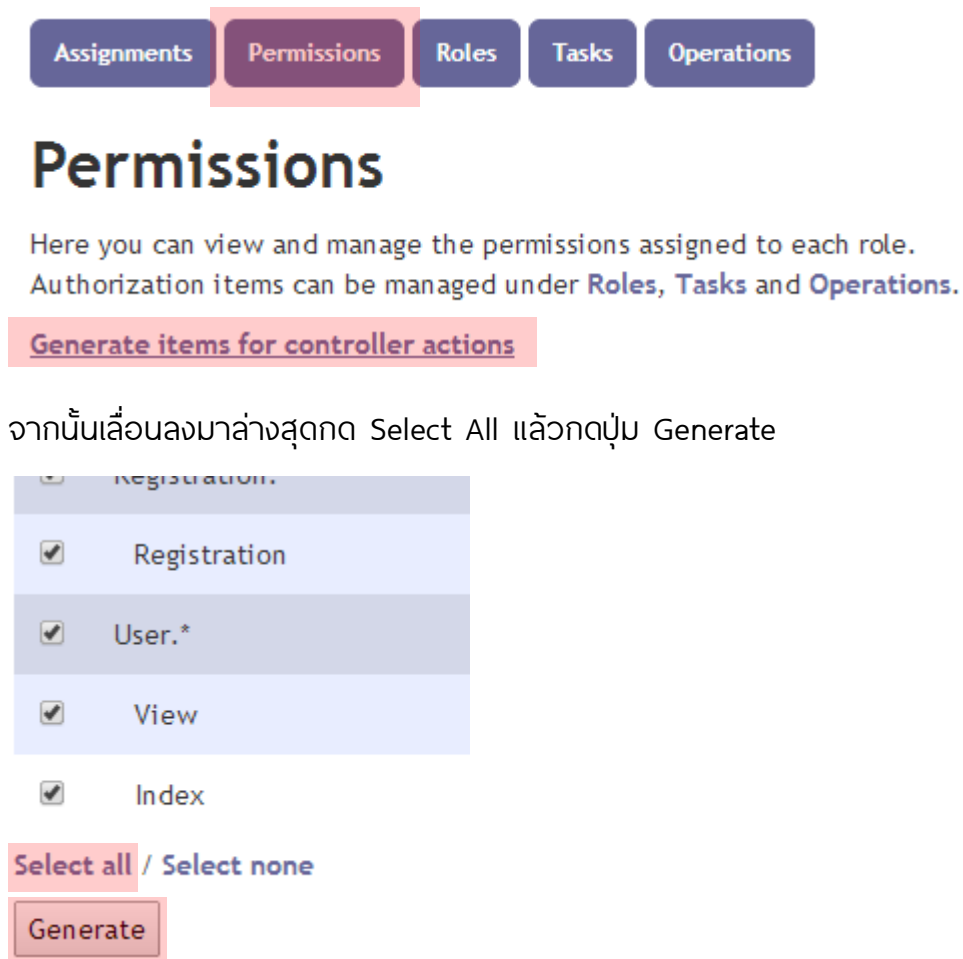


ปรับแต่งไฟล์ protected/components/Controller.php ในส่วนของการ Extends ดังนี้

```
class Controller extends RController
```


การใช้งาน User และ Rights Extension เบื้องต้น

หลังจาก Login เข้าสู่ระบบเป็น admin เรียบร้อยแล้ว แล้วเข้าเมนู Rights จากนั้น กด Permissions แล้วคลิก Link Generate items for controller actions เพื่อสร้าง Item ก่อนการกำหนดสิทธิ์



The screenshot shows the 'Permissions' tab selected in the top navigation bar. Below the tabs, the heading 'Permissions' is displayed, followed by a description: 'Here you can view and manage the permissions assigned to each role. Authorization items can be managed under Roles, Tasks and Operations.' A red button labeled 'Generate items for controller actions' is highlighted. Below this, a list of permissions is shown with checkboxes: 'Registration', 'User.*', 'View', and 'Index'. At the bottom, there are links for 'Select all' and 'Select none', and a red 'Generate' button.

ก็จะปรากฏรายการ Permissions ต่างๆ

<div>Assignments</div> <div>Permissions</div> <div>Roles</div> <div>Tasks</div> <div>Operations</div>		
<h2>Permissions</h2> <p>Here you can view and manage the permissions assigned to each role. Authorization items can be managed under Roles, Tasks and Operations.</p> <p>Generate items for controller actions</p>		
Item	Authenticated	Guest
Products.Admin	Assign	Assign
Products.Create	Assign	Assign
Products.Delete	Assign	Assign
Products.Index	Assign	Assign
Products.Update	Assign	Assign
Products.View	Assign	Assign
ProductTypes.Admin	Assign	Assign
ProductTypes.Create	Assign	Assign
ProductTypes.Delete	Assign	Assign

จากนั้นทดลอง Assign ค่าให้กับ Roles โดยกดที่ Link Assign ใน Role ที่เป็น Guest โดยให้ Guest สามารถเปิด Action Index และ View ได้

Item	Authenticated	Guest
Products.Admin	Assign	Assign
Products.Create	Assign	Assign
Products.Delete	Assign	Assign
Products.Index	Assign	Revoke
Products.Update	Assign	Assign
Products.View	Assign	Revoke

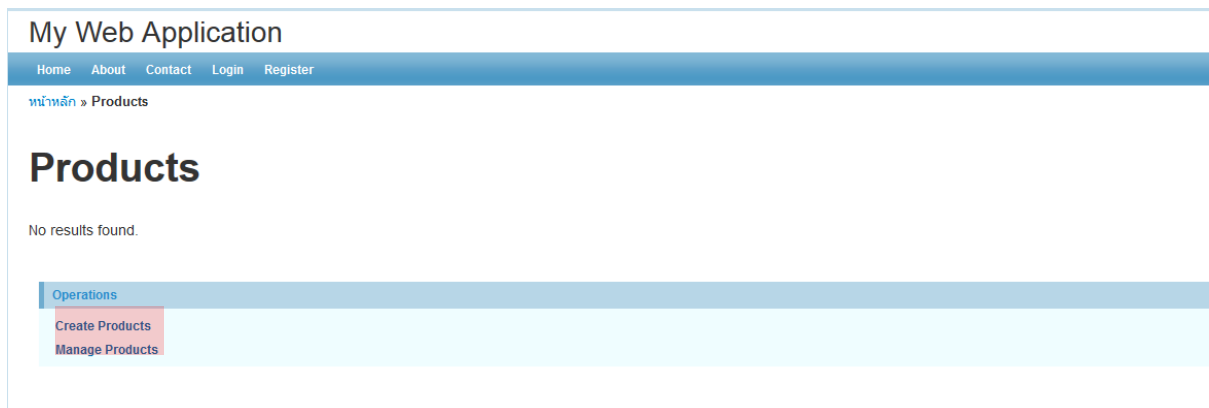
ปรับแต่งไฟล์ protected/controllers/ProductsController.php โดยปรับแต่ง Function filters() ดังนี้

```
public function filters() {
    return array(
        // 'accessControl', // perform
        'rights'
    );
}
```

หากต้องการกำหนด Action ที่ให้สามารถเข้าถึงได้เลยให้สร้าง Action ดังนี้

```
public function allowedActions() {
    return 'index';
}
```

ทดลองเรียก URL `index.php?r=products/index` หรือ `products/index` ก็จะสามารถเข้าเว็บได้เลย



ทดลองกด Create Products ระบบจะ Redirect ไปหน้าให้เข้าสู่ระบบ ทดลองใส่ User demo : demo

Login

Please fill out the following form with your login crede

*Fields with * are required.*

username or email *

password *

[Register](#) | [Lost Password?](#)

☐ Remember me next time

เมื่อเข้าสู่ระบบจะแสดง Error เกิดขึ้น (หมายเหตุ การแสดง Error ได้มีการปรับเปลี่ยน ดูเพิ่มเติมได้ที่ การสร้างและแสดง Flash Message สำหรับ Yii Booster Extension) ดังนี้

หน้าหลัก » Error

Error 403

You are not authorized to perform this action.

ทดลองเข้าสู่ระบบเป็น Admin ก็จะสามารถเพิ่มข้อมูลได้ดังนี้

[หน้าหลัก](#) » [Products](#) » [Create](#)

Create Products

Fields with * are required.

ชื่อสินค้า *

ราคาสินค้า *

ประเภทสินค้า *

Create

Operations

List Products

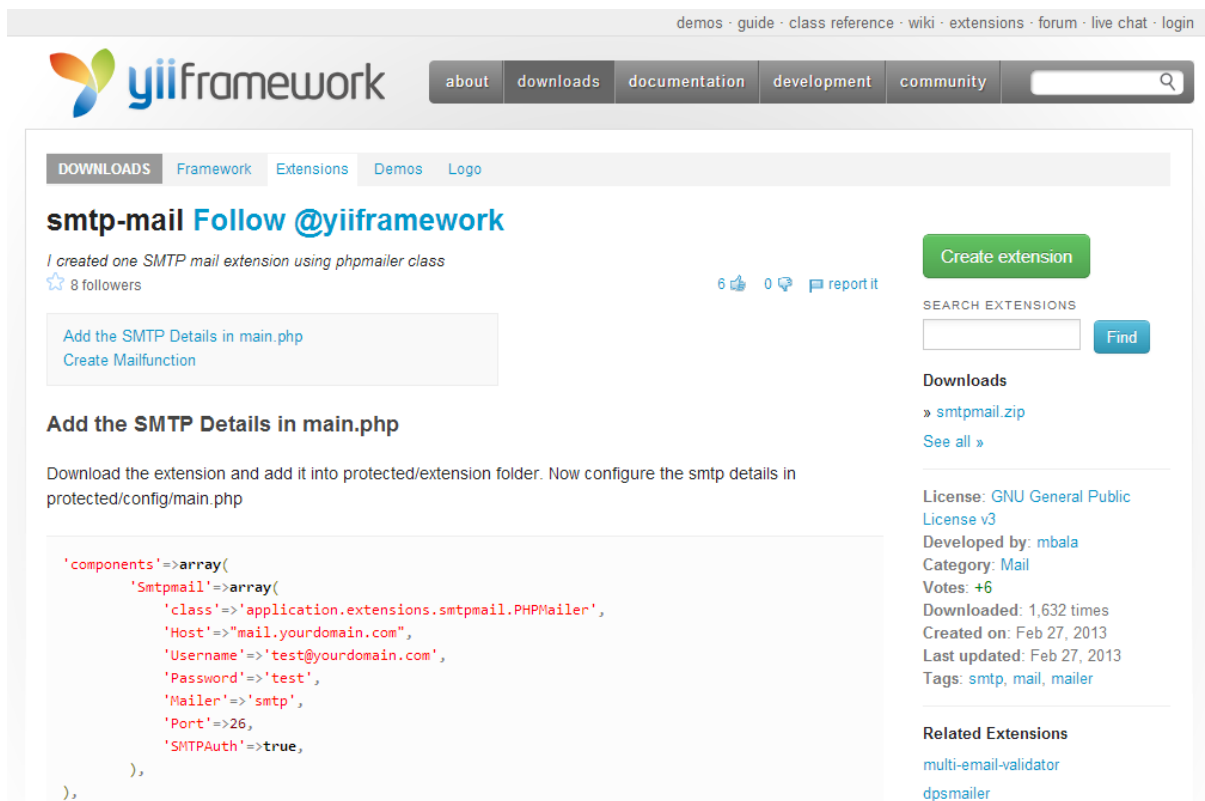
Manage Products

ถึงขั้นตอนนี้ก็สามารถใช้งาน Yii User Extension และ Rights Extension ได้เรียบร้อยแล้วครับ

การส่ง Email ด้วย SMTP Extension

ในการส่ง Email ให้ถึงผู้รับแบบชัวร์ๆ นั้นเราควรส่งด้วย SMTP Server โดยใช้ Extension ที่มีชื่อว่า smtp-mail ซึ่งสามารถดาวน์โหลดได้ที่

<http://www.yiiframework.com/extension/smtp-mail/>



The screenshot shows the Yii Framework website's extension page for 'smtp-mail'. The page header includes the Yii Framework logo and navigation links: about, downloads, documentation, development, community, and a search bar. The main content area has tabs for DOWNLOADS, Framework, Extensions, Demos, and Logo. The 'smtp-mail' extension is highlighted with a 'Follow @yiiframework' button. Below this, it states 'I created one SMTP mail extension using phpmailer class' and shows '8 followers'. There are buttons for 'Add the SMTP Details in main.php' and 'Create Mailfunction'. A code block shows the configuration for the 'smtpmail' component in the 'components' array. On the right, there's a 'Create extension' button, a search bar, and a list of downloads including 'smtpmail.zip'. Below that, it shows the license (GNU General Public License v3), developer (mbala), category (Mail), votes (+6), download count (1,632 times), creation date (Feb 27, 2013), last update date (Feb 27, 2013), and tags (smtp, mail, mailer). At the bottom right, there are 'Related Extensions' like 'multi-email-validator' and 'dpsmailer'.

demos · guide · class reference · wiki · extensions · forum · live chat · login

yiiframework

about downloads documentation development community

DOWNLOADS Framework Extensions Demos Logo

smtp-mail Follow @yiiframework

I created one SMTP mail extension using phpmailer class

8 followers

Create extension

6 0 report it

SEARCH EXTENSIONS

Find

Downloads

» smtpmail.zip

See all »

License: GNU General Public License v3

Developed by: mbala

Category: Mail

Votes: +6

Downloaded: 1,632 times

Created on: Feb 27, 2013

Last updated: Feb 27, 2013

Tags: smtp, mail, mailer

Related Extensions

multi-email-validator

dpsmailer

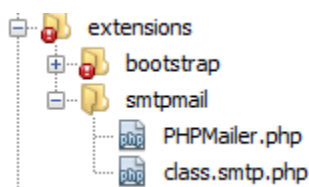
Add the SMTP Details in main.php

Download the extension and add it into protected/extension folder. Now configure the smtp details in protected/config/main.php

```
'components'=>array(
    'smtpmail'=>array(
        'class'=>'application.extensions.smtpmail.PHPMailer',
        'Host'=>'mail.yourdomain.com',
        'Username'=>'test@yourdomain.com',
        'Password'=>'test',
        'Mailer'=>'smtp',
        'Port'=>26,
        'SMTPAuth'=>true,
    ),
),
```

การติดตั้ง

ดาวน์โหลด smtp-mail มาไว้ใน protected/extensions/smtpmail/



จากนั้นเปิดไฟล์ protected/config/main.php แล้วเพิ่มการตั้งค่าในส่วนของ Components


```
46         'components'=>array(  
47             'Smtplib'=>array(  
48                 'class'=>'ext.smtplib.PHPMailer',  
49                 'Host'=>'mail.yourdomain.com',  
50                 'Username'=>'test@yourdomain.com',  
51                 'Password'=>'test',  
52                 'Mailer'=>'smtp',  
53                 'Port'=>26,  
54                 'SMTPAuth'=>true,  
55             ),  
56         ),
```

กำหนด Host, Username, Password และ Port ในการเชื่อมต่อ SMTP Server ให้เรียบร้อย

การใช้งาน

สำหรับการใช้งานให้ใส่ Code ในส่วนของ Action ใน Controller

```
110 public function mailsend($to,$from,$subject,$message){  
111     $mail=Yii::app()->Smtplib;  
112     $mail->SetFrom($from, 'From Name');  
113     $mail->Subject    = $subject;  
114     $mail->MsgHTML($message);  
115     $mail->AddAddress($to, "");  
116     if(!$mail->Send()) {  
117         echo "Mailer Error: " . $mail->ErrorInfo;  
118     }else {  
119         echo "Message sent!";  
120     }  
121 }
```

บทที่ 3 ว่าด้วยเรื่องของ Model

Database Active Record

มาเรียนรู้กันนะครับว่าการ Create Read Update และ Delete ใน Model นั้นสามารถเขียน Code แบบไหนได้บ้างครับไปดูกันเลยครับ

Create

```
$post = new Post;  
$post->title = "หัวข้อ";  
$post->content = "เนื้อหา";  
$post->save();
```

Read

ตัวอย่างที่ 1

```
$post = Post::model()->find(array(  
    'select' => 'title',  
    'condition' => 'id=:postId',  
    'params' => array(':postId'=>5),  
));
```

ตัวอย่างที่ 2

```
$post = Post::model()->find('id=:postId', array(':postId'=>5));
```

ตัวอย่างที่ 3

```
$post = Post::model()->findBySql($sql,$params);
```

ตัวอย่างที่ 4

```
$criteria = new CDbCriteria;  
$criteria->select = 'หัวข้อ';  
$criteria->condition = 'id=:postId';  
$criteria->params = array(':postId'=>5);  
$post = Post::model()->find($criteria);
```

ตัวอย่างที่ 5

การใช้ Query Builder

```
$user = Yii::app()->db->createCommand()  
->select('id,username,firstname,lastname')  
->from('table_user u')  
->join('table_profile p', 'u.id=p.user_id')  
->where('id=:Id',array(':Id'=>$id))  
->queryRow()
```

เป็นเพียงตัวอย่างบางส่วนเท่านั้นนะครับ ที่สามารถใช้งานพื้นฐานทั่วไป โดยสามารถดูเพิ่มเติมได้ที่

<http://www.yiiframework.com/doc/guide/1.1/en/database.ar>

SLUG

การทำ URL ให้เหมาะกับ Search Engine หรือที่เรียกกันคุ้นหูกันว่า Search Engine Optimization หรือ SEO ส่วนหนึ่งก็คือเรื่องของ URL โดยจะมีลักษณะดังต่อไปนี้

<http://www.programmerthailand.com/vdos/1-yii-framework-คืออะไร> จะเห็นว่า 1-yii-framework-คืออะไร จะมีความหมายดังต่อไปนี้

