**تقسيم مجلدات المشروع**

|  |  |
| --- | --- |
|  | 1. Db   وهو يحتوى على قاعده بيانات المشرع "يتم رفعها على السيرفر عند التشغيل" |
|  | 1. **Libs**   يعتبر هذا المجلد بالكلاسات الموجوده بداخله محرك الموقع حيث نعود اليه عندما نريد أنشاء أى عنصر من عناص الموقع مثل صفحه **معينه** أو **بلوك معين** " جزءمن الصفحة مثل قامه المستخدم" ويحتوى على   1. configClass “يحتوي على الخيارت الخاصه بالمشروع” 2. dbClass " الكلاس الخاص بالتعامل مع قواعد البيانات من أدخل وحذف وتعديل ....الخ" 3. htmlClass " الكلاس الخاص بأنشاء النصوص وأكود ال html " 4. lib" الكلاس الخاص بتجميع كل الكلاسات فى مكان واحد لسهوله الوصول اليها" 5. pagesClass " المسئول عن أنشاء الصفحه المتخلفه والتعامل مها مثل (أضافه وتعديل وحذف)" 6. view "يحتوى هذا الكلاس على مجموع من الفكشنس أو الطرق التى تتكر باستمرار وتسعدنى على أنشا الموقع "   وسوف يتم شرح كل كلاس باذن الله |
|  | 1. **Pages**   وتحتوى على الصفحات المختلفه للموقع وكل صفحة عبارة عن أوبشن أوخيارات ترسل الى  الكلاسات بداخل ال libs وذلك لعمل الصفحات أو عناصر المشروع المختلفه  وسوف يتم شرح كل صفحه بأذن الله |
|  | 1. **Panles**   وتحتوى على البلوكات الخاصه بالصفحات بلوك أى هو جزء من الصفحه يتكرر باستمرر ولا يتغير بتغير نوع الصفحه مثل قائمة الادمن "Admin menu" أو قائمة الطلابه “Student menu” |
|  | 1. **Template** "ويحتوى على شكل الموقع أى الاستايل الخاص به من css , html , javascript" |
|  | 1. upload   المجلد الذى يحتوى على الملفات المرفوعه مثل الصور الخاص الطلابه والمشرفين |
|  | 1. **webserves**   وهو يحتوى على الملفات الخاص بالتعامل مع جزئيه الموبيل أى أن جزئية التليفون ترسل وتسقبل البيانات من ملف "reqData" بداخل مجلد webserves |
|  | 1. الملفات التى يبدا منها المشروع وهى 2. Conectoer   الملف الذى يصل جميع أجزاء المشروع بعضها ببعض المكتباات "libs" و الاستايل …. الخ   1. Index   والملف الذى يبد عنده المشروع   1. Mian   هو ملف مسئول عرض محتوى الصفحات بدون أى ستايل ونحتاج ذلك عند أستخدم تقنيت “ajax “ وسوف يتم شرحه باذان الله   1. test   ملف لعمل تجارب على المشروع |

**طريقه عمل المشروع**

,

المشروع يتكون من ثلاث مراحل فى العمل

1-يبدا التشغيل للصفحة الرائيسيه "index" والتى مدروج فيها tempate file

وعن طريق المتغيير فى الربط الصفح “show”

C:\Users\aziz\AppData\Local\Microsoft\Windows\INetCache\Content.Word\img1.png

2- باستخدام الفنكش main فى صفحة Tamplet / index.php يتم أستدعاء الملف من المجلد pages

يقوم الملف بعمل مجموعه من الابوشن

وعند طريق الفنكش \_render يتم ترجمه هذة الابشن الى الصفحة المطلوبه فا فى المثال السابق يتم

1. أستخدم الامر $lib->\_pages للوصل الى class pages واستخدام الاوبشن المختلفه " سيتم شرح الابشن عند شرخ الكلاس pages”
2. استخدام الداله \_render ترجمه هذه الخيارات الى الصفحه
3. يستخد م page class الكلاسات الاخرى عند الحاجه مثل يستخدم الكلاس db class عند تخزين طلاب جديد أو التعديل على طالب موجود من قبل كما يستخدم كلاس html Class عند أنشاء الصفحه

