1:ابتدا با دستور

Yiic webapp name

پروژه را ایجاد کنیم.

2:پایگاه داده را به mysql تغییر میدهیم. در انجا نامی به پایگاه داده میدهیم.main.php

(والا من امروز در main.php چیزی مربوط به پایگاه داده نمی بینم همان ها در database.php قرار دارند گمان میکنم به دلیل اختلاف در نسخه ی من با شهرکی باشد ناگفته نماند که من تنها با تغییر همان که در database.php بود دیدم که برای شل کار کرد

اعلان پسوند جداول در database.php

'tablePrefix'=> 'tbl\_',

3: در phpmyadmin هم پایگاه داده ای با همان نام ایجاد کردیم.

4: اجرا کردن shell

Php protected/yiic.php shell

دستور فوق را در فولدری پروژه می نویسیم

5:تست وصل بودن به پایگاه داده

Echo Yii::app()->db->connectionString;

چون دستور php است نهادن سمی کالن الزامی است. در دستور فوق تنها YوS باید با حروف بزرگ نوشته شوند.

خروج از شل با exit

6:ایجاد جدول(اگر قرار باشد از migrate استفاده کنیم این مرحله را می پرانیم)

7: تغییرات مربوط به پایگاه داده که در main.php انجام شد باید در console.php هم انجام شود. گویا دیگر همش به database.php منتقل شده.

-ایجاد migration

این ها باید زمانی نوشته شوند که ما در protected باشیم(یعنی خط فرمان را به protected منتقل کرده باشیم)

Php yiic migrate create create\_project\_table

Php yiic migrate up

Php yiic migrate down

8:: اعلان پسوند جداول در main.php

'tablePrefix'=> 'tbl\_',

9: فعال کردن gii(یعنی خارج کردن ان از حالت comment در خط 23 main.php ) پسوردو false کن.

مرحله ی ساخت مدل و ... در

localhost/ts/index.php?r=gii

10: ایجاد مدل با gii

11:: ایجاد view با crud در gii(حرف اول باید بزرگ باشد)

-اعتبارسسنجی در فولدر medels متد rules

-تعیین برچسب ها اگر بخواهیم فارسی باشند(متدش پایین تر از rules اما در همان صفحه است)

-وقتی می گوییم

Project::model()->findByPk($projectId);

کلاس project ان است که در فولدر مدل قرار دارد .

Model متدی از کلاس project است. و که متدی با همین نام در کلاس والد صدا میزند

جای کلاس cactiverecord با عکسی در یادداشت ها مشخص شده است.

ما تصمیم گرفتیم که سراغ زیروبم این کار را کمتر بگیریم و فعلا استفاده از دستور فوق را که صدا زدن زنجیری چند متد است را یاد بگیریم.

دستور فوق یک شی از کلاس project برمیگرداند.کدام شی را؟ شی ای که کلید اصلی اش $projectid باشد . (ما خود برنامه هایی نوشتیم و در قسمت پارامتر به صورت دستی عدد 1 و 2 را دادیم و ان گاه دستور:

echo serialize( User::model()->findByPk(2));

را نوشتیم که دیدیم محتویات سطر دوم جدول را برایمان نوشت.

ما این دستور را در userContoroler نوشتیم در متدی به نام علی به شکل زیر:

public function filterali($filterchain)

{

echo serialize( User::model()->findByPk(2));

echo serialize( User::model()->findByPk(1));

//echo $\_GET['pid'];

echo 'i am ali';

}

کل انجه اموختم این است که با دستوری به فرم فوق در کنترولر می توان به جدول دسترسی پیدا کرد.

وقتی در view.php می گوییم $model به فیلد های کلاس مربوطه دسترسی خواهیم داشت. این نکته برای فرم هایی که قرار است مطلبی را نمایش بدهند مناسب است .مثلا view.php من خود در view.php برای user در پروژه ی learn چنین کرده ام.

و اما برعکس گاهی احتیاج است در کلاس مدل به ان چه از طریق فرم دریافت میشود دسترسی داشت که این با $this میسر می شود. اما $this چه؟

گشایش:

ان چه در سطر دوم می اید باید با نام یک فیلد از کلاس مربوط به این view یکی باشد و الا خطا می گیریم اما سطر اول تنها نوشته ایست که نمایش داده میشود.

<div class="row">

<?php echo $form->labelEx($model,'username'); ?>

<?php echo $form->textField($model,'username'); ?>

<?php echo $form->error($model,'username'); ?>

</div>

Username و password ای که از طریق صفحه log in از کاربر گرفته می شوند به کلاس loginform.php ارسال می شوند و از ان جا به کلاس UserIdentity که در فولدر components قرار دارد ارسال می شوند در ان صفحه هرگاه بگوییم $this->password یا $this->username به این مقادیر ارسال شده دسترسی پیدا می کنیم. البته می دانیم که کلاس UserIdentity از کلاس cUserIdentity ارث بری می کند . قبلا برایمان سوال شده بود که کلاس UserIdentity چرا سازند ندارد حالا جواب این سوال را می دانم. علت این است که نیازی به سازنده ندارد وقتی والدش سازنده مناسب را پیاده سازی کرده. محل قرار گرفتن این کلاس والد با عکس مشخص شده. انجا فیلدهای یوزرنیم و پسورد مقدار دهی می شوند. ما خود در کمال برنامه ای نوشتیم و دیدیم که اگر والد سازنده داشته باشد و پسر نداشته باشد هنگام ایجاد شی از کلاس پسر سازنده والد صدا زده می شود. و این هم بدیهی است زیرا هنگام ایجاد شی متد \_\_construct صدا زده می شود بدیهی است که سلسله مراتبی بالا می رود تا ان را بیابد. به گمانم همین یکی از مزایای استفاده از کلمه کلیدی \_\_construct به جای نام کلاس باید باشد.

در کد زیر lastLogin چیست؟ و از کجا می اید؟ ایا فیلدی از جدول به این نام داریم؟

echo Yii::app()->user->lastLogin;

خیر- تمام دستورفوق جز کلمات ثابتی است که در ییی وجود دارد و هیچ ربطی به نام جداول و کلاس ها و فیلدهای جدولهای ما ندارد. اما lastLogin مقدارش را از مقدار قبلی که در فیلدlast\_login\_time قرار دارد می گیرد. با دستور زیر که در userIndetity.php :

$this->setState('lastLogin',date('m/d/y g:i A',strtotime($user->last\_login\_time)));

سه کار بودند:

کپی کردن این کد

'authManager'=>array(

'class'=>'CDbAuthManager',

'connectionID'=>'db',

'itemTable' => 'tbl\_auth\_item',

'itemChildTable' => 'tbl\_auth\_item\_child',

'assignmentTable' => 'tbl\_auth\_assignment',

),

در main و console نوشتن دستورات در commands