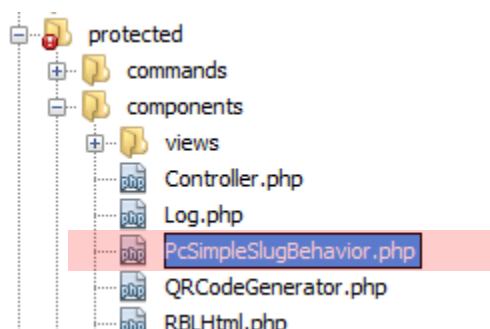
1 คือ ตัวเลข id ของเนื้อหาที่อยู่ในตารางเป็น Primary Key

-yii-framework-คืออะไร คือ ชื่อหัวเรื่องที่เราได้ Post เข้าไปบันทึก โดยตัว SLUG จะทำหน้าที่ตัดช่องว่างและใส่เครื่องหมาย - แทน

ใน Yii Framework มี Extension ที่ทำเรื่องนี้โดยเฉพาะ โดยจะขอแนะนำ PcsimpleSlugBehavior โดยสามารถดาวน์โหลดได้ที่

<http://www.yiiframework.com/extension/pcsimpleslugbehavior/>

หลังจากดาวน์โหลดเสร็จเรียบร้อยแล้วให้นำไปวางที่ protected/components/PcSimpleSlugBehavior.php ดังนี้



เพิ่ม behaviors เข้าไปใน Model ของเรา protected/models/Vdos.php

```
public function behaviors() {  
    return array(  
        'slugmaker' => array(  
            'class' => 'PcSimpleSlugBehavior',  
            // 'sourceIdAttr' => the 'id' attribute n  
            // 'sourceStringPrepareMethod' => If defi  
            // 'sourceStringAttr' => the 'main string  
            // 'maxChars' => Maximum allowed slug tot  
            // 'avoidIdPrefixing' => Setting this to  
        )  
    );  
}
```

ไฟล์ protected/controllers/VdosController.php กำหนดให้มีการเรียก generateUniqueSlug()
ใน actionCreate() ส่วนของ Post

```
if (isset($_POST['Vdos'])) {  
    $rating = new Rating();  
    $rating->save();  
    $model->rating_id = $rating->id;  
    $model->attributes = $_POST['Vdos'];  
    $model->user_id = Yii::app()->user->getId();  
    $model->created = date('Y-m-d H:i:s', time());  
    $model->modified = date('Y-m-d H:i:s', time());  
    $model->slug = $model->generateUniqueSlug();  
    if ($model->save()) {  
  
        $this->redirect(array('/Vdos/vdoview', 'slug' => $model->slug));  
    }  
}
```

ใน actionUpdate(\$id) เพิ่ม \$model->slug = \$model->generateUniqueSlug(); เช่นกับ

```
if (isset($_POST['Vdos'])) {  
    //$model->attributes=$_POST['Vdos'];  
    $model->attributes = $_POST['Vdos'];  
    //$model->user_id = Yii::app()->user->getId();  
    $model->modified = date('Y-m-d H:i:s', time());  
    $model->slug = $model->generateUniqueSlug();  
  
    if ($model->save()) {  
  
        $this->redirect(array('/Vdos/vdoview', 'slug' => $model->slug));  
    }  
}
```

TAGABLE

ในเรื่องของ Tag นั้นก็ค่อนข้างทำยากหน่อยครับ เพราะต้องตรวจสอบหลายส่วน เช่นเมื่อโพส Tag เข้ามาถ้ายังไม่มีให้เพิ่มในตาราง แต่ถ้ามีแล้วให้บวกจำนวนเพิ่มขึ้น และถ้ามีการลบโพสออกก็ให้ลบ Tag ออกด้วยดังนั้นการเขียนโปรแกรมจึงค่อนข้างซับซ้อนหน่อย แต่เราก็สามารถเรียนรู้และทำได้จากตัวอย่างนี้ครับ

มาดูในส่วนของตารางเก็บข้อมูลกันก่อนนะครับ ตัวอย่างนี้จะใช้ตาราง 2 ตารางนั้นคือ blog_post และ blog_tag ครับ

ตาราง blog_post

Name	Type	Collation	Attributes	Null	Default	Extra
id	int(11)			No	None	AUTO_INCREMENT
title	varchar(128)	utf8_unicode_ci		No	None	
content	text	utf8_unicode_ci		No	None	
tags	text	utf8_unicode_ci		Yes	NULL	
status	int(11)			No	None	
create_time	int(11)			Yes	NULL	
update_time	int(11)			Yes	NULL	
author_id	int(11)			No	None	

ตาราง blog_tag

Name	Type	Collation	Attributes	Null	Default	Extra
id	int(11)			No	None	AUTO_INCREMENT
name	varchar(128)	utf8_unicode_ci		No	None	
frequency	int(11)			Yes	1	

ส่วนต่อมาเป็นส่วนของ Model นั้นมี 2 ไฟล์ครับ คือไฟล์ protected/models/Tag.php

```
<?php
```

```
class Tag extends CActiveRecord{
```

```
    public static function model($className=__CLASS__){
        return parent::model($className);
    }
```

```
    public function tableName(){
        return 'blog_tag';
    }
```

```
    public function rules(){
        return array(
            array('name', 'required'),
            array('frequency', 'numerical', 'integerOnly'=>true),
            array('name', 'length', 'max'=>128),
        );
    }
```

```
    public function relations(){
        return array(
        );
    }
```

```
    public function attributeLabels(){
        return array(
            'id' => 'Id',
            'name' => 'Name',
            'frequency' => 'Frequency',
        );
    }
```

```
    public function findTagWeights($limit=20){
        $models=$this->findAll(array(
            'order'=>'frequency DESC',
            'limit'=>$limit,
        ));

        $total=0;
        foreach($models as $model)
            $total+=$model->frequency;

        $tags=array();
        if($total>0)
        {
            foreach($models as $model)
                $tags[$model->name]=8+(int) (16*$model->frequency/($total+10));
            ksort($tags);
        }
        return $tags;
    }
```

```
public function suggestTags($keyword,$limit=20){
    $tags=$this->findAll(array(
        'condition'=>'name LIKE :keyword',
        'order'=>'frequency DESC, Name',
        'limit'=>$limit,
        'params'=>array(
            ':keyword'=>'%'.strtr($keyword,array(
                '%'=>'\\%', '_'=>'\\_', '\\'=>'\\\\\\').'%',
            ),
        ));
    $names=array();
    foreach($tags as $tag)
        $names[]=$tag->name;
    return $names;
}

public static function string2array($tags)
{
    return preg_split('/\\s*,\\s*/',trim($tags),-1,PREG_SPLIT_NO_EMPTY);
}

public static function array2string($tags)
{
    return implode(', ', $tags);
}
```



```
public function updateFrequency($oldTags, $newTags){
    $oldTags=self::string2array($oldTags);
    $newTags=self::string2array($newTags);
    $this->addTags(array_values(array_diff($newTags,$oldTags)));
    $this->removeTags(array_values(array_diff($oldTags,$newTags)));
}

public function addTags($tags){
    $criteria=new CDbCriteria;
    $criteria->addInCondition('name',$tags);
    $this->updateCounters(array('frequency'=>1),$criteria);
    foreach($tags as $name)
    {
        if(!$this->exists('name=:name',array(':name'=>$name)))
        {
            $tag=new Tag;
            $tag->name=$name;
            $tag->frequency=1;
            $tag->save();
        }
    }
}

public function removeTags($tags)
{
    if(empty($tags))
        return;
    $criteria=new CDbCriteria;
    $criteria->addInCondition('name',$tags);
    $this->updateCounters(array('frequency'=>-1),$criteria);
    $this->deleteAll('frequency<=0');
}
}
```

๒๒๓๒ Model ๒๒๓๒ protected/models/post.php

```
<?php
class Post extends CActiveRecord{
    const STATUS_DRAFT=1;
    const STATUS_PUBLISHED=2;
    const STATUS_ARCHIVED=3;

    private $_oldTags;

    public static function model($className=__CLASS__){
        return parent::model($className);
    }
    public function tableName(){
        return 'blog_post';
    }
    public function rules(){
        // NOTE: you should only define rules for those attributes that
        // will receive user inputs.
        return array(
            array('title, content, status', 'required'),
            array('status', 'in', 'range'=>array(1,2,3)),
            array('title', 'length', 'max'=>128),
            array('tags', 'match', 'pattern'=>'/^[\\w\\s,]+$/',
                'message'=>'Tags can only contain word characters.')
            array('tags', 'normalizeTags'),

            array('title, status', 'safe', 'on'=>'search'),
        );
    }
}
```

```
public function relations(){
    return array(
        'author' => array(self::BELONGS_TO, 'User', 'author_id'),
        'comments' => array(self::HAS_MANY, 'Comment', 'post_id',
            'condition'=>'comments.status='.Comment::STATUS_APPROVED,
            'order'=>'comments.create_time DESC'),
        'commentCount' => array(self::STAT, 'Comment', 'post_id',
            'condition'=>'status='.Comment::STATUS_APPROVED),
    );
}

public function attributeLabels(){
    return array(
        'id' => 'Id',
        'title' => 'Title',
        'content' => 'Content',
        'tags' => 'Tags',
        'status' => 'Status',
        'create_time' => 'Create Time',
        'update_time' => 'Update Time',
        'author_id' => 'Author',
    );
}

public function getUrl(){
    return Yii::app()->createUrl('/blog/post/view', array(
        'id'=>$this->id,
        'title'=>$this->title,
    ));
}

public function getTagLinks()
{
    $links=array();
    foreach(Tag::string2array($this->tags) as $tag)
        $links[]=CHtml::link(CHtml::encode($tag),
            array('/blog/post/index', 'tag'=>$tag));
    return $links;
}

public function normalizeTags($attribute,$params){
    $this->tags=Tag::array2string(array_unique(Tag::string2array($this->tags)));
}
```

```
public function addComment($comment) {  
    if(Yii::app()->params['commentNeedApproval'])  
        $comment->status=Comment::STATUS_PENDING;  
    else  
        $comment->status=Comment::STATUS_APPROVED;  
    $comment->post_id=$this->id;  
    return $comment->save();  
}
```

```
protected function afterFind() {  
    parent::afterFind();  
    $this->_oldTags=$this->tags;  
}
```

```
protected function beforeSave() {  
    if(parent::beforeSave())  
    {  
        if($this->isNewRecord)  
        {  
            $this->create_time=$this->update_time=time();  
            $this->author_id=Yii::app()->user->id;  
        }  
        else  
            $this->update_time=time();  
        return true;  
    }  
    else  
        return false;  
}
```

```
protected function afterSave() {  
    parent::afterSave();  
    Tag::model()->updateFrequency($this->_oldTags, $this->tags);  
}
```

```
protected function afterDelete(){
    parent::afterDelete();
    Comment::model()->deleteAll('post_id='.$this->id);
    Tag::model()->updateFrequency($this->tags, '');
}

public function search(){
    $criteria=new CDbCriteria;

    $criteria->compare('title',$this->title,true);

    $criteria->compare('status',$this->status);

    return new CActiveDataProvider('Post', array(
        'criteria'=>$criteria,
        'sort'=>array(
            'defaultOrder'=>'status, update_time DESC',
        ),
    ));
}
```

เพิ่ม actionSuggestTags เพื่อใช้ในการค้นหา Tag สำหรับ Post ต่างๆ ในไฟล์
protected/controllers/PostController.php

```
public function actionSuggestTags() {
    if (isset($_GET['q']) && ($keyword = trim($_GET['q'])) != '') {
        $tags = Tag::model()->suggestTags($keyword);
        if ($tags != array())
            echo implode("\n", $tags);
    }
}
```

กำหนดการแสดงผล Tag ตามจำนวน ความถี่ของ Tag ในไฟล์
protected/components/TagCloud.php

```
class TagCloud extends CPortlet
{
    public $title='Tags';
    public $maxTags=20;

    protected function renderContent()
    {
        $tags=Tag::model()->findTagWeights($this->maxTags);

        foreach($tags as $tag=>$weight)
        {
            $link=CHtml::link(CHtml::encode($tag), array('post/index','tag'=>$tag));
            echo CHtml::tag('span', array(
                'class'=>'tag',
                'style'=>"font-size:{$weight}pt",
            ), $link)."\n";
        }
    }
}
```

เมื่อต้องการแสดงผล Tag ซึ่งเรียกว่า TagCloud ให้ทำการเรียก

```
<?php
    $this->widget('TagCloud', array(
        'maxTags' => 20,
    ));
?>
```

Tags

[blog](#) [book](#) [bootstrap](#) [extension](#) [php](#) [yii](#) [yii framework](#)

บทที่ 4 ว่าด้วยเรื่องของ Controller

การสร้าง Criteria ในการ Select ข้อมูล

Criteria ถือว่าเป็นพระเอกของการ Select ข้อมูลเลยทีเดียวครับ เราสามารถสร้าง Criteria ได้หลายแบบ มาดูตัวอย่างบางส่วนที่จะช่วยให้เราสามารถเข้าใจเรื่องของ Criteria มากยิ่งขึ้นครับ

โดยปกติ เราจะสร้าง Criteria ดังนี้ เป็นการสร้างเงื่อนไข WHERE price > 20 นั่นเอง

```
$criteria = new CDbCriteria();  
$criteria->condition = "price > 20";  
$Products = Product::model()->findAll($criteria)
```

หรือหากต้องการสร้าง Criteria พร้อมกับตอนที่ new ก็ได้เพียงส่งให้ Constructor ดังนี้

```
$criteria = new CDbCriteria(array('condition' => 'price > 20'));  
$Products = Product::model()->findAll($criteria);
```

หรือหากต้องการใช้คำสั่ง Limit ด้วยก็ได้ครับโดยใช้ \$criteria->limit ดังนี้

```
$criteria = new CDbCriteria();  
$criteria->condition = "price > 20";  
$criteria->limit = 1;  
$Products = Product::model()->findAll($criteria);
```

หรือหากต้องการใช้ Offset ด้วยก็ได้ครับโดยใช้ \$criteria->offset (หมายเหตุ บางครั้งเมื่อใช้งาน Offset และ Limit อาจจะต้อง Disable pagination ใน CActiveDataProvider) ดังนี้

```
$criteria = new CDbCriteria();  
$criteria->condition = "price > 20";  
$criteria->limit = 1;  
$criteria->offset = 1;  
$Products = Product::model()->findAll($criteria);
```

หรือหากต้องการใช้การเรียงลำดับ Order ด้วยก็ได้ครับโดยใช้ \$criteria->order ดังนี้

```
$criteria = new CDbCriteria();  
$criteria->condition = "price > 20";  
$criteria->limit = 1;  
$criteria->offset = 1;
```

```
$criteria->order = "name ASC";  
$Products = Product::model()->findAll($criteria);
```

หรือหากต้องการเลือกบาง Column ก็ได้ครับ โดยใช้ `$criteria->select` ครับ

```
$criteria = new CDbCriteria();  
$criteria->condition = "price > 20";  
$criteria->limit = 1;  
$criteria->offset = 1;  
$criteria->order = "name ASC";  
$criteria->select = "id, name";  
$Products = Product::model()->findAll($criteria);
```

หรือหากต้องการ Join หลายตารางก็สามารถทำได้ครับโดยใช้ `$criteria->with` ดังนี้

```
$criteria = new CDbCriteria;  
$criteria->with = array('foreign_table1',  
                      'foreign_table2',  
                      'foreign_table2.foreign_table3');  
$criteria->together = true; // ADDED THIS  
$criteria->select = array('id');  
$criteria->condition = "foreign_table1.col1=:col_val AND  
                      foreign_table3.col3=:col_val2";  
$criteria->params = array(':col_val' => some_val, ':col_val2' =>  
other_val);  
$criteria->order = 'foreign_table3.col5 DESC';  
$criteria->limit = 10;
```


การสร้างระบบ User Login จากฐานข้อมูล

สำหรับการสร้างระบบ User นั้นโดยทั่วไปแล้ว User จะถูกเก็บไว้ในตารางในที่นี่คือตาราง users โดยจะมี Model Users เป็นส่วนทำงานกับตาราง users นั้นเอง

การสร้างระบบ Login โดยตรวจสอบข้อมูลจากฟอร์มว่าตรงกันกับฐานข้อมูลนั้นเราจะทำการแก้ไข Class UserIdentity ที่อยู่ใน protected/components/UserIdentity.php

การ Login เข้าสู่ระบบนั้นจะเป็นการดึงข้อมูลจากตาราง users มาตรวจสอบ Username และ Password ที่ได้กรอกจากฟอร์มเข้าสู่ระบบแล้วมาตรวจสอบว่าตรงกันกับในตาราง users หรือไม่ ทั้งนี้ในการบันทึกข้อมูล Password ในตาราง users นั้นก็จะใช้การเข้ารหัสข้อมูลด้วย ในที่นี่จะเข้ารหัสเป็น MD5 โดยจะปรับ Class UserIdentity ดังนี้

```
class UserIdentity extends CUserIdentity{
    private $_id;
    public function authenticate(){
        $user = Users::model()->findByAttributes(array('username'=>$this->username));

        if($user === null){
            $this->errorCode = self::ERROR_USERNAME_INVALID;
        }else if($user->password !== md5($this->password)){
            $this->errorCode = self::ERROR_PASSWORD_INVALID;
        }else{
            $this->errorCode = self::ERROR_NONE;
            $this->setState('role',$user->user_type);
            $this->_id = $user->id;
        }
        return !$this->errorCode;
    }
    public function getId(){
        return $this->_id;
    }
}
```