شرح كيفيه يتم عرض صفحه supervisoer

يتم أرسال المتغيير superviseor رابط الصفحة كالتى

C:\Users\aziz\AppData\Local\Microsoft\Windows\INetCache\Content.Word\img1.png

2- فى الصفحة test نجد أمر

include './template/index.php'; "line 4"

وهو الامر المسئول عن أدراج صفى الاستايل

1. وفى صفحه template/index.php

توجد الداله

$lib->\_view->getfileTovar("pages", $\_GET['show'], "") ; "line 4"

وهى الداله المسئوله عن أحضار الملف من المجلد pages وهى أحدى دوال الكلاس view

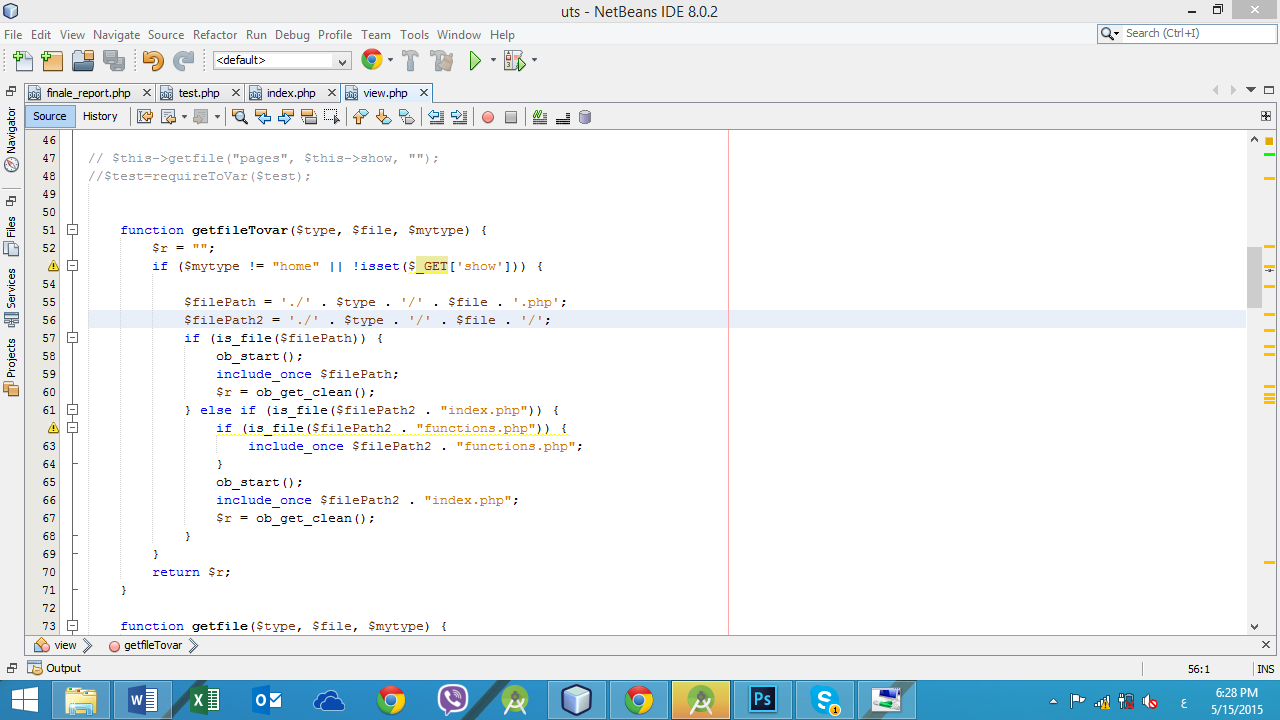
وتاخذ خياران

* الاول pages " هو المجلد المراد الاحضار منه"
* ثانى $\_GET['show'] " هو أسم الملف "

$\_GET هو أمر php المسئول عن أحضر متغيير موجود برابط الصفحة

الداله

getfileTovar " line 51 "



