**oop Bài thực hành 09**

**PHP– Hướng đối tượng**

***Phần 1. Mục tiêu***

* Nắm được cách tạo và cấu trúc của một class
* Hiểu về các bổ từ truy cập public, protected, private
* Hiểu về phương thức và thuộc tính static
* Hiểu và sử dụng phương hức khởi tạo \_\_construct()

***Phần 2. Kiến thức cần nắm vững***

* Sử dụng từ khóa class để tại class mới, theo sao class là tên class
* class TenCLass { nội dung class khai báo trong đây... }
* Mỗi class sẽ có các thuộc tính và phương thức
* ĐI kèm với các thuộc tính sẽ có các bổ từ truy cập: public, protected, private
* Ngoài ra còn có các thuộc tính và phương thức kiểu static
* Các thuộc tính, phương thức **pubic** có thể truy cập bất kỳ đâu, chỉ cần khởi tạo class
* Các thuộc tính, phương thức **protected** có thể truy cập tại chính nó và cho phép thay đổi tại class con kế thừa
* Các thuộc tính, phương thức **private** chỉ cho phép truy cập tại chính nó để tránh việc truy cập không cần thiết bên ngoài class
* các thuộc tính, phương thức **static** có thể truy cập bất kỳ đâu mà không cần khởi tạo class
* Một class có thể thừa kế class khác với extends tên class
* class TenCLass extends TenClassCha{ nội dung ... }
* **abstract**: Khi một lớp, một phương thức có từ khóa này thì nó chỉ được sử dụng cho các lớp thành viên
* **final**: Lớp hay phương thức có từ khóa này thì không thể chính sửa.

***3. Bài thực hành Step by Step***

**BÀI THỰC HÀNH 1**: Tạo class php tên là Database, class này hỗ trợ việc thực hiện một số thao tác với CSDL như: kết nối, select, delete, getById…

**CHÚ Ý:** hãy tạo săn 1 CSDL và có các bảng như category, product như các bài thực hành trước

**Bước 1**: Kết nối CSDL - Tạo mới thư mục project và tạo file Database.php, sau đó khai báo code sau

<?php

class Database

{

    /\*\*

    \* Khai báo các thuộc tính kết nối CSDL

    \*/

    private $host = 'localhost'; // địa chỉ máy chủ domain hoặc IP

    private $user = 'root'; // tài khoản quản trị CSDL

    private $password = ''; // Mật khẩu tài khoản quản trị CSDL

    private $db\_name = 'demo\_db1'; // tên CSDL cần kết nối

    protected $conn; // biến lưu đối tượng kết nối

  protected $table; // biến lưu tên bảng cần truy vấn

    // sử dụng phương thức khởi tạo để kết nối CSDL va gán đôi tượng cho thuộc tính $conn

    public function \_\_construct(){

        $this->conn = new Mysqli($this->host, $this->user, $this->password, $this->db\_name);

    }

}

?>

**Bước 2:** Tạo các phương thức thao tác dữ liệu

**1. Phương thức table($table)**, phương thức này luôn được gọi trước, có mục đích để gán giá trị tên bảng đang muốn thao tác

// phương thức table để thực hiện gán giá trị cho thuộc tính $table

public static function table($table){

    $instance = new static;

    $instance->table = $table;

    return $instance;

}

Ví dụ về cách dùng

Tạo thêm file demo.php sau đó test code sau

<?php

include 'Database.php'; // gọi thư viện vào

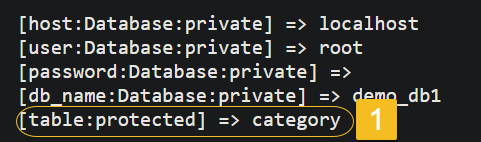
$cats = Database::table('category');

echo '<pre>';

print\_r($cats);

?>

Kết quar in ra thông tin có dạng, trong đó chú ý thuộc tính table được gán



**2. Phương thức get,** phương thức này có nhiệm vụ thao tác lệnh SELECT và trả về mảng dữ liệu là các dòng trong bảng

Khai báo tiếp phương thức get có code sau

public function get(){

    $results = [];

    $query = $this->conn->query("SELECT \* FROM $this ->table");

    while($row = $query->fetch\_object()){

        $results[] = $row;

    }

    return $results;

}

VÍ dụ cách sử dụng, mở lại trang demo.php đang test ở bước trên, truy cập đếp phương thức get

<?php

include 'Database.php'; // gọi thư viện vào

$cats = Database::table('category')->get();

echo '<pre>';

print\_r($cats);

?>

Kết quả sẽ in ra mảng dữ liệu của bảng nếu có dữ liệu trong bảng

Array

(

    [0] => stdClass Object

        (

            [id] => 1

            [name] => Áo quần

            [status] => 0

        )