เพียงเท่านี้ก็สามารถ Login เข้าสู่ระบบโดยใช้ข้อมูลจากตารางในฐานข้อมูลได้แล้วครับ

บทที่ 5 ว่าด้วยเรื่องของ View

การ Register ไล่ว์ JavaScript และ CSS

เราสามารถ Register ไล่ว์ JavaScript และ CSS ในหน้า Layout และหน้าเว็บของเราได้โดยใช้ Script ดังนี้

การ Register Core Script

เป็นการโหลด Script พื้นฐานโดยตัว Script จะถูกบรรจุอยู่ที่ `yii/framework/web/js/source`

หากต้องการดูว่าสามารถ Register Script ตัวไหนได้บ้างให้ไปดูได้ที่ไล่ว์ `framework/web/js/packages.php`

วิธีการใช้งานให้ทำการเขียน Code บริเวณที่ต้องการ Register เช่นในส่วน Header ดังนี้ `Yii::app()->clientScript->registerCoreScript('jquery');`

เป็นการ Register ไล่ว์ jQuery โดยจะแสดงผลดังนี้

```
<script type="text/javascript" src="/assets/3869f222/jquery.js"></script>
```

การ Register JavaScript ไล่ว์

JavaScript ก็เป็น Script อีกอย่างที่ได้ใช้กันค่อนข้างบ่อยมากในการทำ Application โดยมีตัวอย่างของการ Register Script ดังนี้

```
Yii::app()->clientScript->registerScriptFile(Yii::app()->theme->baseUrl.'/assets/js/default.js',CClientScript::POS_END);
```

จะได้ผลลัพธ์ดังนี้

```
<script type="text/javascript"
src="/themes/hanuman/assets/js/default.js"></script>
```

จะเห็นว่าคำสั่ง `CClientScript::POS_END` โดยที่ POS มาจากคำว่า Position นั่นเองครับ นั่นก็แสดงว่าเราสามารถวางตำแหน่งของ Script ของเราได้หลายตำแหน่ง ไปดูกันเลยว่าวางได้ตำแหน่งไหนบ้างครับ

วางที่ Tag Head ของ Application

```
Yii::app()->clientScript->registerScriptFile(Yii::app()->theme->baseUrl.'/assets/js/default.js',CClientScript::POS_HEAD);
```

วางหลังจาก Tag Body ของ Application

```
Yii::app()->clientScript->registerScriptFile(Yii::app()->theme->baseUrl.'/assets/js/default.js',CClientScript::POS_BEGIN);
```

วางด้านล่างของเว็บก่อนปิด Tag Body

```
Yii::app()->clientScript->registerScriptFile(Yii::app()->theme->baseUrl.'/assets/js/default.js',CClientScript::POS_END);
```

การ Register ไฟล์ CSS

ไฟล์ CSS หรือ Cascade Style Sheet เป็นไฟล์ที่กำหนดรูปแบบของเว็บไซต์ เช่น สี สัน การจัดตำแหน่ง ต่างๆ ใน Application เป็นต้นโดยมีการใช้งานดังนี้

```
Yii::app()->clientScript->registerCssFile(Yii::app()->theme->baseUrl.'/assets/css/style.css');
```

จะได้ผลลัพธ์ดังนี้

```
<link rel="stylesheet" type="text/css" href="/themes/hanuman/assets/css/style.css" />
```

นอกจากนั้นเรายังสามารถเพิ่มรายละเอียดเข้าไปได้ เช่นเป็นการกำหนดว่า CSS นี้เป็นส่วนของ Screen หรือว่าส่วน Print ได้อีกด้วย

```
Yii::app()->clientScript->registerCssFile(Yii::app()->theme->baseUrl.'/assets/css/print.css', 'print');
```

จะได้ผลลัพธ์ดังนี้

```
<link rel="stylesheet" type="text/css" href="/themes/hanuman/assets/css/style.css" media="print" />
```

การ Register Script ที่เป็ JavaScript แบบ Inline

การ Register Script ลักษณะนี้เรียกว่าเป็นการ Register แบบ Inline โดยหลังจากประมวลผลแล้วจะมีการเขียน Script ลงบนหน้า Application โดยตรงโดยที่ไม่ได้ Link มาเหมือนแบบอื่นๆ โดยมีตัวอย่างดังต่อไปนี้

```
<?php
Yii::app()->clientScript->registerScript('JS',"
    $('#loading-btn').on(ace.click_event, function() {
        var btn = $(this);
        btn.button('loading')
        setTimeout(function() {
            btn.button('reset')
        }, 2000)
    });

    $('#id-button-borders').attr('checked', 'checked').on('click',
function() {
    $('#default-buttons .btn').toggleClass('no-border');
});
");
?>
```

จะเห็นว่าจะมีการสร้าง jQuery(function(\$){...} ให้โดยอัตโนมัติ

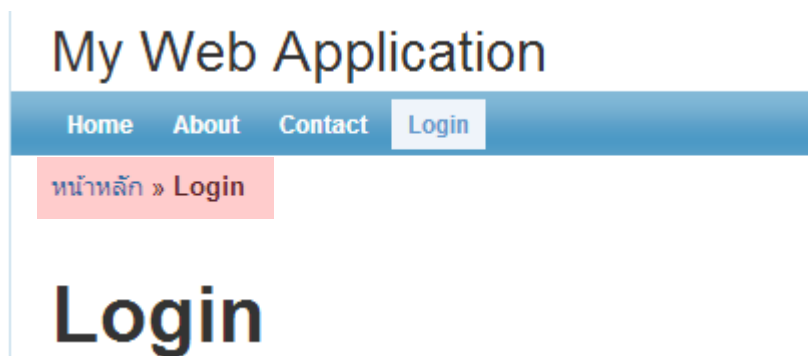
```
<script type="text/javascript">
/**/
jQuery(function($) {
    $('#loading-btn').on(ace.click_event, function() {
        var btn = $(this);
        btn.button('loading')
        setTimeout(function() {
            btn.button('reset')
        }, 2000)
    });
    $('#id-button-borders').attr('checked', 'checked').on('click',
function() {
    $('#default-buttons .btn').toggleClass('no-border');
});
});
/*]]&gt;*/
&lt;/script&gt;</pre></div><div data-bbox="361 908 633 954" data-label="Page-Footer"><p><img alt="yii framework logo" data-bbox="412 912 455 938"/> yii framework<br/>www.programmerthailand.com</p></div>
```

การเปลี่ยนข้อความ “Home” ใน Breadcrumbs เป็น “หน้าหลัก”

Breadcrumbs หรือเรียกอีกอย่างหนึ่งว่าป้ายนำทาง โดยการใช้งาน Breadcrumbs นั้นค่าเริ่มต้นของหน้าหลักของ Application เรานั้นก็คือคำว่า Home ซึ่งเราจะสามารถเปลี่ยนเป็นภาษาไทยได้ โดยแทนที่คำว่า Home เป็นคำว่าหน้าหลัก โดยเปิดไฟล์ที่เป็น Layout ของ Application เรา ในที่นี้จะใช้ Layout พื้นฐานโดยจะอยู่ที่ protected/views/layouts/main.php

```
53 <?php if(isset($this->breadcrumbs)):?>
54     <?php $this->widget('zii.widgets.CBreadcrumbs', array(
55         'homeLink'=> CHtml::link('หน้าหลัก',array('/site/index')),
56         'links'=>$this->breadcrumbs,
57     )); ?><!-- breadcrumbs -->
58 <?php endif?>
```

ทดลองเปิดหน้า Login หรือ เข้าสู่ระบบดูจะเห็นว่าการแสดงผลดังนี้

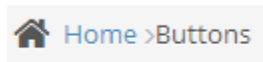


การปรับ Breadcrumbs ให้อยู่ในรูปแบบของ Tag ul และ li

โดยส่วนใหญ่แล้ว Theme ที่มีการพัฒนาขึ้นมาในส่วนของ Breadcrumbs นั้นจะมีการแสดงผลในรูปแบบ Tag ul และ li โดยปกติแล้วถ้าไม่มีการกำหนดจะแสดงเป็น Tag div ซึ่งจะไม่สามารถใช้กับ Theme ได้ดังนั้นเราสามารถเปลี่ยนมาใช้ Tag ul และ li ได้ดังนี้

```
<?php if(isset($this->breadcrumbs)):
$this->widget('zii.widgets.CBreadcrumbs', array(
    'links'=>$this->breadcrumbs,
    'homeLink'=>false,
    'homeLink'=>'<li><i class="icon-home home-icon"></i><a href="'.Yii::app()->homeUrl.'">Home</a><i class="icon-angle-right arrow-icon"></i></li>',
    'tagName'=>'ul',
    'separator'=>'',
    'activeLinkTemplate'=>'<li><a href="{url}">{label}</a><span
class="divider"><i class="icon-angle-right arrow-icon"></i></span></li>',
    'inactiveLinkTemplate'=>'<li><span>{label}</span></li>',
    'htmlOptions'=>array ('class'=>'breadcrumb')
)); ?><!-- breadcrumbs -->
<?php endif; ?>
```

เมื่อแสดงผลแล้วจะได้นี้



เมื่อดู Code จะเห็นดังนี้

```
<ul class="breadcrumb">
<li><i class="icon-home home-icon"></i><a href="/hanuman-
template/index.php">Home</a><i class="icon-angle-right arrow-
icon"></i></li>
<li><span>Buttons</span></li>
</ul><!-- breadcrumbs -->
```

การสร้าง Drop Down List สำหรับการเก็บข้อมูลแบบ ENUM

ชนิดข้อมูลแบบ ENUM เป็นการเก็บข้อมูลลักษณะที่กำหนดค่าของข้อมูลไว้ล่วงหน้า อย่างเช่น รายชื่อจังหวัด ชนิดของผู้ใช้งาน เป็นต้น โดยมีการกำหนดลักษณะดังนี้

Name	Type	Length/Values	Default	Collation
province	ENUM	'กรุงเทพมหานคร','อุบล'	None	utf8_unicode_ci

สร้างไฟล์ protected/components/ZHtml.php

```
<?php
2 class ZHtml extends CHtml{
3     public static function enumItem($model,$attribute)
4     {
5         $attr=$attribute;
6         self::resolveName($model,$attr);
7         preg_match('/\((.*)\)/',$model->tableSchema->columns[$attr]->dbType,$matches);
8         foreach(explode(',',$matches[1]) as $value)
9         {
10             $value=str_replace('"',null,$value);
11             $values[$value]=Yii::t('enumItem',$value);
12         }
13
14         return $values;
15     }
16
17     public static function enumDropDownList($model, $attribute, $htmlOptions)
18     {
19         return CHtml::activeDropDownList($model, $attribute,ZHtml::enumItem($model, $attribute),
20             $htmlOptions);
21     }
22
23 }
24
```

สำหรับการใช้งานเปิดไฟล์ _form.php แล้วแก้ไขดังนี้

จังหวัด *

กรุงเทพมหานคร

▼

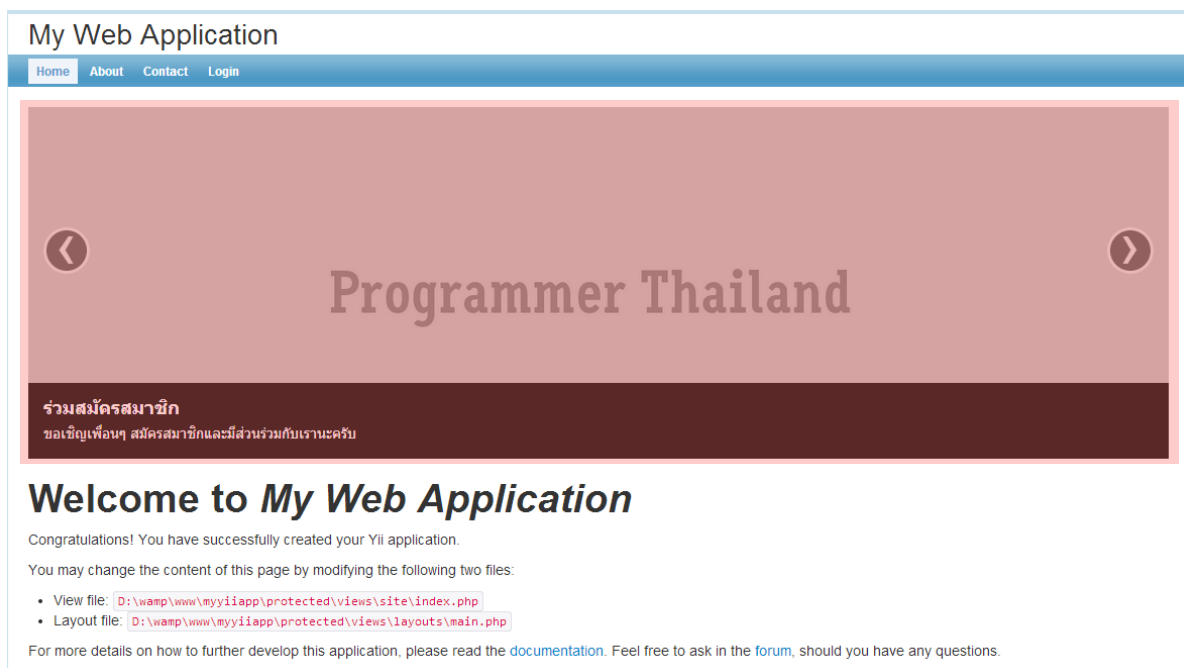
ที่มา: <http://www.yiiframework.com/forum/index.php/topic/10079-enum-db-type-in-yii/>

การสร้างม้าหมุน (Carousel) ใน Yii Booster Extension

ม้าหมุนหรือ Carousel เป็นการแสดงผลเน้นการอธิบายของหน้า Application ของเราโดยใน Yii Booster Extension นั้นมีการสร้างเป็น Widget ให้เราพร้อมสำหรับการเรียกใช้งานเรียบร้อยแล้ว ตัวอย่างเช่นหากเราต้องการให้ protected/views/site/index.php มีการแสดงผลเป็น Carousel ให้แก้ไขดังนี้

```
6 <?php
7 $this->widget('bootstrap.widgets.TbCarousel',array(
8     'items'=>array(
9         array(
10             'image'=>'http://placeholder.it/1300x400&text=Programmer Thailand',
11             'label'=>'สื่อเพื่อการเรียนรู้',
12             'caption'=>'มาเรียนรู้เทคโนโลยีกับ Programmer Thailand นะครับ',
13         ),
14         array(
15             'image'=>'http://placeholder.it/1300x400&text=Programmer Thailand',
16             'label'=>'ร่วมสมัครสมาชิก',
17             'caption'=>'ขอเชิญเพื่อนๆ สมัครสมาชิกและมีส่วนร่วมกับเรานะครับ',
18         ),
19     ),
20 ))
21 ?>
```

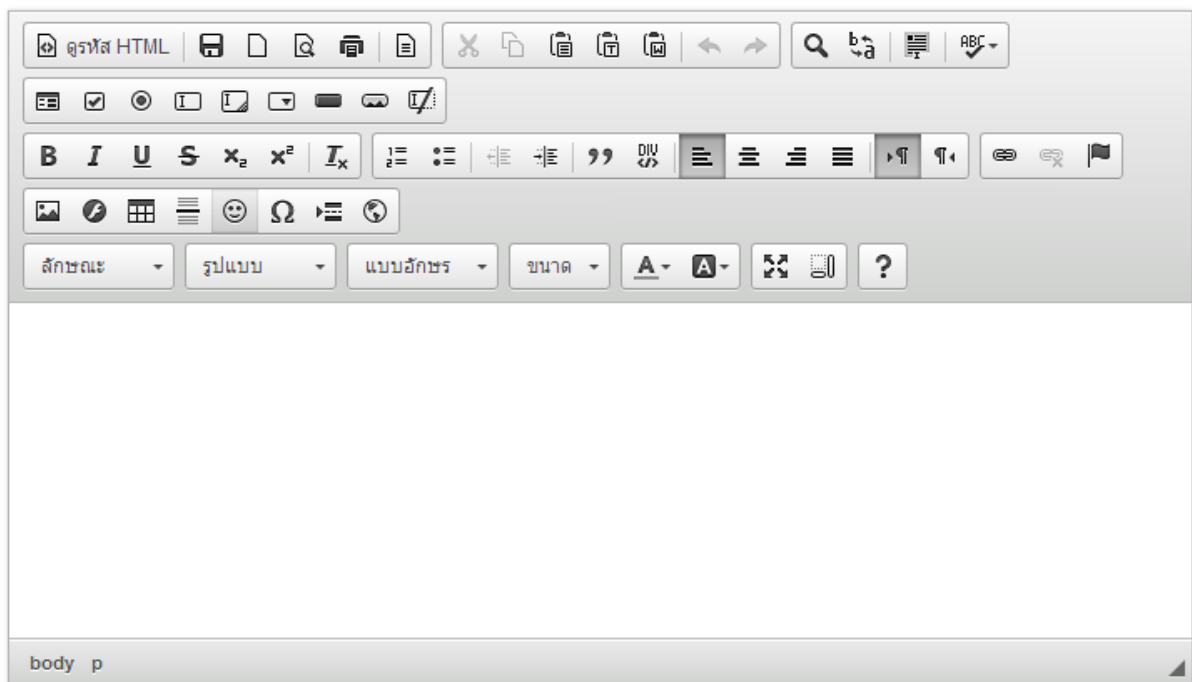
จากนั้นทดสอบเปิด index.php?r=site/index ก็จะปรากฏหน้า Application ดังนี้