المسئوله عن أستدعاء الملفات

* فى السطر الاول أنشاء متتغير $rوالذى سوف يحمل محتوى الملف
* السطر الثانى عمل جمله if ذولك للتاكيد انا المتغيير$\_GET['show'], به قيمه وليس فارغ حتى لا يحدث خطا
* السطر الثالث والرابع يتم تكويين رابط الملف من المتغيرات المرسله

المره الاولى كرابط ملف والمره الثانيه كرايط لمجلد حيث من الممكن أرادج ملف أو أدراج مجلد كامل

* السطر الخامس عمل جمله if لمعرفه هل الملف موجد أو لا عن طريق داله الphp is\_file

أذا كان الملف موجود يتم أدارج الملف عن طريق السطر السلدس والسابع والثامن

* أما إذا لم يكن موجود يتم عمل جمله if جديد لمعرفه أذا كان هنا مجلد بهذا الاسم ّم لا
* اذا وجود المجلد يتم أدراج ملف index.php وملف functions من داخله فى السطور من عاشره الى خمسه عشر

وبهذا يتم الوصول الى الملف المطلوب

**التعامل مع قواعد البيانات**

للتعامل مع قواعد البيانات من أضافه وتعديل وحذف وجلب بيانات

الكلاس الخاص بذلك هو الكلاس DBClass والموجود فى فولدر libs/db

وينقسم الملف الى

|  |  |  |
| --- | --- | --- |
| عمل الداله | الداله | سطر رقم |
| خاصه بالاتصال بقاعده اليبانات | Connect() | 18 |
| لجلب صف واحد من قاعده البيانات  تاخذ   1. Table أسمة الجدول 2. sql\_felds الحقول المطلوب 3. Where أحضر البينات وفقا لشرط معين | get\_row($table, $sql\_felds = "\*", $where = "") | 105 |
| لجلب صفوف متعدده من قاعده البيانات  تاخذ   1. Table أسمة الجدول 2. sql\_felds الحقول المطلوب   Where أحضر البينات وفقا لشرط معين | get\_data($table, $data = "\*", $where = "") | 135 |
| حفظ بيانات بداخل جدول معين   1. Table أسمة الجدول 2. Data البيانات المدخله | set\_insert($table, $data) | 175 |
| تعديل بيانات بداخل جدول معين   1. Table أسمة الجدول 2. Data البيانات المدخله 3. Where التعديل وفقا لشرط معين | set\_update($table, $data, $id = 0, $where = "") | 210 |
| حذف بيانات بداخل جدول معين   1. Table أسم الجدول 2. Id أى دى الجدول المرد حذفه 3. Where حذف وفقا لشرط معين | set\_delete($table, $id = 0, $where = "", $pri\_key = 'id') | 260 |

|  |  |  |
| --- | --- | --- |
| خاصه بالاتصال بقاعده اليبانات | Connect() | 18 |

يتم عن طريقها الاتصال بقاعدة البيانات

# function connect() {

تقوم الداله mysql\_pconnect تحديد السيرفر الذى سوف يتم الاتصال وتاخذ ثلاث متغييرات

* Db host من الاكلاس config عنوان السيرفر
* db user من الاكلاس config أسم المستخدم
* Db pass من الاكلاس config كلمه السرفر

# $c = mysql\_pconnect($this->db\_host, $this->db\_user, $this->db\_pass);

تقوم mysql\_select\_db أختيار قاعده البيانات من السيفر

تأخذ متغييرن

* db\_name أسم قاعدة البيانات
* C السيرفر راجع من السطر السابق

تقوم الداله بأرجع true فى حاله الاتصال

ولهذا يتم عمل if

# if (!mysql\_select\_db($this->db\_name, $c)) {

فى حاله عدم الاتصال أطبع خطأ لا يمكن الاتصال بقاعده البيانات

# print ("ERROR: Cannot find database " . $this->db\_name);

# }

# }

|  |  |  |
| --- | --- | --- |
| لجلب صف واحد من قاعده البيانات  تاخذ   1. Table أسمة الجدول 2. sql\_felds الحقول المطلوب 3. Where أحضر البينات وفقا لشرط معين | get\_row($table, $sql\_felds = "\*", $where = "") | 105 |

