**الفئة database:**

فى الجزء التالى سوف نقوم بإنشاء فئة خاصة بنا لإجراء الإستعلامات بسهولة ودقة أكثر وتجنباً للأخطاء كإدخال قيم فارغة وخلافه لقاعدة البيانات. سنقوم بإنشاء فئة جديدة باسم database. وفى داخل الفئة سنقوم بتعريف مجموعة من المتغيرات الخاصة (الخصائص): المتغير $RDBMType لحفظ نوع قاعدة البيانات والمتغير $username لحفظ اسم المستخدم و المتغير$password لحفظ كلمة السر والمتغير $password لحفظ كلمة السر و المتغير $dbhost لحفظ اسم الخادم و المتغير $database لحفظ اسم قاعدة البيانات.

سوف نستخدم دالة البناء $construct() و نضع فيها الأكواد التى تستخدم فى عملية الأتصال بقاعدة البيانات باستخدام الفئة PDO .و اليك الفئة كاملة وسوف نشرح دوال الفئة فى الأجزاء التالية:

<?php

class database {

مجموعة المتغيرات الخاصة

private $RDBMType = "mysql";

private $username = "root";

private $password = "";

private $dbhost = "localhost";

private $database = "forum";

دالة البناء و التى يتم تنفيذها بطريقة آلية عند انشاء كائنات من هذه الفئة.

function \_\_construct() {

//Connection string

$this->pdo = new PDO($this->RDBMType . ":host=" . $this->dbhost . ";dbname=" . $this->database, $this->username, $this->password, array(PDO::MYSQL\_ATTR\_INIT\_COMMAND => "SET NAMES UTF8"));

}

الدالة التى تستخدم فى قطع الأتصال.

public function Disconnect(){

$this->pdo = null;

$this->isConnected = false;

}

public function getRow($query, $params=array()) {

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->fetch();

}

public function getCount($query, $params=array())

{

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->rowCount();

}

public function getRows($query, $params=array()){

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->fetchAll();

}

public function insertRow($query, $params){

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->rowCount();

}

public function updateRow($query, $params){

return $this->insertRow($query, $params);

}

public function deleteRow($query, $params){

return $this->insertRow($query, $params);

}

public function phpAlert($msg) {

echo '<script type="text/javascript">alert("' . $msg . '")</script>';

}

}

انشاء كائن اسمه $db من الفئة database.

$db = new database;

**الشرح:**

**الدالة الأولى getRow()** سوف تستخدم فى ارجاع سجل واحد فقط.

public function getRow($query, $params=array()){

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->fetch();

}

تاخذ هذه الدالة الأستلام $query المطلوب تنفيذه و مصفوفة من القيم $params و التى لها قيمة افتراضية مصفوفة فارغة. فى السطر الأول نقوم باستخدام الطريقة prepare() لتهيئة الأستعلام .وفى السطر الثانى نقوم بتنفيذ الأستعلام عن طريق الدالة execute() و فى السطر الأخير تعيد الدالة سجلا واحدا من ناتج الأستعلام عن طريق الدالة fetch().

**مثال:**

$query = "SELECT NameA,level,GPA FROM students WHERE StuID =?";

$row = $db->getRow($query,array($id));

$name = $row[0];

$level=$row[1];

$Gpa=$row[2];

**الدالة الثانية getRows()** تستخدم فى اعادة كل السجلات الناتجه عن تنفيذ جملة الاستعلام:

public function getRows($query, $params=array()){

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->fetchAll();

}

تأخذ هذه الدالة كمدخلات الأستعلام $query و مصفوفة االمعاملات $params . فى السطر الأول نقوم باستخدام الطريقة prepare() لتهيئة الأستعلام .وفى السطر الثانى نقوم بتنفيذ الأستعلام عن طريق الدالة execute() و فى السطر الأخير تعيد الدالة كل السجلات الناتجة من تنفيذ الأستعلام عن طريق الدالة fetchAll().

مثال على كيفية استخدام الدالة:

$query = "SELECT CourseName, CourseCode, CrhTotal, prerequisite

from courses where Semester=? and level <=?";

$getrows = $db->getRows($query,array($Semester,$level));

**الدالة الثالثة getCount()** تقوم بإعادة عدد السجلات الناتجة من تنفيذ الأستعلام.

public function getCount($query, $params=array()) {

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->rowCount();

}

تأخذ هذه الدالة كمدخلات الأستعلام $query و مصفوفة االمعاملات $params . فى السطر الأول نقوم باستخدام الطريقة prepare() لتهيئة الأستعلام .وفى السطر الثانى نقوم بتنفيذ الأستعلام عن طريق الدالة execute() و فى السطر الأخير تعيد الدالة عدد السجلات الناتجة من تنفيذ الأستعلام عن طريق الدالة rowCount().

مثال على كيفية استخدام الدالة:

$query = "SELECT id FROM members WHERE stuId = ?";

$row = $db->getCount($query,array($stuid));

**الدالة الرابعة insertRow()** لإضافة سجل الى الجدول و تعيد عدد السجلات التى تأثرت :

public function insertRow($query, $params){

$stmt = $this->pdo->prepare($query);

$stmt->execute($params);

return $stmt->rowCount();

}

مثال على الأستخدام:

// Insert the new user into the database

$insertrow = $db ->insertRow("INSERT INTO members

(stuid,password, salt) VALUES (?, ?,?)",

array($stuid, $password,$random\_salt));

if ($insertrow==0) {

header('Location: error.php?err=Registration failure: INSERT');

exit();

}

**فى الفئة المقترحة يتم اجراء عمليىة التحديث عن طريق الدالة الخامسة updateRow():**

public function updateRow($query, $params){

return $this->insertRow($query, $params);

}

تأخذ هذه الدالة كمدخلات الأستعلام $query و مصفوفة االمعاملات $params . فى السطر الوحيد نقوم بمناداة الدالة insertRow() و التى تعيد عدد السجلات التى تأثرت.

**فى الفئة المقترحة يتم اجراء عمليىة التحديث عن طريق الدالة deleteRow ():**

public function deleteRow($query, $params){

return $this->insertRow($query, $params);

}

تأخذ هذه الدالة كمدخلات الأستعلام $query و مصفوفة االمعاملات $params . فى السطر الوحيد نقوم بمناداة الدالة insertRow() و التى تعيد عدد السجلات التى تأثرت.