การสร้าง CKEditor สำหรับ TextArea

ในการเพิ่มข้อมูล Field ที่เก็บข้อมูลแบบ Text ควรที่จะมี Editor สำหรับเป็นตัวช่วยเขียนเนื้อหา ตัวอย่างเช่นการเขียนบันทึก (Weblog) หากเรามีการกำหนดรูปแบบตัวหนา ตัวเอียง ตัวขีดเส้นใต้ หรืออื่นๆ แล้วบันทึกลงในฐานข้อมูล เมื่อมีการดึงข้อมูลมาแสดงผล รูปแบบต่างๆ ที่ได้มีการกำหนดไว้ก็จะแสดงผลตามด้วยเช่นกัน ดังนั้น ในช่องกรอกที่เป็นลักษณะ TextArea จึงควรมี Editor สำหรับช่วยเขียนเนื้อหา โดยใน YiiBooster นั้นได้รวมเอา CKEditor มาไว้ให้เรียบร้อยแล้ว โดยมีรูปร่างหน้าตาดังตัวอย่าง

ที่อยู่ *



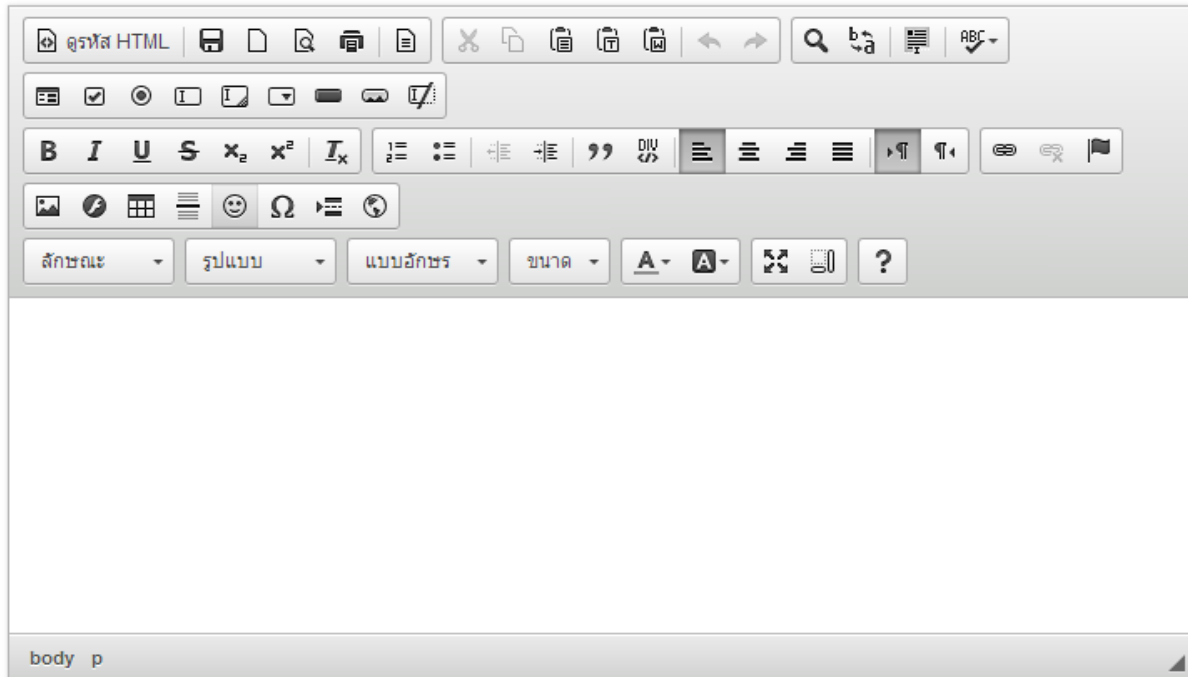
สำหรับการกำหนดให้กำหนดไฟล์ที่แสดงผลส่วนใหญจะอยู่ใน _form.php ดังนี้

```
<?php echo $form->labelEx($model,'address');?>
<?php $this->widget(
    'bootstrap.widgets.TbCKEditor',
    array(
        'model'=>$model,
        'attribute'=>'address',
    )
);
?>
```

<?php echo \$form->labelEx(\$model,'address');?> เป็นการแสดง Label ที่ดึงมาจาก Model

จากนั้นเรียกใช้งาน widget TbCKEditor เพื่อสร้าง Editor โดยส่ง \$model จาก controller เข้าไป และกำหนด Attribute เป็นตัวที่ต้องการในที่นี่คือ address นั่นเอง เมื่อเรียกใช้งานก็จะแสดงผลดังนี้

ที่อยู่ *



การดึงข้อมูลจากตารางอื่นมาแสดงเป็น Drop Down List

การนำข้อมูลจากตารางอื่นมาแสดงเป็น Drop Down List นั้นจะค่อนข้างใช้บ่อย เช่น การเพิ่มสินค้าที่ดึงประเภทสินค้ามาเป็นรายการให้เลือก เป็นต้น

โดยจะแก้ไขที่ไฟล์ _form.php

```
15         <?php echo $form->dropDownListRow($model,'product_type_id',
16             CHtml::listdata(ProductTypes::model()->findAll(
17                 array('order'=>'product_type ASC')),
18                 'id','product_type')); ?>
```

จากนั้นแสดงผลข้อมูลจะเห็นเป็น Drop Down List ดังนี้

Create Products

Fields with * are required.

ชื่อสินค้า *

ราคาสินค้า *

ประเภทสินค้า *

DVD

▼

DVD

หนังสือ

Create

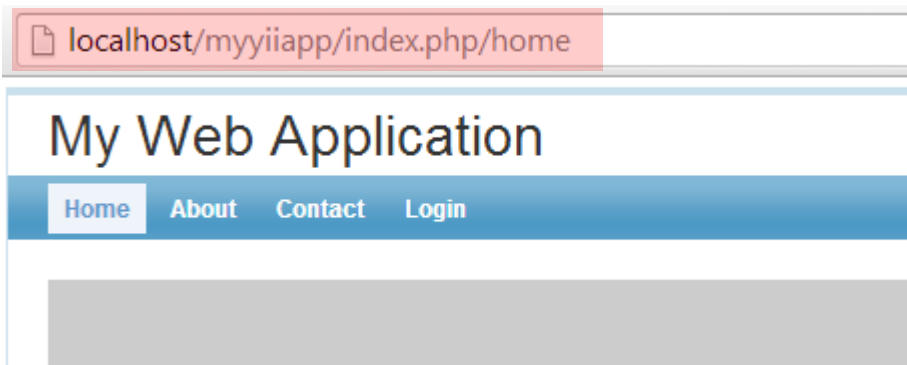
การเปลี่ยน site/index เป็น home ใน URL

สำหรับการเปลี่ยน URL ที่เรียกหน้าหลักของเว็บไซต์นั้นจะช่วยให้สามารถจดจำหน้าหลักได้ง่ายยิ่งขึ้น เช่น ก่อนเปลี่ยนจะเรียกเป็น index.php/site/index ให้เปลี่ยนเป็น index.php/home โดยเปิดไฟล์ protected/config/main.php จากนั้นตั้งค่าดังนี้

เพิ่ม 'home'=>'site/index', ที่ด้านบนของ Rules Array()

```
59         'urlManager'=>array(  
60             'urlFormat'=>'path',  
61             'rules'=>array(  
62                 'home'=>'/site/index',  
63                 '<controller:\w+>/<id:\d+>'=>'<controller>/view'  
64                 '<controller:\w+>/<action:\w+>/<id:\d+>'=>'<cont  
65                 '<controller:\w+>/<action:\w+>'=>'<controller>/<  
66  
67             ),  
68         ),
```

ทดลองกดที่หน้า Home ก็จะปรากฏ URL ดังนี้



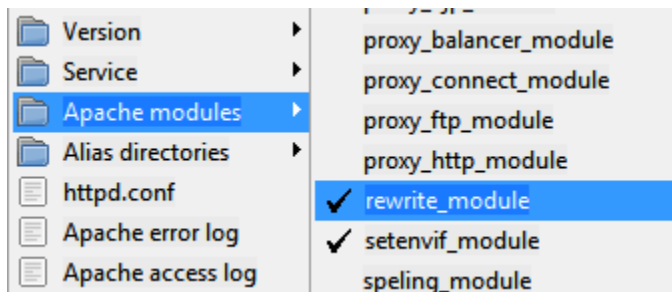
การซ่อนไฟล์ index.php เมื่อเรียก URL

การเรียก URL โดยปกติจะเรียก index.php เป็นไฟล์แรกที่ Run เราสามารถซ่อนไฟล์ index.php ได้ โดยใช้คุณสมบัติของ urlManager() โดยจะทำให้สามารถเรียก http://yourweb.com/controller/action เป็นต้น โดยมีการตั้งค่าดังนี้

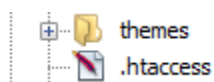
เพิ่ม 'showScriptName'=>false, และ 'caseSensitive'=>false, ในส่วนของ urlManager

```
59         'urlManager'=>array(  
60             'urlFormat'=>'path',  
61             'showScriptName'=>false,  
62             'caseSensitive'=>false,  
63             'rules'=>array(  
64                 '<controller:\w+>/<id:\d+>':  
65                 '<controller:\w+>/<action:\w+>',  
66                 '<controller:\w+>/<action:\w+>':  
67                 '<controller:\w+>/<action:\w+>',  
68                 'home'=>'/site/index',  
69             ),  
        ),
```

ทำการเปิด rewrite_mode ใน Apache



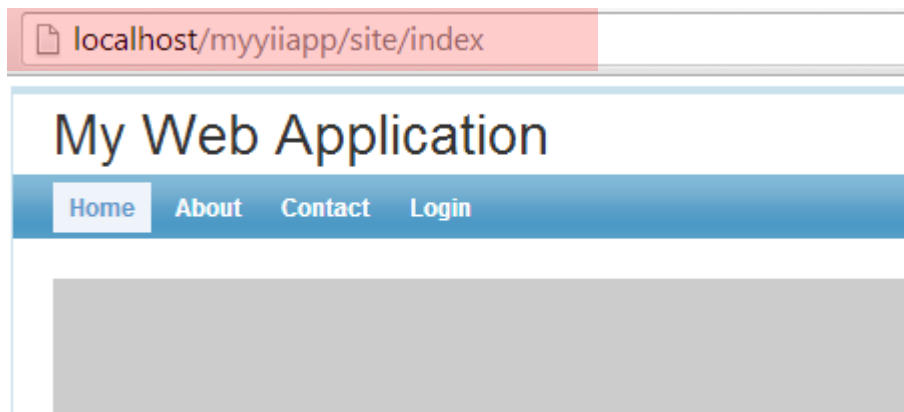
สร้างไฟล์ .htaccess ใน Root ของ Project



โดยมีข้อมูลข้างในดังนี้

```
1  RewriteEngine on  
2  
3  # if a directory or a file exists, use it directly  
4  RewriteCond %{REQUEST_FILENAME} !-f  
5  RewriteCond %{REQUEST_FILENAME} !-d  
6  
7  # otherwise forward it to index.php  
8  RewriteRule . index.php
```

ทดสอบการ Run โดย Refresh หน้า Application จะปรากฏการแสดงผลดังนี้



การสร้าง Captcha ให้กับฟอร์มด้วย CCaptcha

บ่อยครั้งที่เราจะสังเกตเห็นว่าเวลาที่มีการกรอกข้อมูลในฟอร์ม เช่น แบบฟอร์มสมัครสมาชิกจะมีช่องให้กรอกตัวอักษรที่อ่านไม่ค่อยออกหรืออ่านค่อนข้างยาก ตัวอักษรเหล่านั้นเรียกว่า Captcha ซึ่งสร้างขึ้นเพื่อป้องกันการกรอกข้อมูลอัตโนมัติของ Bot หรือ Script ที่จะมีการ Boom เข้ามาในเว็บหรือใน Application ของเรา ทำให้เกิด Record ที่เราไม่ต้องการเต็มไปด้วยหมดในตารางฐานข้อมูล โดยมีการกำหนดดังนี้

ในส่วนของ Model เพิ่ม array('verifyCode', 'captcha', 'allowEmpty' => !CCaptcha::checkRequirements()), ในส่วน rules()

```
public function rules() {  
    // NOTE: you should only define rules for those attributes that  
    // will receive user inputs.  
    return array(  
        array('firstname, lastname, username, password, address, province, zipcode, crea  
        array('firstname, lastname', 'length', 'max' => 100),  
        array('username, password', 'length', 'max' => 45),  
        array('province', 'length', 'max' => 39),  
        array('zipcode, user_type', 'length', 'max' => 5),  
        // The following rule is used by search().  
        // @todo Please remove those attributes that should not be searched.  
        array('id, firstname, lastname, username, password, address, province, zipcode,  
        array('verifyCode', 'captcha', 'allowEmpty' => !CCaptcha::checkRequirements()),  
    );  
}
```

ในส่วนของ Controller เพิ่ม function action()

```
public function actions() {  
    return array(  
        // captcha action renders the CAPTCHA  
        'captcha' => array(  
            'class' => 'CCaptchaAction',  
            'backColor' => 0xFFFFFF,  
        ),  
    );  
}
```

เพิ่ม captcha ในการตรวจสอบ Rules สำหรับการเข้าถึงแบบ Public

```
public function accessRules() {  
    return array(  
        array('allow', // allow all users to perform 'index' and 'view' act  
            'actions' => array('index', 'view', 'registration', 'captcha'),  
            'users' => array('*'),  
        ),  
    );  
}
```

ในส่วนของ View เพิ่ม Code สำหรับแสดง Captcha

```
<?php if(Captcha::checkRequirements()): ?>
<div class="row">
    <?php echo $form->labelEx($model,'verifyCode'); ?>
    <div>
        <?php $this->widget('Captcha'); ?>
        <?php echo $form->textField($model,'verifyCode'); ?>
    </div>
    <div class="hint">กรณหากกรอกสิ่งที่เห็นในภาพ</div>
    <?php echo $form->error($model,'verifyCode'); ?>
</div>
<?php endif; ?>
```

เมื่อแสดงผลจะได้ Captcha หน้าตาดังนี้

รหัสตรวจสอบ

fanogg

สร้างรหัสใหม่

กรณหากกรอกสิ่งที่เห็นในภาพ

การสร้างฟอร์มแบบ Horizontal สำหรับ Yii Booster Extension

มีบางครั้งที่เราต้องการให้การแสดงผลฟอร์มอยู่ในแนวเดียวกันกับ Object ของฟอร์มแบบแนวนอน ซึ่งเราสามารถกำหนดรูปแบบการแสดงผลของฟอร์มสำหรับ Yii Booster Extension โดยกำหนดที่ไฟล์ `_form.php` โดยเพิ่มตัวแปร `'type'=>'horizontal'`, ดังนี้

```
<?php $form=$this->beginWidget('bootstrap.widgets.TbActiveForm',array(
    'id'=>'products-form',
    'enableAjaxValidation'=>false,
    'type'=>'horizontal',
)); ?>
```

จากนั้นเรียกการแสดงผลได้ลักษณะดังนี้

Create Products

Fields with * are required.

ชื่อสินค้า *	<input type="text"/>
ราคาสินค้า *	<input type="text"/>
ประเภทสินค้า *	<input type="text" value="DVD"/>

Create

การแสดงผลจากตารางอื่นในความสัมพันธ์แบบ

BelongsTo

การนำข้อมูลจากตารางอื่นมาแสดงนั้นเกิดขึ้นจากที่เราสร้างความสัมพันธ์แบบ BelongsTo ไว้ ตัวอย่างเช่น ในการแสดงผลรายการสินค้า จะมีการดึงประเภทสินค้ามาแสดงด้วยจะต้องมีการตั้งค่าดังต่อไปนี้

ในส่วนของ Products Model มีการเชื่อมโยงความสัมพันธ์

```
public function relations()
{
    // NOTE: you may need to adjust the relation name and the related
    // class name for the relations automatically generated below.
    return array(
        'orders' => array(self::MANY_MANY, 'Orders',
            'orders_has_products(product_id, order_id)'),
        'productType' => array(self::BELONGS_TO,
            'ProductTypes', 'product_type_id'),
    );
}
```

ในส่วนของ views/products/_view.php

```
<b><?php echo CHtml::encode($data->getAttributeLabel('product_type_id')); ?></b>
<?php echo CHtml::encode($data->productType->product_type); ?>
<br />
```

เมื่อมีการเรียก URL products/index จะปรากฏหน้า Application ดังนี้

Products

Displaying 1-2 of 1 result.

ID: 1
ชื่อสินค้า: Yii Framework
ราคาสินค้า: 399
ประเภทสินค้า: หนังสือ

ID: 2
ชื่อสินค้า: Yi Framework Workshop
ราคาสินค้า: 399
ประเภทสินค้า: หนังสือ

ในส่วนของ views/view.php

```
<?php $this->widget('bootstrap.widgets.TbDetailView',array(
    'data'=>$model,
    'attributes'=>array(
        'id',
        'product_name',
        'product_price',
        'productType.product_type',
    ),
)); ?>
```

เมื่อมีการเรียก URL products/view/1

View Products #1







ID 1
ชื่อสินค้า Yii Framework
ราคาสินค้า 399
ประเภทสินค้า หนังสือ

ในส่วนของ views/admin.php

```
<?php
$this->widget('bootstrap.widgets.TbGridView', array(
    'id' => 'products-grid',
    'dataProvider' => $model->search(),
    'filter' => $model,
    'columns' => array(
        'id',
        'product_name',
        'product_price',
        'productType.product_type',
        array(
            'class' => 'bootstrap.widgets.TbButtonColumn',
        ),
    ),
));
?>
```

เมื่อมีการเรียก URL products/admin

Displaying 1-2 of 1 result.