# تقوم هذه الداله باحضر صف واحدمن قاعدة اليبانات وترجعه فى شكل array

function get\_row($table, $sql\_felds = "\*", $where = "") {

أذا كان المتغيير where يحمل قيمه ضع قبله كلماه WHERE حتى تتم أنشاء جمله sql

if ($where) {

$where = "WHERE " . $where;

}

تقوم الداله mysql\_query جمله sql المراد تنفيذها وفى هذه الهى جمل

SELECT $sql\_felds FROM $table $where

وهى الجمله المسئوله عن أحضار بينات من قاعده اليبانات

$query\_id = mysql\_query( "SELECT $sql\_felds FROM $table $where");

تقوم الداله mysql\_fetch\_array البيانات المطلوب الى مصفوفه

$row = mysql\_fetch\_array($query\_id, MYSQL\_ASSOC);

يتم أرجع المصوفوفه التى تحتوى على البيانات المحضرة من قاعده البينات

return $row;

}

مثلا على أستخدام الداله

]تم أحض صف من

جدول tasks

\*تسخدم لجلب كل الحقول

عندما يكون id يساوى 5

$u = get\_row("tasks", "\*", " id='5' ");

يتم طباعه القيمه تبع حقل title

Print( $u[‘title’] );

|  |  |  |
| --- | --- | --- |
| لجلب صفوف متعدده من قاعده البيانات  تاخذ   1. Table أسمة الجدول 2. sql\_felds الحقول المطلوب   Where أحضر البينات وفقا لشرط معين | get\_data($table, $data = "\*", $where = "") | 135 |

# تقوم هذه الداله باحض ر صفوف متعدهمن قاعدة اليبانات وترجعه فى شكل array

function get\_data ($table, $sql\_felds = "\*", $where = "") {

أذا كان المتغيير where يحمل قيمه ضع قبله كلماه WHERE حتى تتم أنشاء جمله sql

if ($where) {

$where = "WHERE " . $where;

}

تقوم الداله mysql\_query جمله sql المراد تنفيذها وفى هذه الهى جمل

SELECT $sql\_felds FROM $table $where

وهى الجمله المسئوله عن أحضار بينات من قاعده اليبانات

$query\_id = mysql\_query( "SELECT $sql\_felds FROM $table $where");

تقوم الداله mysql\_fetch\_array البيانات المطلوب الى مصفوفه ولان الداله تحمل أكثر من صف يتم عمل لوب أو تكرر ووضع كل صف فى المصفوف $rows

$i = 0;

while ($row = mysql\_fetch\_array($query\_id, MYSQL\_ASSOC)) {

$rows[] = $row;

$i++;

}

يتم أرجع المصوفوفه التى تحتوى على البيانات المحضرة من قاعده البينات

return $rows;

}

مثلا على أستخدام الداله

]تم أحض صف من

جدول tasks

\*تسخدم لجلب كل الحقول

عندما يكون id لا يساوى 5

$u = get\_data("tasks", "\*", " id<>'5' ");

يتم طباعه القيمه تبع حقل title فى الصف رقم 1

Print( $u[1][‘title’] );

|  |  |  |
| --- | --- | --- |
| حفظ بيانات بداخل جدول معين   1. Table أسمة الجدول 2. Data البيانات المدخله | set\_insert($table, $data) | 175 |

# تقوم هذه الداله با\خال صفوف متعدده الى قاعدة اليبانات

function set\_insert($table, $data) {

تقوم filter\_values التاكد ان البيانات الموقع المصفوفه $data تتطايق مع الحقول الموجوده بداخل الجدوال وإذاله الزيادات

$data = $this->filter\_values($table, $data);

هنا يتم تحويل المصفوفه الى جمله sql مفهومه تقوم بالادخال الى قاعدة البينات

foreach ($data as $k => $v) {

if ($k == "password") {

$v = md5($v);

}

$sql\_felds.="`" . $k . "`,";

$sql\_valus.="'" . $v . "',";

}

هنا يتم ازاله , الموجود فى أخر الجمله الsql

$sql\_felds = substr($sql\_felds, 0, (strlen($sql\_felds) - 1));

$sql\_valus = substr($sql\_valus, 0, (strlen($sql\_valus) - 1));

تقوم الداله mysql\_unbuffered\_query تننفذ جمله sql أرجع true فى حاله التنفذ بدون أخطا والعكس

$result = mysql\_unbuffered\_query("INSERT INTO $table ($sql\_felds)VALUES($sql\_valus)");

return $result;

}مثال على أستخدم الداله

يتم أدخال بيانات الى جدول

Studentes

فيتم أدخال بحقل ال title

Amrmosho

وحقلEmail

empcland@gmail.com

$data[‘title’]=”amrmosho”;

$data[‘email]=”empcland@gmail.com”;

$db->set\_insert("studentes", $data);

|  |  |  |
| --- | --- | --- |
| تعديل بيانات بداخل جدول معين   1. Table أسمة الجدول 2. Data البيانات المدخله 3. Where التعديل وفقا لشرط معين | set\_update($table, $data, $id = 0, $where = "") | 210 |

# تقوم هذه الداله بالتعديل صفوف متعدده الى قاعدة اليبانات

function set\_update($table, $data) {

تقوم filter\_values التاكد ان البيانات الموقع المصفوفه $data تتطايق مع الحقول الموجوده بداخل الجدوال وإذاله الزيادات

$data = $this->filter\_values($table, $data);

فى حاله أن متتغير where و متغيير id لا يحتوى على قيمه أخرج من الداله مع أرجع قيمه 0 او false

if (!$id && !$where) {

return 0;

}

هنا يتم تحويل المصفوفه الى جمله sql مفهومه تقوم بالادخال الى قاعدة البينات

foreach ($data as $k => $v) {

if ($k == "password") {

$v = md5($v);

}

$sql\_data.="`$k`='$v',";

}

هنا يتم ازاله , الموجود فى أخر الجمله الsql

$sql\_data = substr($sql\_data, 0, (strlen($sql\_data) - 1));

فى هذه الجمل إذا كان قيمه id لاتساوى 0 كون جمله الشرط ب id والعكس كون جمله الشرط بدون id

if ($id != 0) {

$where = " WHERE id=$id";

} else {

$where = " WHERE " . $where;

}

تقوم الداله mysql\_unbuffered\_query تننفذ جمله sql أرجع true فى حاله التنفذ بدون أخطا والعكس

$result = mysql\_unbuffered\_query("UPDATE `$table` SET $sql\_data $where");

return $result;

}

مثال على أستخدم الداله

يتم تعديل البيانات فى الجدول

Studentes

فيتم أدخال بحقل ال title

Amrmosho

وحقلEmail

[empcland@gmail.com](mailto:empcland@gmail.com)

عندما يكون id =5

$data[‘title’]=”amrmosho”;

$data[‘email]=”empcland@gmail.com”;

$db->set\_ update ("studentes", $data ,5);

|  |  |  |
| --- | --- | --- |
| حذف بيانات بداخل جدول معين   1. Table أسم الجدول 2. Id أى دى الجدول المرد حذفه 3. Where حذف وفقا لشرط معين | set\_delete($table, $id = 0, $where = "", $pri\_key = 'id') | 260 |

# تقوم هذه الداله بحذف صفوف متعدده الى قاعدة اليبانات

function set\_delete($table, $id = 0, $where = "", $pri\_key = 'id') {

فى هذه الجمل إذا كان قيمه id لاتساوى 0 كون جمله الشرط ب id والعكس كون جمله الشرط بدون id

if ($id != 0) {

$where = " WHERE id=$id";

} else {

$where = " WHERE " . $where;

}

تقوم الداله mysql\_unbuffered\_query تننفذ جمله sql أرجع true فى حاله التنفذ بدون أخطا والعكس

$result = mysql\_unbuffered\_query("DELETE FROM $table WHERE $where LIMIT 1 ");

return $result;