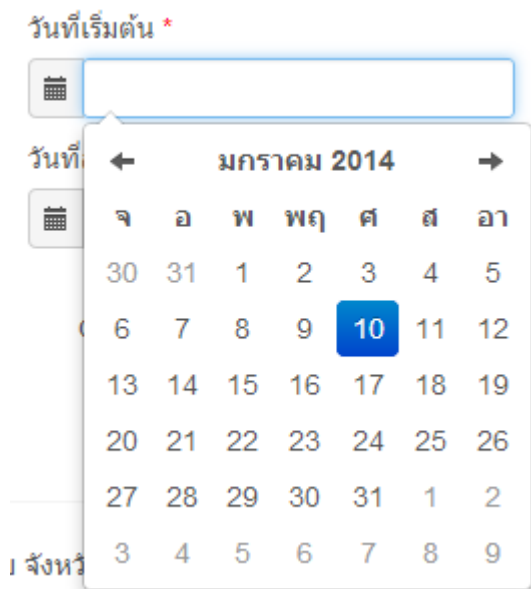
ID	ชื่อสินค้า	ราคาสินค้า	ประเภทสินค้า	
<input type="text"/>	<input type="text"/>	<input type="text"/>		
1	Yii Framework	399	หนังสือ	  
2	Yi Framework Workshop	399	หนังสือ	  

การสร้าง Date Picker สำหรับเก็บลง Field DATE ใน MySQL

สำหรับ Field เก็บข้อมูลแบบ DATE นั้นจะอยู่ในรูปของ ปี.ศ.-เดือน-วัน เช่น 2014-01-20 เป็นต้น ดังนั้นใน Yii Booster Extension จะมีส่วนเสริมของ Bootstrap นั่นคือ bootstrap.datepicker.js ซึ่งสามารถ เรียกใช้งานผ่าน Widget ของ Yii Booster Extension ได้ โดยกำหนดในฟอร์มสำหรับรับข้อมูลวันที่ พร้อมจัดรูปแบบให้อยู่ในรูปของ ปี.ศ.-เดือน-วัน สำหรับการเก็บข้อมูลลง Field DATE ในฐานข้อมูล ดังนี้

```
<?php
echo $form->datepickerRow($model, 'date_start',
    array('prepend'=>'<i class="icon-calendar"></i>'
        , 'options'=>array( 'format' => 'yyyy-mm-dd',
                            'weekStart'=> 1,
                            'showButtonPanel' => true,
                            'showAnim'=>'fold',)
    )
);
?>
```

เมื่อทำการแสดงผลจะปรากฏช่องกรอก หากมีการคลิกที่ช่องจะมีปฏิทินเกิดขึ้นดังนี้



การแสดงผลวันเวลาภาษาไทยจากฐานข้อมูล

การแสดงผลวันเวลาเป็นภาษาไทยโดยดึงข้อมูลมาจากฐานข้อมูลนั้น เราจะเรียกใช้ Object จาก Class CDateFormatter มาดูตัวอย่างในการแสดงผลดังนี้

วันอาทิตย์ที่ 5 มกราคม ค.ศ. 2014, 21 นาฬิกา 14 นาที 12 วินาที ICT

ไปดู Code กันเลยนะครับ ในส่วนของ View ที่ต้องการแสดงผล เราจะส่งวันเวลาเข้าไป

```
<?php echo Yii::app()->dateFormatter->formatDateTime(strtotime($data->created), 'full', 'full'); ?>
```

โดยหากต้องการการแสดงผลเพิ่มเติมสามารถดูเพิ่มเติมได้ที่

<http://www.yiiframework.com/doc/api/1.1/CDateFormatter>

การแก้ไขหน้า Error สำหรับ Yi Framework

สำหรับหน้า Error นั้นเราสามารถแก้ไขให้อยู่ในรูปของ Bootstrap ได้โดยการเรียกใช้ Class ที่แสดงเป็น Error ในที่นี้คือ `class="alert alert-danger"` โดยเราสามารถกำหนด path ได้โดยเข้าไปดูใน `protected/config/main.php` ในส่วนของ `errorHandler`

```
'errorHandler'=>array(  
    // use 'site/error' action to display errors  
    'errorAction'=>'site/error',  
),
```

จะเห็นว่าไฟล์ View ของ Error จะอยู่ที่ `views/site/error.php` เปิดไฟล์ดังกล่าวแล้วแก้ไขให้เป็นแบบฉบับที่ต้องการ ในที่นี้จะปรับแก้ไขให้อยู่ในรูปของ Bootstrap ดังนี้

```
<div class="alert alert-danger">  
    <h4>Error <?php echo $code; ?></h4>  
    <?php echo CHtml::encode($message); ?>  
</div>
```

ลองทดสอบเรียก URL ที่ไม่มี Action เช่น `testerror` จะแสดงหน้า Error ดังนี้

Error 404

ระบบไม่สามารถดำเนินการ "testerror" ตามคำสั่งได้

การสร้างและแสดง Flash Message สำหรับ Yii Booster Extension

Flash Message คือข้อความที่แสดงเมื่อมีการกระทำใน Action ตัวอย่างเช่น หากมีการเข้าสู่ระบบสำเร็จก็แสดงข้อความบอกว่าเข้าสู่ระบบเรียบร้อยแล้ว แต่ถ้าหากว่ากรอก Username หรือ Password ผิดก็ให้แสดงข้อความแจ้งว่า Username หรือ Password ไม่ถูกต้อง เป็นต้น

ใน Yii Booster Extension นั้นจะมี CSS Class ที่ได้กำหนดไว้คือ success, error, warning, danger และ info

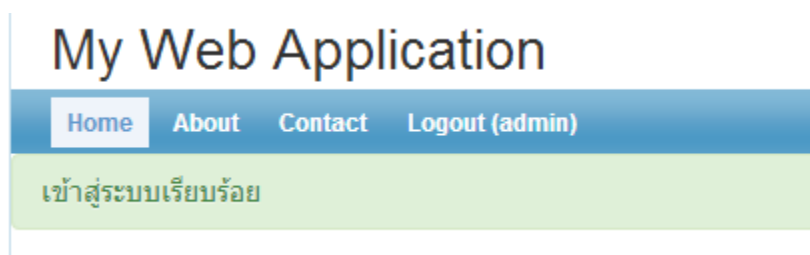
ตัวอย่างต่อไปนี้เป็นกำหนัดสำหรับการเข้าสู่ระบบ โดยมีการปรับปรุงไฟล์ต่างๆ ดังนี้
ส่วนของไฟล์ protected/components/UserIdentity.php ปรับแก้ส่วนคำสั่ง IF

```
if(!isset($users[$this->username])){  
    $this->errorCode=self::ERROR_USERNAME_INVALID;  
    Yii::app()->user->setFlash('danger', "Username ไม่ถูกต้อง");  
}elseif($users[$this->username]!=$this->password){  
    $this->errorCode=self::ERROR_PASSWORD_INVALID;  
    Yii::app()->user->setFlash('danger', "Password ไม่ถูกต้อง");  
}else{  
    $this->errorCode=self::ERROR_NONE;  
    Yii::app()->user->setFlash('success', "เข้าสู่ระบบเรียบร้อยแล้ว");  
}  
return !$this->errorCode;
```

ส่วนของไฟล์ที่เป็น Layout ในที่นี้คือ protected/views/layouts/main.php เพิ่ม Code บริเวณที่ต้องการให้แสดง Flash Message ดังต่อไปนี้

```
<?php  
foreach(Yii::app()->user->getFlashes() as $key => $message) {  
    echo '<div class="alert alert-' . $key . '">' . $message . "</div>\n";  
}  
?>
```

เมื่อทำการเข้าสู่ระบบสำเร็จจะแสดงผลข้อมูลเป็นลักษณะดังนี้



เมื่อทำการเข้าสู่ระบบไม่สำเร็จจะแสดงผลข้อมูลเป็นลักษณะดังนี้

My Web Application

[Home](#) [About](#) [Contact](#) [Login](#)

หน้าหลัก » Login

Username ไม่ถูกต้อง

การสร้าง AJAX Button

AJAX หรือเรียกว่า Asynchronous JavaScript and XML เป็นลักษณะของการเขียนโปรแกรมแบบใหม่โดยใช้คุณสมบัติของ JavaScript ที่ทำงานในฝั่ง Client แล้ว Request ไปประมวลผลกับ Script ในฝั่ง Server แล้วนำข้อมูลที่ได้หลังจากการประมวลผลมา Update บริเวณที่ต้องการได้ โดยที่หน้าเว็บไซต์ไม่ Refresh ใหม่ ทำให้มีความรวดเร็วในการใช้งาน Application แต่ข้อเสียของ AJAX นั้นก็คือเรื่องของ SEO จึงไม่เหมาะกับการทำ Application ลักษณะ CMS

สร้าง actionAjaxButton() ใน Controller สำหรับสร้างฟอร์ม

```
public function actionAjaxButton() {  
    $this->render('ajaxbutton');  
}
```

สร้าง View สำหรับแสดงผลฟอร์ม ajaxbutton.php พร้อมปุ่มส่งข้อมูลแบบ AJAX โดยที่ส่งตัวแปร item เข้าไปประมวลผลที่ site/ajaxrequest

```
<select id="myData" name="testData">  
    <option value="test1">Test1</option>  
    <option value="test2">Test2</option>  
    <option value="test3">Test3</option>  
</select>  
<?php  
echo CHtml::ajaxButton ('กดลองดู',  
    $this->createUrl('site/ajaxrequest'),  
    array(  
        'data' => array('item'=> 'js: $("#myData option:selected").val()'),  
        'update' => '#dataToUpdate'  
    )  
);  
<?>  
<div id="dataToUpdate"></div>
```

เมื่อแสดงผลจะได้แบบนี้



The screenshot shows a web form with a dropdown menu on the left containing the text 'Test1' and a small downward arrow. To the right of the dropdown is a rectangular button with the Thai text 'กดลองดู' (Click to try).

จะเห็นว่าเมื่อกดปุ่ม กดลองดู จะส่งข้อมูลไปประมวลผลที่ site/ajaxrequest ดังนั้นใน site Controller ก็จะสร้าง actionAjaxRequest() ด้วย

```
public function actionAjaxRequest(){  
    echo ''. $_GET['item'].'<br />';  
    echo date("Y-m-d H:i:s");  
}
```

ทดลองกดปุ่มเพื่อส่งข้อมูลจะได้ผลดังนี้

Test1 ▼

ทดลองดู

test1
2014-01-20 04:37:04

จะเห็นว่า มี Item ที่เลือก พร้อมกับวันเวลาที่ได้เขียนโปรแกรมไว้ใน actionAjaxRequest() นั่นเอง

การสร้าง AJAX Link

ได้เห็น Concept ของ AJAX Button กันแล้วคราวนี้มาลองกับ Link กันบ้างนะครับ โดย Concept ก็ยังคงคล้ายๆ กันอยู่ครับ แต่จะเปลี่ยนการส่งเป็นแบบ POST แทนนะครับ

สร้าง actionAjaxLink() ใน Controller สำหรับสร้างฟอร์ม

```
public function actionAjaxLink(){
    $this->render('ajaxlink');
}
```

สร้าง View สำหรับสร้างฟอร์มชื่อ ajaxlink.php พร้อม Link สำหรับกดโดยจะใช้ Class ของ Bootstrap

```
<?php
echo CHtml::ajaxLink(
    "ข้อความลึกลับ",
    Yii::app()->createUrl('site/ajaxrequest2'),
    array( // ajaxOptions
        'type' => 'POST',
        'beforeSend' => "function(request){
            alert('ก่อนส่งนะครับ')
        }",
        'data' => array('val1' => '1', 'val2' => '2'),
        'update'=>'#dataToUpdate',
    ),
    array( //htmlOptions
        'href' => Yii::app()->createUrl('site/ajaxrequest2'),
        'class' =>'btn btn-success'
    )
);
?>
<div id="dataToUpdate"></div>
```

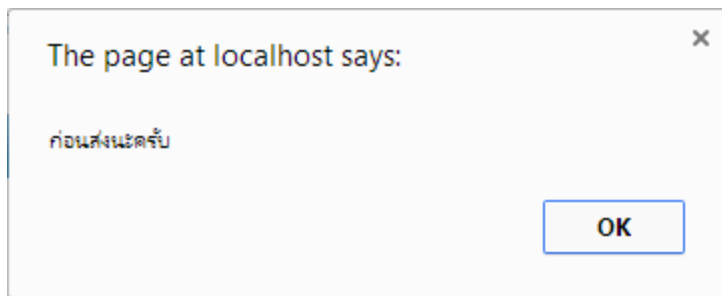
เมื่อแสดงผลจะได้แบบนี้ครับ

ข้อความลึกลับ

จะเห็นว่าเมื่อคลิกนั้นจะส่งข้อมูลไปประมวลผลที่ site/ajaxrequest2 แล้วไปสร้าง actionAjaxRequest2() ใน SiteController ดังนี้นะครับ

```
public function actionAjaxRequest2() {  
    echo $_POST['val1'].'<br />';  
    echo $_POST['val2'].'<br />';  
    echo date("Y-m-d H:i:s");  
}
```

จะเห็นว่าเราส่งข้อมูลมาแบบ POST โดยส่ง val1 และ val2 เข้ามา และเขียนโปรแกรมในการแสดงผลตัวแปรทั้ง 2 พร้อม วันเดือนปีและเวลาด้วย โดยทดลองกดปุ่มลิงค์ จะแสดงผลดังนี้ โดยจะปรากฏ Alert ก่อนส่ง เมื่อกดปุ่ม OK ถึงจะส่งข้อมูลครับ



จากนั้นจะแสดงข้อมูลที่เกิดจากการประมวลผลแล้วส่งมาแสดงผลที่
<div id="dataToUpdate"></div> ดังนี้

ข้อความลิงค์

1
2
2014-01-20 05:06:19

การสร้าง Facebook Like Button

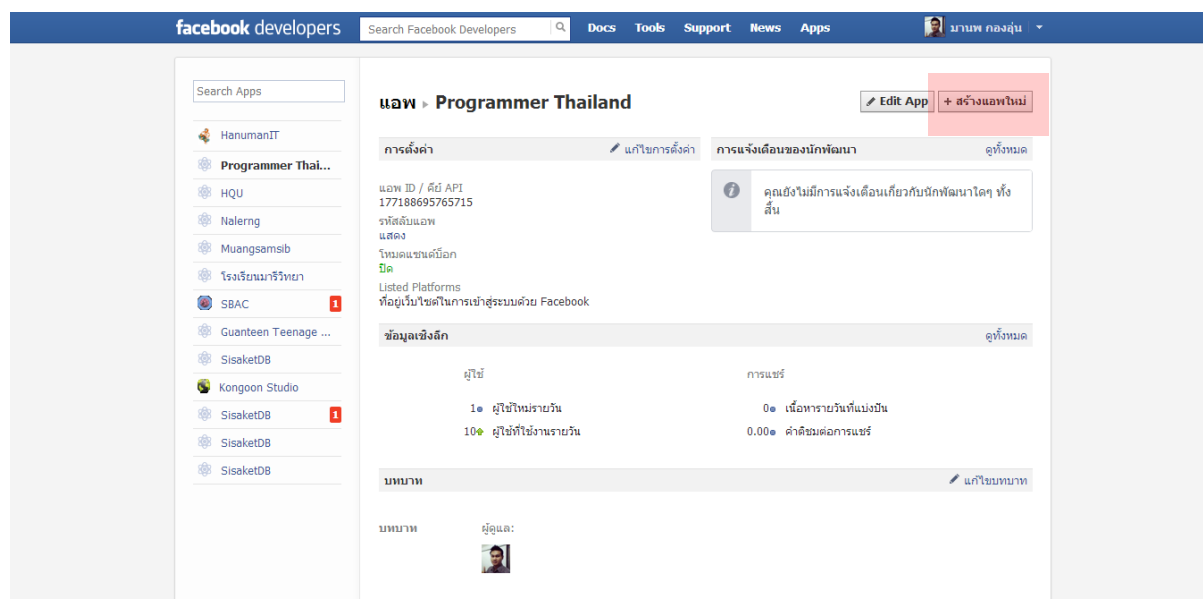
สำหรับ Application ที่ต้องการให้สมาชิกหรือบุคคลทั่วไปให้สามารถกดปุ่ม Like หรือชอบหน้านี้แล้วให้ไปแสดงสถานะใน Facebook ของคนๆ นั้น เราเรียกว่าการกดไลค์ โดยเราสามารถนำมาแสดงผลใน View ของเราได้ดังนี้

สร้าง Application ใน Facebook

ก่อนอื่นเราต้องไปสร้าง Application ใน Facebook ก่อนนะครับ โดยเข้าไปที่ URL <https://developers.facebook.com/>



แล้วคลิกที่เมนู Apps

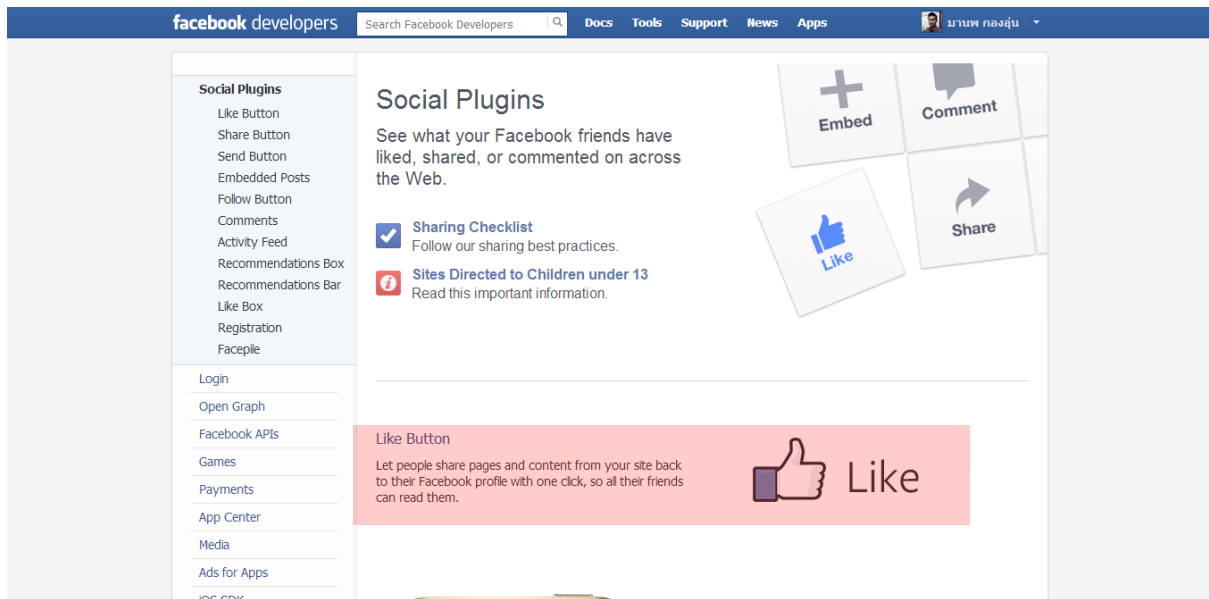


จากนั้นคลิกที่ปุ่ม สร้างแอปใหม่ เพื่อสร้าง Application ใน Facebook ใหม่ ให้ทำตามขั้นตอน โดยขั้นตอนนี้จะไม่ขอล่าวถึงรายละเอียดครับ

สร้าง Facebook Like Button

ในการสร้าง Facebook Like Button นั้นไปที่ URL

<https://developers.facebook.com/docs/plugins/> เพื่อเรียกการสร้าง Plugin แบบต่างๆ



ในที่นี้เลือก Like Button

Like Button

Let people share pages and content from your site back to their Facebook profile with one click, so all their friends can read them.