}

مثال على أستخدم الداله

يتم حذف البيانات فى الجدول

Studentes

عندما يكون id =5

$data[‘title’]=”amrmosho”;

$data[‘email]=”empcland@gmail.com”;

$db->set\_ delete ("studentes", 5);

**التعامل مع أكود html**

|  |  |  |
| --- | --- | --- |
| عمل الداله | الداله | سطر رقم |
| 1. المسئوله عن ترجمه خيرات عنصر html واحد | \_render() | 24 |
| 1. المسئوله عن ترجمه عناصر متعدد من html | \_renderArray | 34 |
| 1. مسئوله عن رفع الملفات | uploadFile($name) | 51 |
| 1. مسئوله عن التاك من ان القميه موجوده قبل أحضاراها | getValue($v) | 67 |
| 1. المسئوله عن تكون خصيه من خصائص html | getAttar($a, $v) | 88 |
| 1. المسئوله عن تحويل المصفوفه الى قائمة مندلسه | getSelectoptions($data) | 99 |
| 1. تحويل عنصر html واحد | getHEMLItem($data) | 139 |

المتغيير Data

هو الذى يحمل الخيرات المتخلفه التى بناء عليه يتم أنشاء كود الhtml

وياخذ هذا الشكل

$lib->\_html->data = array(

فهنا

نوع العنصر المرد عمله text

وعنوانه Title

والكلاس الخاص به col-sm-12

"title" => array("type" => "text", "title" => "Title", "pclass" => "col-sm-12"),

فهنا

نوع العنصر المرد عمله file

وعنوانه file

والكلاس الخاص به col-sm-6

"file" => array("type" => "file", "title" => "file", "pclass" => "col-sm-6"),

فهنا

نوع العنصر المرد عمله progress

وعنوانه progress

والكلاس الخاص به col-sm-6

"progress" => array("type" => "progress", "title" => "progress", "pclass" => "col-sm-6"),

فهنا

نوع العنصر المرد عمله textarea

وعنوانه dec

والكلاس الخاص به col-sm-12

"dec" => array("type" => "textarea", "title" => "dec", "pclass" => "col-sm-12")//,

);

هوه الشكل الموجود فى كل صفحات المشروع ولككننا نرسله الى كلاس pages وهو يرسله الى htmlClass وياحصل على النتيجه ويدمجها مع نتائج أخر لعمل الصفحه

|  |  |  |
| --- | --- | --- |
| 1. المسئوله عن ترجمه خيرات عنصر html واحد | \_render() | 24 |

هى فقط ترسل الخيارات والتى قديميه من المتغير dataالى الداله getHEMLItem وتستقبل العائد منها وهى فقط لتسهيل التعامل لا كثر

function \_render() {

return $this->getHEMLItem($this->data);

}

|  |  |  |
| --- | --- | --- |
| 1. المسئوله عن ترجمه عناصر متعدد من html | \_renderArray | 34 |

هى تفعل نفس الشئ والذى تقوم به render ولكن هنا تتعمل مع عناص متعدده فالخيرات الموجودة بداخل المتغيير data عباره عن مصفوفه

function \_renderArray() {

عمل المتغيير والذى يحمل المحتويات بعد ترجمتها

$r = "";

عمل تكرر لمحتويات المصفوفه data

foreach ($this->data as $k => $d) {

if (!isset($d["name"])) {

$d["name"] = $k;

}

يتم أرسال كل عنصر فى المصفوفه $data والموجد داخل $d الى الداله getHEMLItem والحصول على النتائج ووضعه فى المتغير $ي

$r .= $this->getHEMLItem($d);

}

أعاده المتغيير $r والذى يحمل على النتائج المحصله

return $r;

}

|  |  |  |
| --- | --- | --- |
| 1. مسئوله عن رفع الملفات | uploadFile($name) | 51 |

يتم من خلالها أستقبال الملفات المراد رفعهل على النت عن طريق المتغيير الخاص بالphp والمستمى $\_Files ثم عمل نسخه منه على السيرفر

$ name اسم قميه الملف

function uploadFile($name) {

عمل if للتاكد ان هناك متغيير من نوع $\_FILE وان هذا المتغير به قيمه

if (isset($\_FILES[$name]) && !$\_FILES[$name]['name'] == "") {

الحصول على كا البنانات الخاصه بالملف مثل أسمه وأمتداده وحجمه عن طرق الداله pathinfo

$path\_parts = pathinfo($\_FILES[$name]['name']);

عمل أسم جديد للملف المرافع وبنفس الامتدادوذلك لانه ممكن الممكن وجود ملف بنفس الاسم

$filename = uniqid() . "." . $path\_parts['extension'];

تحريك الملف الى المكان المرد الرفع فيه والموجود فى المتغيير upload\_to الموجود أعلى الصفحه

move\_uploaded\_file($\_FILES[$name]['tmp\_name'], $this->upload\_to . "/" . $filename);

أرجع اسم الملف الجديد الى المتغغير $\_POST وهو المتغيير الذلى استخدمه فى التعامل مع قاعده البيانات

$\_POST[$name] = $filename;

}

}

|  |  |  |
| --- | --- | --- |
| 1. مسئوله عن التاك من ان القميه موجوده قبل أحضاراها | getValue($v) | 67 |

الهدف من هذه الداله هو التكاد من القيمه الخاص بالخاص فى html قبل أضافتها ستخدم فى الدال getHEMLItem($data)

function getValue($v) {

$r = "";

التاكد من أنا القيمه مرساله

if (isset($v)) {

إذا كانت مرسله ضعها فى المتغيير R

$r = $v;

}

إذا المتغيير R يعود بقيمه

return $r;

}

|  |  |  |
| --- | --- | --- |
| 1. المسئوله عن تكون خصيه من خصائص html | getAttar($a, $v) | 88 |

تفعل ما تفعله الداله السابقه تقرييبا ولكن مع قيمه وأسم

function getAttar($a, $v) {

$r = "";

التاكد من القيمة والاسم موجديين

if (isset($a) && isset($v)) {

تكوين الساكل المناسب للخاصيه html

$r = $a . "=\"" . $v . "\" ";

}

return $r;

}

|  |  |  |
| --- | --- | --- |
| 1. المسئوله عن تحويل المصفوفه الى قائمة مندلسه | getSelectoptions($data) | 99 |

عمل هذه الداله هو أرجع قائمة مندلسه من قاعدة البينات فمثلا غندما تكون الخيارات

"group" => array("type" => "select", "title" => "group",

يتم أرجع قائمة من من الجدول supervisor\_groups وتكون النصوص من حقل title والقيم من من حقل id

"select\_type" => "db", "select\_table" => "supervisor\_groups", "select\_text" => "title", "select\_value" => "id",

"pclass" => "col-sm-6"),

شكل قائمة المندلسه فى hrml

<option value=”id”>title</option>

<option value=”id”>title</option>

<option value=”id”>title</option>

<option value=”id”>title</option>

function getSelectoptions($data) {

$r = "";

يتم معرفه النوع الملف للبينات هذا الهدف منه التوسع فى المتسقبل اذا أردنا عمل قوائم أخرى من بعيد عن قاعدة البينات مثل عمل قائمة من xml

switch ($data['select\_type']) {

case "db":

أجيد كل البينات من قاعد البينات للجدول الموجد فى المتغير'select\_table' المجود فى المصفوفه data

$s = $db->get\_data($data['select\_table']);

تعود اليبانات فى شكل متغيير مصفوفه يتم وضعه افى المتغيير $s

وعمل تكرار لكل صف فى هذه المصفوفه

foreach ($s as $a) {

$m = "";

أرسال متغيير فى حاله أذا رادنا أختيار متغيير معين

if ($data['value'] == $a[$data['select\_value']]) {

$m = "selected";

} else {

$m = "";

}

يتم تكون عنصر من عناصر القائمه كما قلنا مسبقا

$r.="<option $m value='" . $a[$data['select\_value']] . "'>" . $a[$data['select\_text']] . "</option>";

}

break;

}

return $r;