แล้วกำหนดค่าได้ตามต้องการ

Like Button

The Like button is the quickest way for people to share content with their friends.

A single click on the Like button will 'like' pieces of content on the web and share them on Facebook. You can also display a Share button next to the Like button to let people add a personal message and customize who they share with.

URL to Like	Width
<input type="text" value="http://www.programmerthailand.com"/>	<input type="text" value="The pixel width of the plugin"/>
Layout	Action Type
<input type="text" value="standard"/>	<input type="text" value="like"/>
<input checked="" type="checkbox"/> Show Friends' Faces	<input checked="" type="checkbox"/> Include Share Button

 155 คนถูกใจสิ่งนี้ เป็นคนแรกจากเพื่อนของคุณ

Get Code

เมื่อตั้งค่าเสร็จเรียบร้อยแล้วให้กดปุ่ม Get Code แล้วเลือก app ID ให้เรียบร้อย

Your Plugin Code

HTML5

XFBML

IFRAME

URL

This script uses the app ID of your app

Programmer Thailand

Include the [JavaScript SDK](#) on your page once, ideally right after the opening <body> tag.

```
<div id="fb-root"></div>
<script>(function(d, s, id) {
  var js, fjs = d.getElementsByTagName(s)[0];
  if (d.getElementById(id)) return;
  js = d.createElement(s); js.id = id;
  js.src =
  "//connect.facebook.net/th_TH/all.js#xfbml=1&appId=177188695765715";
  fjs.parentNode.insertBefore(js, fjs);
})(document, 'script', 'facebook-jssdk');</script>
```

Place the code for your plugin wherever you want the plugin to appear on your page.

```
<div class="fb-like" data-
href="http://www.programmerthailand.com" data-layout="standard"
data-action="like" data-show-faces="true" data-shares="true">
```


จากนั้นนำ Code ทั้ง 2 ส่วนไปไว้ใน view ที่เราต้องการได้เลย ในที่นี้เอาไว้ใน
protected/views/site/index.php

```
22 <div id="fb-root"></div>
23 <script>(function(d, s, id) {
24     var js, fjs = d.getElementsByTagName(s)[0];
25     if (d.getElementById(id)) return;
26     js = d.createElement(s); js.id = id;
27     js.src = "//connect.facebook.net/th_TH/all.js#xfbml=1&appId=177188695765715";
28     fjs.parentNode.insertBefore(js, fjs);
29 })(document, 'script', 'facebook-jssdk');</script>
30
31 <div class="fb-like" data-href="http://www.programmerthailand.com" data-layout="standard"
```

เมื่อเรียกใช้งานจะแสดงผลหน้า Application ดังนี้

ร่วมสมัครสมาชิก

ขอเชิญเพื่อนๆ สมัครสมาชิกและมีส่วนร่วมกับเรานะครับ

👁 ถูกใจ

แชร์

155 คนถูกใจสิ่งนี้ เป็นคนแรกจากเพื่อนของคุณ

Welcome to *My Web Application*

Congratulations! You have successfully created your Yii application.

บทที่ 6 ว่าด้วยเรื่องของรายงาน

การแสดงผลข้อมูลแบบ JSON

การแสดงผลข้อมูลแบบ JSON นิยมในการสร้างข้อมูลเพื่อทำกราฟ หรือแม้แต่การทำ Web Service ในที่นี้จะเป็นการแสดงผลข้อมูลแบบ JSON เพื่อส่งให้ jqWidgets นำไปแสดงผลเป็นกราฟ โดยสร้าง action Json() เพื่อแสดงผลข้อมูลดังนี้

```
public function actionJson(){  
  
    $model= Yii::app()->db->createCommand('SELECT SUM(money) as money,acc_type  
        FROM account WHERE user_id='.Yii::app()->user->getId().''  
        GROUP BY acc_type')->queryAll();  
    header("Content-type: application/json");  
    echo '{"data":'.CJSON::encode($model).'}';  
  
}
```

เมื่อเรียก URL mycontroller/json ก็จะปรากฏข้อมูลดังนี้

```
{"data":[{"money":"30000","acc_type":"\u0e23\u0e31\u0e1a"}, {"money":"5735","acc_type":"\u0e08\u0e48\u0e32\u0e22"}]}
```

การสร้างกราฟด้วย jqWidgets



jqWidgets เป็น JavaScript Library ที่รวบรวมการแสดงผลต่างๆ หนึ่งในนั้นคือกราฟ โดยการแสดงผลกราฟนั้นก็มากมายหลายรูปแบบ รูปแบบหนึ่งที่น่าสนใจนั้นก็คือการแสดงกราฟแบบวงกลม หรือแบบ Pie นั่นเอง

สิ่งที่น่าสนใจของ jqWidgets นั้นเป็นเรื่องของการดึงข้อมูลแบบ JSON มาแสดงเป็นกราฟ โดยเมื่อเราเตรียมข้อมูลแบบ JSON ใน action หนึ่งแล้ว เราสามารถเรียก URL เพื่อนำข้อมูลนั้นมาแสดงได้ทันที

สามารถดูรายละเอียดเพิ่มเติมได้ที่เว็บไซต์ <http://www.jqwidgets.com/>

ในส่วนของ Controller ทดลองสร้าง Action ใหม่ใน protected/Controllers/SiteController.php สร้าง actionChart()

```
public function actionChart() {  
    $this->render('chart');  
}
```

ในส่วนของ View สร้างไฟล์ protected/views/site/chart.php โดยให้มีข้อมูลดังนี้

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
<link rel="stylesheet"
      href="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/styles/jqx.base.css"
      type="text/css" />
<script type="text/javascript"
src="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/jqxcore.js"></script>
<script type="text/javascript"
src="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/jqxchart.js"></script>
<script type="text/javascript"
src="http://www.jqwidgets.com/jquery-widgets-demo/jqwidgets/jqxdata.js"></script>
```

```
<script type="text/javascript">
$(document).ready(function() {
    // prepare chart data as an array
    var source =
        {
            datatype: "json",
            datafields: [
                {name: 'money', type: 'number'},
                {name: 'acc_type', type: 'string'}
            ],
            url: 'http://[redacted]/json'
        };

    var dataAdapter = new $.jqx.dataAdapter(source, {async: false, autoBind: true,
        loadError: function(xhr, status, error) {
            alert('Error loading "' + source.url + '" : ' + error);
        }});
});
```

```
var settings = {
    title: "รายการรับจ่าย",
    description: "(กราฟสรุปรับจ่าย)",
    enableAnimations: true,
    showLegend: true,
    legendLayout: {left: 500, top: 140, width: 300, height: 200, flow: 'vertical'},
    padding: {left: 5, top: 5, right: 5, bottom: 5},
    titlePadding: {left: 0, top: 0, right: 0, bottom: 10},
    source: dataAdapter,
    colorScheme: 'scheme03',
    seriesGroups:
        [
            {
                type: 'pie',
                showLabels: true,
                series:
                    [
                        {
                            dataField: 'money',
                            displayText: 'acc_type',
                            labelRadius: 120,
                            initialAngle: 15,
                            radius: 95,
                            centerOffset: 0,
                            formatSettings: {suffix: ' บาท', decimalPlaces: 1}
                        }
                    ]
            }
        ]
};
```

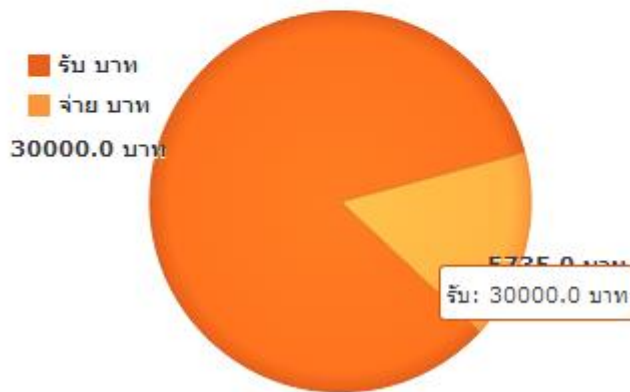
```
// setup the chart
$('#jqxChart').jqxChart(settings);

});
</script>

<div id='jqxChart' style='width: 100%; height: 400px; position: relative; left: 0px;
top: 0px;'>
</div>
```

เมื่อเรียก URL เพื่อแสดง `site/chart` จะได้การแสดงผลดังนี้

รายการรับจ่าย
(กราฟสรุปรับจ่าย)



การแสดงกราฟด้วย Highcharts JS



Highcharts JS ก็เป็น JavaScript อีกตัวหนึ่งที่น่าสนใจในการทำแผนภูมิรูปแบบต่างๆ โดยสามารถทดลองใช้โดยการคลิกที่เมนู Demo

ในส่วนของ Controller ทดลองสร้างกราฟแท่งโดยสร้าง Action ใน Controller `protected/controllers/SiteController.php` โดยสร้าง `actionChart2()`

```
public function actionChart2() {  
    $this->render('chart2');  
}
```

ในส่วนของ View สร้างไฟล์ `protected/views/site/chart2.php` โดยให้มีข้อมูลดังนี้

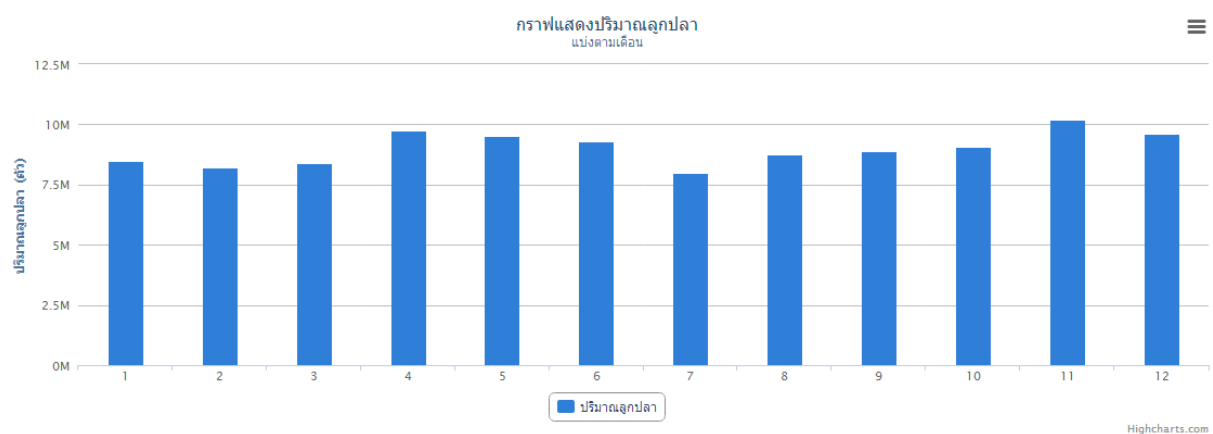
```
<?php
$data = array('1' => 8468193,
    '2' => 8196690,
    '3' => 8389832,
    '4' => 9755049,
    '5' => 9503702,
    '6' => 9281599,
    '7' => 7988226,
    '8' => 8757747,
    '9' => 8875358,
    '10' => 9087173,
    '11' => 10196143,
    '12' => 9598499);
$mymonth = "";
$mydata = "";
$i = 1;
$j = count($data);
foreach ($data as $k => $v) {
    if ($i < $j) {
        $c = ",";
    } else {
        $c = "";
    }
    $mymonth .= $k . $c;
    $mydata .= $v . $c;
    $i++;
}
?>

<script src="http://code.highcharts.com/highcharts.js"></script>
<script src="http://code.highcharts.com/modules/exporting.js"></script>

<div id="chart_container" style="min-width: 310px; height: 400px; margin: 0 auto"></div>
<script>
$(function() {
    $('#chart_container').highcharts({
        chart: {
            type: 'column'
        },
        title: {
            text: 'กราฟแสดงปริมาณลูกปลา'
        },
        subtitle: {
            text: 'แบ่งตามเดือน'
        },
        xAxis: {
            categories: [<?php echo $mymonth; ?>]
        },
        yAxis: {
            min: 0,
            title: {
                text: 'ปริมาณลูกปลา (ตัว)'
            }
        }
    });
});
```

```
        tooltip: {
            headerFormat: '<span style="font-size:10px">{point.key}</span><table>',
            pointFormat: '<tr><td style="color:{series.color};padding:0">{series.name}: </td>' +
                '<td style="padding:0"><b>{point.y:.1f} ล้านบาท</b></td></tr>',
            footerFormat: '</table>',
            shared: true,
            useHTML: true
        },
        plotOptions: {
            column: {
                pointPadding: 0.2,
                borderWidth: 0
            }
        },
        series: [{
            name: 'ปริมาณลูกปลา',
            data: [<?php echo $mydata; ?>]
        }]
    });
</script>
```

เมื่อเรียก URL เพื่อแสดง page site/chart2 จะแสดงผลดังนี้



การ Export ข้อมูลแบบ CSV

การ Export หรือส่งออกข้อมูลจากหน้า admin นั้นจะใช้ Extension เสริมของ Yii Framework ที่ชื่อว่า ExportTableGridBehavior เป็นตัวช่วย โดยสามารถดาวน์โหลดได้ที่ <http://www.yiiframework.com/extension/exportablegridbehavior/>

สำหรับการติดตั้งและตั้งค่านั้นเมื่อดาวน์โหลดมาแล้วให้นำมาเก็บไว้ที่ Folder `protected/components/ExportableGridBehavior.php` จากนั้นให้เขียนโปรแกรมในส่วนต่างๆ ดังนี้

ในส่วนของ Controller ให้เพิ่ม function behaviors() ดังนี้

```
public function behaviors() {  
    return array(  
        'exportableGrid' => array(  
            'class' => 'application.components.ExportableGridBehavior',  
            'filename' => 'PostsWithUsers.csv',  
            'csvDelimiter' => ';', //i.e. Excel friendly csv delimiter  
        ));  
}
```

และใน function admin() ให้ปรับแก้ดังนี้

```

public function actionAdmin() {
    $model = new Students('search');
    $model->unsetAttributes(); // clear any default values
    if (isset($_GET['Students']))
        $model->attributes = $_GET['Students'];
    if ($this->isExportRequest()) {
        $this->exportCSV(array('Student:'), null, false);
        $this->exportCSV($model, array_keys($model->attributeLabels()), false, 3);
        $this->exportCSV($model->search(), array('id','level.level',
            'prefix.prefix',
            'first_name',
            'last_name',
            'email',
            'tel',
            'parentPrefix.prefix',
            'parent_first_name',
            'parent_last_name',
            'parent_tel',
            'parent_email',
            'created',
            'address',
            'tambon.tambon_name',
            'tambon.district.district_name',
            'tambon.province.province_name',
            'status',));
    }
    $this->render('admin', array(
        'model' => $model,
    ));
}

```


ในส่วนของ View ให้รับแก้ admin.php ดังนี้