}

|  |  |  |
| --- | --- | --- |
| 1. تحويل عنصر html واحد | getHEMLItem($data) | 139 |

هى الداله الرارئيسيه والتى تقوم بتحول الخيارات الى أكود html

function getHEMLItem($data) {

$r = "";

هنا يتم وضع خيار للنوع

switch ($data['type']) {

إذان كان النوع form

الكود المطلوب <form action="demo\_form.asp" method="get">

case "form":

$h = "<form enctype=\"multipart/form-data\" "

. $this->getAttar("action", $data['action'])

. $this->getAttar("method", $data['method'])

. $this->getAttar("class", $data['class'])

. $this->getAttar("name", $data['name'])

;

$h.="/>";

break;

</form>

case "endform":

$h = "</form ";

$h.="/>";

break;

<select> الكود المطلوب  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
</select>

case "select":

$h = "<select "

. $this->getAttar("class", $data['class'] . " form-control ")

. $this->getAttar("name", $data['name'])

;

$h.=">" . $this->getSelectoptions($data, $data['value']) . "</select>";

break;

الكود المطلوب

<textarea rows="4" cols="50">

case "texta

rea":

if (isset($data['datavalue'])) {

$data['value'] = $data['datavalue'];

}

$h = "<textarea "

. $this->getAttar("class", $data['class'] . " form-control ")

. $this->getAttar("name", $data['name'])

;

$h.=">" . $this->getValue($data['value']) . "</textarea>";

break;

الكود المطلوب

<img src="smiley.gif" alt="Smiley face" height="42" width="42">

case "Image":

case "image": $h = "";

if (isset($data['value'])) {

$h = "<img src='upload/" . $data['value'] . "' />";

}

الكود المطلوب

<input type="file" name="lname">

$h .= "<input "

. $this->getAttar("type", "file")

. $this->getAttar("value", $data['value'])

. $this->getAttar("class", $data['class'] . " form-control ")

. $this->getAttar("name", $data['name'])

;

$h.="/>";

break;

الكود المطلوب

<input type="text" name="lname">

case "datetime":

$h = "<input "

. $this->getAttar("type", $data['type'])

. $this->getAttar("value", $data['value'])

. $this->getAttar("class", $data['class'] . " form-control datetimepicker")

. $this->getAttar("name", $data['name'])

;

$h.="/>";

break;

الكود المطلوب

<div class ='html-control' >This is a heading</div>

case "label":

$h = "<div class ='html-control'> "

. $data['value'];

$h.="</div>";

break;

الكود المطلوب

<a href="http://www.w3schools.com">Visit W3Schools.com!</a>

case "link":

$h = "<a "

. $this->getAttar("href", $data['href'])

. $this->getAttar("class", $data['class'] . " html-control ");

$h.="/>";

$h.= $data['value'];

$h.="</a>";

break;

الكود المطلوf

<input type="checkbox" name="lname">

case "checkbox":

$m = "";

$h = "<input $m "

. $this->getAttar("type", $data['type'])

. $this->getAttar("checked", $data['checked'])

. $this->getAttar("value", $data['value'])

. $this->getAttar("class", $data['class'] . " form-control ")

. $this->getAttar("name", $data['name'])

;

$h.="/>";

break;

الكود المطلوب

<input type="$data['type'])" name="lname">

default:

$h = "<input "

. $this->getAttar("type", $data['type'])

. $this->getAttar("value", $data['value'])

. $this->getAttar("class", $data['class'] . " form-control ")

. $this->getAttar("name", $data['name'])

. $this->getAttar("maxlength", $data['maxlength'])

;

$h.="/>";

break;

}

بعد ترجمه الكود يتم دمجه من html لوضع حدود له ووضع العنوان

if ($data['type'] == "form" || $data['type'] == "endform") {

$r.=$h;

} else {

$r.="<div class='formrow " . $data['type'] . "\_p " . $data['pclass'] . "'><label>" . $this->getValue($data['title']) . "</label><div>" . $h . "</div></div>";

}

مرجع للتاجز الخاصه

http://www.w3schools.com/tags/tag\_div.asp

return $r;

}