```


<?php
$gridWidget=$this->widget('bootstrap.widgets.TbGridView', array(
    'id' => 'students-grid',
    'dataProvider' => $model->search(),
    'filter' => $model,
    'columns' => array(
        'id',
        'level.level',
        'prefix.prefix',
        'first_name',
        'last_name',
        'email',
        array(
            'class' => 'bootstrap.widgets.TbButtonColumn',
        ),
    ),
));
$this->renderExportGridButton($gridWidget,'Export Grid Results',
    array('class'=>'btn btn-info pull-right'));
?>

```





การสร้างกราฟด้วย RGraph Extension

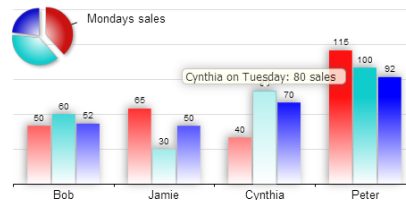
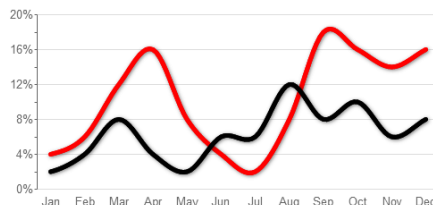


RGraph: *HTML5 charts library*
RGraph produces interactive charts using JavaScript and the HTML5 canvas tag

Help support
RGraph:


Home Download Examples Documentation Support Donate Blog Reference About

Please share with your friends:
   



Use *HTML5 charts to build faster websites*

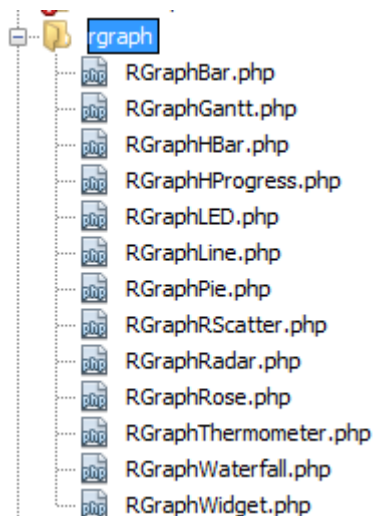
RGraph is a HTML5 canvas and JavaScript based library built for web charts and supports over twenty different types of visualisation. Using the <canvas> tag, RGraph creates these "HTML5 charts" inside the web browser using JavaScript, meaning quicker pages and less web server load. This leads to smaller page sizes, lower costs and faster websites.

Open *Source and free to use!*



RGraph is truly open and free to use for any purpose under the **MIT license**. The full source is included in the download and you can edit it as you need.

RGraph Extension เป็นส่วนเสริมที่สร้างขึ้นเพื่อการแสดงผลกราฟ โดยมีรูปแบบให้เลือกมากมายเช่นกัน โดยมีรูปแบบให้เลือกดังนี้



สามารถดาวน์โหลด Extension นี้ได้จาก URL

<http://www.yiiframework.com/extension/rgraph>

เมื่อดาวน์โหลดเรียบร้อยแล้วให้นำไปวางไว้ที่ `protected/extensions/rgraph/`

จากนั้นดาวน์โหลด Library ของ RGraph จากเว็บไซต์ <http://www.rgraph.net/> แล้วนำไปวางไว้ที่ `protected/vendors/rgraph/`

สำหรับวิธีการใช้งานสามารถเรียก Widget ของ Graph ได้เลย

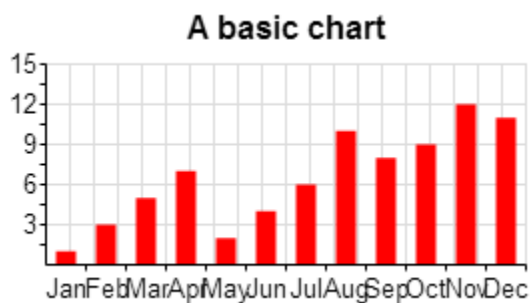
ในส่วนของ Controller protected/controllers/SiteController.php สร้าง Action
actionRGraphBar()

```
public function actionRGraphBar() {  
    $this->render('rgraphbar');  
}
```

ในส่วนของ View สร้างไฟล์ protected/views/site/rgraphbar.php แล้วใส่ข้อมูลดังนี้

```
<?php  
$this->widget('ext.rgraph.RGraphBar', array(  
    'data' => array(1, 3, 5, 7, 2, 4, 6, 10, 8, 9, 12, 11),  
    'options' => array(  
        'chart' => array(  
            'gutter' => array(  
                'left' => 35,  
            ),  
            'colors' => array('red'),  
            'title' => 'A basic chart',  
            'labels' => array(  
                'Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'  
            ),  
        ),  
    ),  
));
```

เมื่อเรียก URL เพื่อแสดงพลา site/rgraphbar จะได้การแสดงผลดังนี้



การสร้างกราฟด้วย Google Chart Extension

สำหรับ Google Chart นั้นก็มีผู้ทำ Extension ไว้โดยตั้งชื่อว่า googlechart ซึ่งสามารถดาวน์โหลดได้ที่ <http://www.yiiframework.com/extension/googlechart/> หลังจากดาวน์โหลดเสร็จเรียบร้อยแล้วให้นำไปวางใน protected/extensions/Hzl/

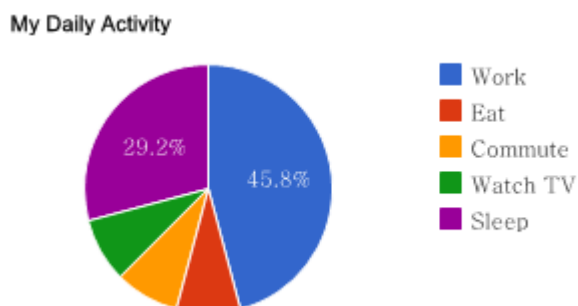
Pie Chart

มาเริ่มต้นกับ Pie Chart กันก่อนนะครับ โดยเปิด View ที่ต้องการสร้าง แล้วเขียนคำสั่งดังนี้

```
<?php
//very useful google chart
$this->widget('ext.Hzl.google.HzlVisualizationChart',
array('visualization' => 'PieChart',
      'data' => array(
        array('Task', 'Hours per Day'),
        array('Work', 11),
        array('Eat', 2),
        array('Commute', 2),
        array('Watch TV', 2),
        array('Sleep', 7)
      ),
      'options' => array('title' => 'My Daily Activity')));

$this->widget('ext.Hzl.google.HzlVisualizationChart',
array('visualization' => 'LineChart',
      'data' => array(
        array('Task', 'Hours per Day'),
        array('Work', 11),
        array('Eat', 2),
        array('Commute', 2),
        array('Watch TV', 2),
        array('Sleep', 7)
      ),
      'options' => array('title' => 'My Daily Activity')));
?>
```

ทดลองเรียก URL จะได้กราฟดังนี้



Gauge

ตัวอย่างที่ 2 การแสดงผลแบบมาตรวัดหรือ Gauge โดยเปิด View ที่ต้องการสร้างแล้วเขียน Code ดังนี้

```
<?php
    $this->widget('ext.Hzl.google.HzlVisualizationChart',
array('visualization' => 'Gauge', 'packages' => 'gauge',
    'data' => array(
        array('Label', 'Value'),
        array('Memory', 80),
        array('CPU', 55),
        array('Network', 68),
    ),
    'options' => array(
        'width' => 400,
        'height' => 120,
        'redFrom' => 90,
        'redTo' => 100,
        'yellowFrom' => 75,
        'yellowTo' => 90,
        'minorTicks' => 5
    )
));
?>
```

ทดลองเรียก URL จะแสดงผลดังนี้



Line Chart

ตัวอย่างที่ 3 เป็นกราฟแบบเส้นหรือ Line Chart โดยเปิด View ที่ต้องการสร้าง แล้วเขียน Code ดังนี้

```
<?php
$this->widget('ext.Hzl.google.HzlVisualizationChart', array('visualization' =>
'LineChart',
    'data' => array(
        array('Year', 'Sales', 'Expenses'),
        array('2004', 1000, 400),
        array('2005', 1170, 460),
        array('2006', 660, 1120),
        array('2007', 1030, 540),
    )
));
```

```
    ),  
    'options' => array(  
        'title' => 'My Company Performance2',  
        'titleTextStyle' => array('color' => '#FF0000'),  
        'vAxis' => array(  
            'title' => 'Scott vAxis',  
            'gridlines' => array(  
                'color' => 'transparent' //set grid line transparent  
            ),  
            'hAxis' => array('title' => 'Scott hAixs'),  
            'curveType' => 'function', //smooth curve or not  
            'legend' => array('position' => 'bottom'),  
        ),  
    );  
?>
```

ทดลองเรียก URL เพื่อแสดงผลจะแสดงผลได้ดังนี้



บทที่ 7 กรณีตัวอย่าง

การสร้าง Drop Down List เลือกจังหวัดโชว์อำเภอ เลือก อำเภอโชว์ตำบล แบบ Dependence (AJAX)

การเลือกจังหวัดโชว์อำเภอ เลือกอำเภอโชว์ตำบล เป็นการเลือกลักษณะที่เป็น Dependence Drop Down List โดยใช้คุณสมบัติ AJAX ซึ่งจะไม่ทำให้หน้าเว็บไซต์ Refresh เพื่อป้องกันข้อมูลหายในขณะกรอกฟอร์ม โดยในตัวอย่างนี้เป็นการเลือกจังหวัด แล้วจะมีรายการอำเภอในจังหวัดนั้นแสดงใน Drop Down List ของอำเภอ และเมื่อเรียกอำเภอก็จะมีรายการของตำบลนั้นแสดงใน Drop Down List ของตำบล จากนั้นเมื่อบันทึกข้อมูลในตารางก็จะเก็บเอาเพียงแค่ ID ของตำบลไปเก็บเท่านั้น

ก่อนอื่น ไปดู Data Dictionary หรือพจนานุกรมข้อมูลกันก่อนนะครับ ว่าโครงสร้างตารางนั้นเป็นอย่างไร

provinces

Table comments: จังหวัด

Column	Type	Null	Default	Comments
id	int(11)	No		
province_name	varchar(60)	No		ชื่อจังหวัด
province_name_en	varchar(60)	No		ชื่อจังหวัดอังกฤษ

districts

Table comments: อำเภอ

Column	Type	Null	Default	Comments
id	int(11)	No		
province_id	int(11)	No		รหัสจังหวัด

district_name	varchar(60)	No		ชื่ออำเภอ
---------------	-------------	----	--	-----------

tambons

Table comments: ตำบล

Column	Type	Null	Default	Comments
id	int(11)	No		
district_id	int(11)	No		รหัสอำเภอ
province_id	int(11)	No		รหัสจังหวัด
tambon_name	varchar(60)	No		ชื่อตำบล

ในส่วนของ Controller ให้สร้างเรียกข้อมูลจังหวัดขึ้นมาก่อนครับ แล้วส่งให้ View เพื่อแสดงผลรายการของจังหวัด ส่งให้ View ผ่านตัวแปร \$province

```
$province = Provinces::model()->findAll(array('order'=>'province_name ASC'));

$this->render('index', array(
    'province' => $province,
));
```

สร้าง Function เพิ่มเติมเพื่อเรียกข้อมูลเมื่อมีการเลือกจังหวัด โดยเลือกข้อมูลจาก ID จังหวัดที่ส่งมา

```
public function actionLoadDistrict() {
    $data = Districts::model()->findAll('province_id=:province_id',
        array(':province_id' => (int) $_POST['province_id']));

    $datalist = CHtml::listData($data, 'id', 'district_name');

    echo "<option value=''>เลือกอำเภอ</option>";
    foreach ($datalist as $value => $district_name)
        echo CHtml::tag('option',
            array('value' => $value), CHtml::encode($district_name), true);
}
```

สร้าง Function เพิ่มเติมเพื่อเรียกข้อมูลเมื่อมีการเลือกอำเภอ โดยเลือกข้อมูลจาก ID ของอำเภอที่ส่งมา

```
public function actionLoadSubdistrict() {  
    $data = Tambons::model()->findAll('district_id=:district_id',  
        array(':district_id' => (int) $_POST['district_id']));  
  
    $datalist = CHtml::listData($data, 'id', 'tambon_name');  
  
    echo "<option value=''>เลือกตำบล</option>";  
    foreach ($datalist as $value => $tambon_name)  
        echo CHtml::tag('option',  
            array('value' => $value), CHtml::encode($tambon_name), true);  
}
```

ในส่วนของ View ให้ทำการแสดงข้อมูล จังหวัดที่ส่งมาจาก Controller ผ่านตัวแปร \$province โดยเมื่อมีการเลือกรายการให้ส่งไปที่ Action loaddistrict จากนั้นส่งค่ากลับมาที่ #district

```
<div class="control-group">  
    <?php echo CHtml::label(Yii::t('demo', 'จังหวัด '), 'province',  
        array('class' => 'control-label')); ?>  
    <div class="controls">  
        <?php  
            echo CHtml::dropDownList('province_id', '',  
                CHtml::listData($province, 'id', 'province_name'), array(  
                    'prompt' => 'เลือกจังหวัด',  
                    'id' => 'province',  
                    'ajax' => array(  
                        'type' => 'POST',  
                        'url' => CController::createUrl('loaddistrict'),  
                        'update' => '#district',  
                        'data' => array('province_id' => 'js:this.value'),  
                    )  
                )  
            );  
        ?>  
    </div>  
</div>
```

เมื่อมีการเลือกรายการอำเภอให้ส่งไปที่ Action loadsubdistrict จากนั้นส่งค่ากลับมาที่ #subdistrict

```
<div class="control-group">
    <?php echo CHtml::label(Yii::t('demo', 'อำเภอ '), 'district',
        array('class' => 'control-label')); ?>
    <div class="controls">
        <?php
        echo CHtml::dropDownList('district_id', '', array(), array(
            'prompt' => 'เลือกอำเภอ',
            'id' => 'district',
            'ajax' => array(
                'type' => 'POST',
                'url' => CController::createUrl('loadsubdistrict'),
                'update' => '#subdistrict',
                'data' => array('district_id' => 'js:this.value'),
            )
        ));
    ?>
    </div>
</div>
```

สุดท้ายก็จะสามารถเลือกรายการของตำบลได้

```
<?php
echo $form->dropDownListRow($model, 'tambon_id', array(), array(
    'prompt' => 'เลือกตำบล',
    'id' => 'subdistrict',
));
?>
```

เรียก URL เพื่อทดสอบ จะเห็นการแสดงผลดังนี้

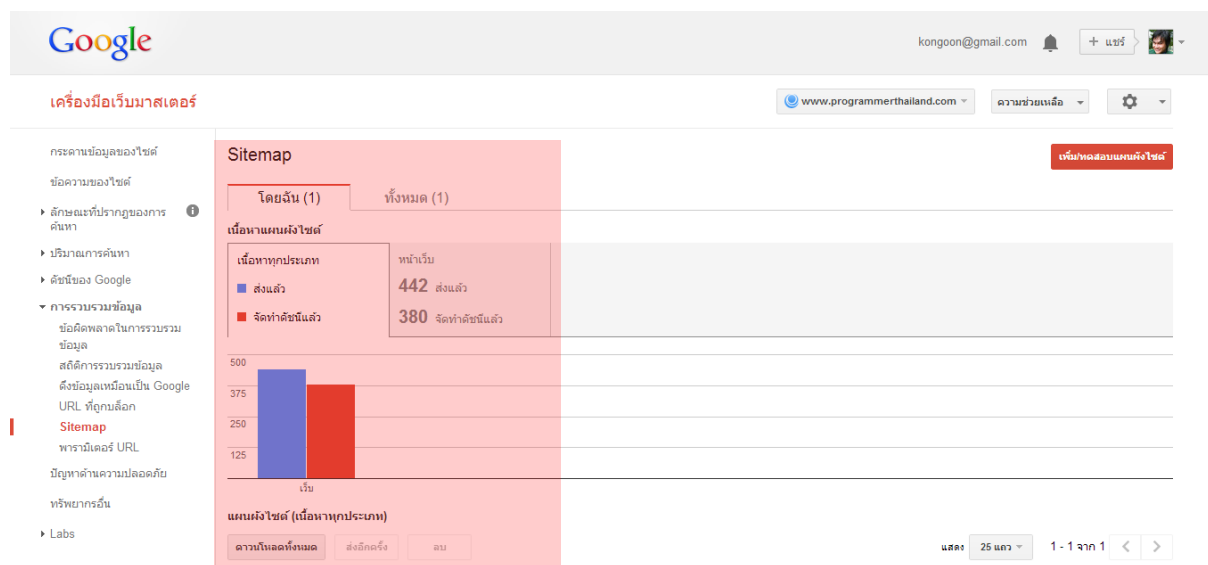
จังหวัด	<input type="text" value="กระบี่"/>
อำเภอ	<input type="text"/>
ตำบล *	<input type="text"/>

เมื่อทดลองเลือกจังหวัด จะโชว์รายการอำเภอในจังหวัดนั้น และเมื่อเรียกอำเภอในจังหวัดนั้นจะโชว์รายการตำบลในอำเภอที่เลือกนั้น ดังนี้

จังหวัด	กรุงเทพมหานคร	▼
อำเภอ	บางกะปิ	▼
ตำบล *	ห้วยหมาก	▼

การสร้าง sitemap.xml ใน Yii Framework ส่งให้ Google Webmaster Tool และ Search Engine

Sitemap ใน Google Webmaster Tool ถือได้ว่ามีความสำคัญอย่างหนึ่งซึ่งเป็นส่วนหนึ่งของการทำ Search Engine Optimization หรือ SEO เป็นการส่ง Link ต่างๆ เข้ามาเพื่อการจัดทำ Index หรือ ดัชนีเว็บไซต์เราใน Google นั้นเอง ดังตัวอย่างต่อไปนี้



ในส่วนของ Controller สร้าง Action actionSitemapXML() เพื่อแสดงผลในรูปแบบของ XML ที่ protected/controllers/SiteController.php

```
public function actionSitemapXML() {  
    $list = array();  
  
    $this->populateSitemap($list);  
  
    header('Content-Type: application/xml');  
    $this->renderPartial('sitemapxml', array('list' => $list));  
}
```

และสร้าง Function populateSitemap() สำหรับเป็นรายการโดยสามารถดึงมาจากฐานข้อมูลได้ดังนี้

```
public function populateSitemap(&$list) {  
  
    $vdos = Vdos::model()->findAll();  
    $list[] = array(  
        'loc' => $this->createAbsoluteUrl('/'),  
        'frequency' => 'weekly',  
        'priority' => '1',  
    );  
    $list[] = array(  
        'loc' => $this->createAbsoluteUrl('/site/contact'),  
        'frequency' => 'yearly',  
        'priority' => '0.8',  
    );  
    $list[] = array(  
        'loc' => $this->createAbsoluteUrl('/user/registration'),  
        'frequency' => 'yearly',  
        'priority' => '0.8',  
    );  
    $list[] = array(  
        'loc' => $this->createAbsoluteUrl('/user/login'),  
        'frequency' => 'yearly',  
        'priority' => '0.8',  
    );  
  
    foreach ($vdos as $row) {  
        $list[] = array(  
            'loc' => $this->createAbsoluteUrl('/vdos/vdoview',  
                array('slug' => $row->slug)),  
            'frequency' => 'weekly',  
            'priority' => '0.5',  
        );  
    }  
  
}
```

ในส่วนของ View สร้างไฟล์ protected/views/site/sitemapxml.php โดยมีเนื้อหาดังนี้

```
<?php echo '<?xml version="1.0" encoding="UTF-8"?>' ?>

<urlset
    xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9
        http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd">

    <?php foreach($list as $row): ?>
        <url>
            <loc><?php echo CHtml::encode($row['loc']); ?></loc>
            <changefreq><?php echo $row['frequency']; ?></changefreq>
            <priority><?php echo $row['priority']; ?></priority>
        </url>
    <?php endforeach; ?>

</urlset>
```

ปรับส่วนของ urlManager โดยเพิ่มส่วน Rewrite ใน rules ดังนี้

'sitemap.xml'=>'site/sitemapxml',

```
'urlManager'=>array(
    'urlFormat'=>'path',
    'showScriptName'=>false,
    'caseSensitive'=>false,
    'rules'=>array(
        'sitemap.xml' => 'site/sitemapxml',
        '<controller:\w+>/<id:\d+>'=>'<controller>/view',
        '<controller:\w+>/<action:\w+>/<id:\d+>'=>'<controller>/<action>',
        '<controller:\w+>/<action:\w+>'=>'<controller>/<action>',
        'home'=>' /site/index',
    ),
),
```

ทดสอบเรียก URL <http://localhost/myyiiapp/sitemap.xml> จะได้หน้าตาของ XML ลักษณะดังต่อไปนี้

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼ <urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd">
  ▼ <url>
    <loc>http://www.programmerthailand.com</loc>
    <changefreq>weekly</changefreq>
    <priority>1</priority>
  </url>
  ▼ <url>
    <loc>http://www.programmerthailand.com/site/contact</loc>
    <changefreq>yearly</changefreq>
    <priority>0.8</priority>
  </url>
  ▼ <url>
    ▼ <loc>
      http://www.programmerthailand.com/user/registration
    </loc>
    <changefreq>yearly</changefreq>
    <priority>0.8</priority>
  </url>
  ▼ <url>
    <loc>http://www.programmerthailand.com/user/login</loc>
    <changefreq>yearly</changefreq>
    <priority>0.8</priority>
```

จากนั้นก็สามารส่งเข้าไปใน Search Engine ได้เรียบร้อยแล้วครับ

การทำเว็บไซต์หลายภาษาใน Yii Framework

การทำเว็บไซต์หลายภาษาใน Yii Framework สามารถทำได้ง่ายๆ โดยมีขั้นตอนไม่ที่ขั้นตอนก็สามารถทำได้หลายภาษาไทย โดยการกำหนดค่าภาษาจะสร้างเป็น Session ขึ้นมาเพื่อตรวจสอบว่า ณ ขณะปัจจุบันนั้นใช้ภาษาอะไรอยู่ จากนั้นก็จะทำการ เรียกตัวแปรในภาษานั้นขึ้นมาแสดงที่หน้าเว็บไซต์หรือหน้า Application เราได้ โดยมีขั้นตอนดังต่อไปนี้

เพิ่ม Menu ใน protected/views/layouts/main.php

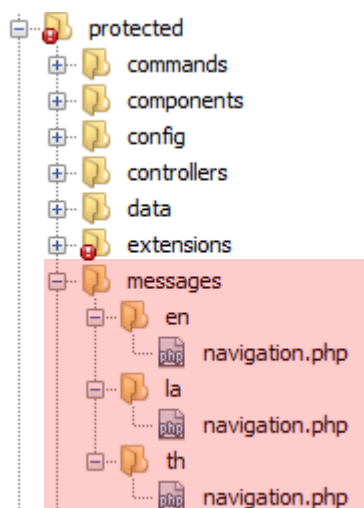
โดยกำหนดค่าให้กับ label โดยเรียก Function ดังนี้ Yii::t('ชื่อไฟล์ภาษา','ชื่อตัวแปร')

```
<?php $this->widget('zii.widgets.CMenu',array(
    'items'=>array(
        array('label'=>Yii::t('navigation','home'),
            'url'=>array('/site/index')),
        array('label'=>Yii::t('navigation','product_type'),
            'url'=>array('productTypes/index')),
        array('label'=>Yii::t('navigation','product'),
            'url'=>array('products/index')),
        array('label'=>Yii::t('navigation','about'),
            'url'=>array('/site/page', 'view'=>'about')),
        array('label'=>Yii::t('navigation','contact'),
            'url'=>array('/site/contact')),
        array('label'=>Yii::t('navigation','login'),
            'url'=>array('/site/login'),
            'visible'=>Yii::app()->user->isGuest),
        array('label'=>'For Administrator','url'=>'#',
            'visible'=>Yii::app()->user->getState('role')=='admin'),
        array('label'=>'Logout ('.Yii::app()->user->name.')',
            'url'=>array('/site/logout'),
            'visible'=>!Yii::app()->user->isGuest),
        array('label'=>'EN','url'=>array('setlanguage','lang'=>'en')),
        array('label'=>'LA','url'=>array('setlanguage','lang'=>'la')),
        array('label'=>'TH','url'=>array('setlanguage','lang'=>'th')),
    ),
)); ?>
```

ทดลองเรียก URL เพื่อแสดงผล เช่น <http://localhost/myyiiapp/> จะเห็นเมนูเกิดขึ้นดังนี้ โดยจะมี Link สำหรับเปลี่ยนภาษา 3 ภาษา นั่นคือ ภาษาอังกฤษ (EN) ภาษาลาว (LA) และภาษาไทย (TH)



สร้าง Folder 3 Folder ใน protected/messages/ คือ en, la และ th



จากนั้นสร้างไฟล์ protected/messages/en/navigation.php โดยมีข้อมูลดังนี้

```
<?php
return array(
    'home'=>'Home',
    'product_type'=>'Product Type',
    'product'=>'Product',
    'about'=>'About',
    'contact'=>'Contact',
    'login'=>'Login',
);
?>
```

สร้างไฟล์ protected/messages/la/navigation.php โดยมีข้อมูลดังนี้

```
<?php
return array(
    'home'=>'ເຮົາ',
    'product_type'=>'ປະເພດ',
    'product'=>'ສິນຄ້າ',
    'about'=>'ກ່ຽວກັບ',
    'contact'=>'ຕິດຕໍ່',
    'login'=>'ເຂົ້າສູ່ລະບົບ',
);
?>
```

สร้างไฟล์ protected/messages/th/navigation.php โดยมีข้อมูลดังนี้

```
<?php
return array(
    'home'=>'หน้าหลัก',
    'product_type'=>'ประเภทสินค้า',
    'product'=>'สินค้า',
    'about'=>'เกี่ยวกับเรา',
    'contact'=>'การติดต่อ',
    'login'=>'เข้าสู่ระบบ',
);
?>
```

ในส่วนของ protected/components/Controller.php เพิ่ม Function init() และ Function actionSetlanguage()

```
public function init(){
    parent::init();

    $session = new CHttpSession;
    $session->open();
    Yii::app()->language =
        isset(Yii::app()->session['sess_lang']) ?
        Yii::app()->session['sess_lang'] : 'en';
}
public function actionSetlanguage($lang = null){
    Yii::app()->session['sess_lang']= strtolower($lang);
    Yii::app()->request->redirect(Yii::app()->user->returnUrl);
}
```

ทดลองเรียกใช้งานโดยกดที่คำว่า EN จะกลายเป็นภาษาอังกฤษ



กดที่คำว่า LA จะกลายเป็นภาษาลาว



กดที่คำว่า TH จะกลายเป็นภาษาไทย



ขั้นตอนง่ายๆ เพียงเท่านี้ก็สามารถเรียกใช้งานเว็บไซต์เราได้หลายภาษาเรียบร้อยแล้วครับ

การ Upload ไฟล์รูปภาพ

ในส่วนของ View _form.php กำหนดให้มีช่องสำหรับ Browse ไฟล์รูปภาพ โดยหากเป็นการแก้ไขข้อมูลให้ดึงรูปภาพที่มีอยู่แล้วมาแสดงในความกว้าง 200 Pixel

```
<?php
echo $form->labelEx($model, 'product_photo');
echo $form->fileField($model, 'product_photo');
echo $form->error($model, 'product_photo');
?>
<?php if($model->isNewRecord!=1){?>
<div>
    <?php echo CHtml::image(Yii::app()->request->baseUrl.'/productimages/'.$model->product_photo,
        "image",array('width'=>200));?>
</div>
<?php }?>
```

ในส่วนของ Controller ใน actionCreate() บริเวณของการ \$_POST ค่ามากำหนด ให้ชื่อไฟล์เกิดจากการ Random ตัวเลข 1-9999 และตามด้วยชื่อไฟล์ที่ Upload เข้าไป โดยให้ไปเก็บไว้ใน myyiiapp/productimages/

```
if(isset($_POST['HsProducts']))
{
    $uploadedFile = CUploadedFile::getInstance($model, 'product_photo');
    $rnd = rand(1,9999);
    $fileName = $rnd."-".$uploadedFile;
    $model->attributes=$_POST['HsProducts'];
    $model->product_photo = $fileName;
    $model->created = date("Y-m-d H:i:s");
    $model->modified = date("Y-m-d H:i:s");
    if($model->save()){
        $uploadedFile->saveAs(Yii::app()->basePath.'../productimages/'.$fileName,true);
        $this->redirect(array('view','id'=>$model->id));
    }
}
```

ในส่วนของ Controller ใน actionUpdate() บริเวณของการ \$_POST ค่า ซึ่งหมายความว่าหากมีการ Upload ไฟล์ใหม่ ให้ไฟล์ใหม่ไปทับไฟล์เก่าโดยให้ใช้ชื่อเดิมซึ่งมีข้อมูลดังต่อไปนี้

```
if(isset($_POST['HsProducts']))
{
    $_POST['HsProducts']['product_photo'] = $model->product_photo; // กำหนดชื่อไฟล์ก่อน upload
    $uploadedFile = CUploadedFile::getInstance($model, 'product_photo');
    $model->attributes=$_POST['HsProducts'];
    $model->modified = date("Y-m-d H:i:s");
    if($model->save()){
        if(!empty($uploadedFile)){
            $uploadedFile->saveAs(
                Yii::app()->basePath.'../../productimages/'.$model->product_photo);
        }
        $this->redirect(array('view','id'=>$model->id));
    }
}
```

การสร้างเมนูจากตารางในฐานข้อมูล MySQL

เราสามารถสร้างเมนูโดยให้ดึงจากฐานข้อมูล MySQL ได้โดยสร้างตารางเก็บเมนูและกำหนดลักษณะต่างๆ ที่เมนูควรมี เช่น ชื่อเมนู (Label), URL (Url), สถานะ (Status) และ ตำแหน่ง (Position)

ตารางในฐานข้อมูล MySQL ตาราง mymenus

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id</u>	int(11)			No	None	AUTO_INCREMENT
2	<u>label</u>	varchar(255)	utf8_unicode_ci		No	None	
3	<u>url</u>	varchar(255)	utf8_unicode_ci		No	None	
4	<u>status</u>	int(11)			No	None	
5	<u>position</u>	int(11)			No	None	

โดยมีข้อมูลดังนี้

id	label	url	status	position
	ข้อความลิง	Link	สถานะ	ตำแหน่ง
1	หน้าหลัก	index.php?r=site/index	1	1
2	กระดานข่าว	webboard/index	1	2
3	Weblog	weblog/index	1	3
4	ร้านค้า	index.php?r=shop	0	4

สร้าง Model และ CRUD โดยใช้ Gii Generator ตามปกติ

จากนั้นสร้าง Widget สำหรับเรียกใช้งานใน protected/components/WidgetMyMenu.php โดยมีข้อมูลดังต่อไปนี้

```
<?php
Yii::import('zii.widgets.CMenu',true);

class WidgetMyMenu extends CMenu{
    public function init() {

        $criteria = new CDbCriteria();
        $criteria->condition = 'status=1';
        $criteria->order = 'position ASC';

        $items = Mymenus::model()->findAll($criteria);

        foreach ($items as $item) {
            $this->items[] = array('label'=>$item->label, 'url'=>$item->url);
        }
        parent::init();
    }
?>
```

ในส่วนของ protected/views/layouts/main.php กำหนดการแสดงผลดังนี้

```
<div id="mainmenu">
    <?php $this->widget('application.components.WidgetMyMenu');?>
</div>
```

ทดลองแสดงผลจะได้ดังนี้



จะเห็นว่าเมนูที่มีสถานะเป็น 0 จะไม่ปรากฏ

การสร้าง Webservice ด้วย SOAP

Web Service คืออะไร คำตอบง่ายๆ ครับคือระบบที่พัฒนาขึ้นมาเพื่อแลกเปลี่ยนข้อมูลข่าวสารกันในรูปของ XML ผ่านเครือข่ายอินเทอร์เน็ต ซึ่งข้อดีของ Web Service นั่นก็คือเราสามารถเขียนโปรแกรมภาษาอะไรก็ได้ให้ Request ข้อมูลมาที่ Web Service แล้วนำข้อมูลไปใช้งาน วิธีการเรียกนี้เรียกว่า RPC (Remote Procedure Call)

สำหรับส่วนนี้เป็นวิธีการใช้งานแบบตัวอย่างนะครับ โดยเราจะสร้าง Application ที่ให้บริการ Web Service ด้วย Yii Framework แล้วเขียน Application อีกตัวหนึ่งที่เรียกใช้งาน Web Service

มาเริ่มกันในส่วนของตัว Web Service ครับ โดยผมสร้าง Project Skeleton ตั้งชื่อเป็น mywebservice เมื่อเรียก URL จะเรียกดังนี้ครับ <http://localhost/mywebservice>

สร้าง Controller สำหรับ Webservice ใน protected/controllers/ServiceController.php

```
<?php
ini_set ('soap.wsdl_cache_enabled',0);
class ServiceController extends Controller{
    public function actions() {
        return array(
            'route'=>array(
                'class'=>'CWebServiceAction',
            ),
        );
    }
    /**
     * @param string the symbol of the Stat
     * @return integer the stat for browser
     * @soap
     */
    public function getStat($s) {
        $stats = array('IE'=>10, 'CHROME'=>40, 'FIREFOX'=>20, 'OTHER'=>30);
        if(isset($stats[$s])) {
            $r = $stats[$s];
        }else{
            $r = 0;
        }
        return $r;
    }
}
```

```
/**
 * @param string the $name
 * @return string the say hello
 * @soap
 */
public function getHello($name){
    return "Hello ".$name;
}

}
?>
```

ข้อสำคัญ อย่าลืมใส่ Comment ที่แสดงรายละเอียดว่ารับอะไร (param) และ Return อะไรออกมาครับ

ในที่นี้มี 2 Function นั่นก็คือ getStat() และ getHello() ไปดูกันเลยว่าเมื่อเรียกทั้ง 2 Function นี้จะมีหน้าตาอย่างไรครับ

อีก Application หนึ่งผมสร้าง Project Skeleton เป็น Client จากนั้นเขียนโปรแกรมเพิ่มในส่วน ของ SiteController เพื่อให้ทำการ Request ไปยัง Webservice ครับ

```
public function actionWebService(){
    $this->render('webservice');
}
```

ในส่วนของ View นั้นผมสร้างดังนี้ครับ protected/views/site/webservice.php

```
<?php
$client=new SoapClient('http://localhost/mywebservice/index.php?r=service/qoute');
echo $client->getStat('CHROME').'<br />';
echo $client->getHello('Manop Kongoon');
$functions = $client->__getFunctions ();
var_dump ($functions);
?>
```

ให้ไปเรียก Web Service ใน Function ต่างๆ ที่ได้สร้างไว้หลังจากทำการเรียก URL จะได้ดังนี้ครับ

40

Hello Manop Kongoon

```
array (size=2)
  0 => string 'integer getStat(string $s)' (length=26)
  1 => string 'string getHello(string $name)' (length=29)
```

เพียงเท่านี้ก็สามารถใช้งาน Web Service ได้แล้วครับ

ดรรชนี

- \$_POST, 133
- AJAX, 98, 100, 120
- Application, 1, 11, 12, 34, 35, 37, 39, 44, 74, 75, 76, 77, 80, 86, 87, 90, 98, 102, 103, 105, 129, 137
- Chart, 117, 118
- Comment, 22, 138
- Controller, 37, 57, 71, 87, 98, 100, 107, 110, 113, 116, 121, 122, 125, 131, 133, 137
- CRUD, 135
- Data Dictionary, 29, 120
- Dependence Drop Down List, 120
- Extension, 39, 41, 45, 56, 60, 80, 89, 93, 96, 113, 115, 117
- Function, 121, 125, 129, 131, 138
- Gii Generator, 22, 31, 135
- Library, 13, 107, 115
- Model, 21, 31, 33, 39, 58, 60, 62, 65, 73, 81, 87, 90, 135
- MySQL, 21, 93, 135
- Project Skeleton, 11, 137, 138
- Request, 98, 137, 138
- RPC, 137
- Search Engine Optimization, 60, 125
- SEO, 60, 98, 125
- Session, 129
- URL, 16, 31, 39, 43, 60, 84, 85, 90, 91, 95, 102, 103, 106, 107, 109, 112, 115, 116, 117, 118, 119, 123, 127, 129, 135, 137, 138
- View, 39, 74, 88, 94, 95, 98, 100, 102, 105, 107, 110, 114, 116, 117, 118, 121, 122, 126, 133, 138
- Web Service, 106, 137, 138
- Widget, 41, 80, 93, 115, 135
- XML, 98, 125, 127, 137
- Yii Framework, 1, 6, 8, 13, 14, 19, 20, 34, 35, 41, 60, 113, 125, 129, 137
- ฐานข้อมูล, 21, 24, 25, 26, 29, 73, 81, 87, 93, 94, 125, 135
- ภาษา, 34, 129, 131, 137