    [1] => stdClass Object

        (

            [id] => 2

            [name] => Giày dép

            [status] => 0

        )

)

**3. Phương thức getById($id)**, phương thức này sẽ trả về 1 dòng dữ liệu theo $id truyền vào chính là giá trị id của dòng đó

TIếp tục tải file Database.php soạn tiếp phương thức này

public function getById($id){

    $query = $this->conn->query("SELECT \* FROM $this->table WHERE id = $id");

    return $query->fetch\_object();

}

Demo cách sử dụng, mở lại file **demo.php** đang test và sử dụng như sau

<?php

**4.** **Phương thức** **delete($id),** phương thức này thực hiện xóa 1 dòng dữ liệu theo id của dòng muốn xóa

Mở lại file **Database.php** soạn code tiếp như sau

public function delete($id){

    return $this->conn->query("DELETE FROM $this->table WHERE id = $id");

}

Cách sử dụng tương tự như trên

<?php

include 'Database.php'; // gọi thư viện vào

$deleted = Database::table('category')->delete(1);

echo '<pre>';

var\_dump($cat); // true hoặc false

?>

***Phần 3. Bài tập tự làm***

**Bài 1.** Tương tự như các bài trên, sinh viên hãy xây dựng các phương thức như sau đây:

**1. create($ata)**: phương thức này thực hiện thêm mới vào bảng, với data là mảng gồm key là tên cột và value là giá trị của cột VD:

<?php

include 'Database.php'; // gọi thư viện vào

$data = [

    'name' => 'Demo categỏy',

    'status' => 1

];

$created = Database::table('category')->create($data);

?>

**2.** **update($data):** phương thức này thực hiện cập nhật dữ liệu trong bảng theo id, data là mảng dữ liệu cần update có dạng sau

<?php

include 'Database.php'; // gọi thư viện vào

$data = [

    'id' => 2,

    'name' => 'Demo categỏy',

    'status' => 1

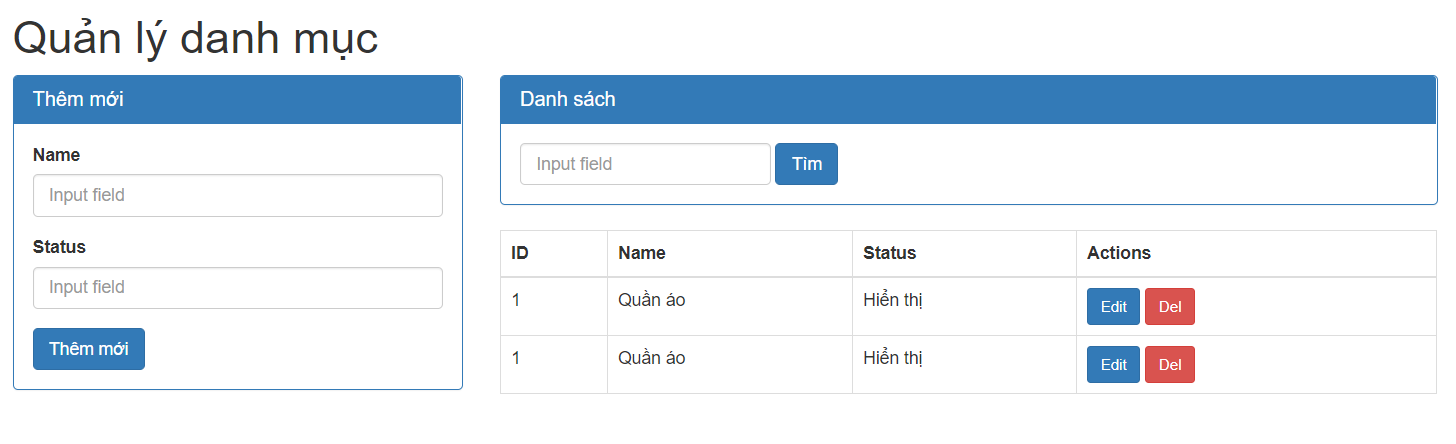
];

$updateted = Database::table('category')->update($data);

?>

**Bài 2:** Áp dụng class Database để thực hiện các chức năng sau

Hiển thị danh sách danh mục, xóa, sửa và thêm mới danh mục



Tương tự với Quản lý sản phẩm

Bài 2: Xây dựng class Cart xử lý giỏ hàng gồm có các thuộc tính và phương thức như:

<?php

class Cart

{

    public $items=[]; // lưu các sản phẩm trong giỏ hàng

    public $totalPrice; // lưu tống tiền

    public $totalQuantity; // lưu tổng số lượng

    public function \_\_construct(){

        // các giá trị mặc định khi khởi tạo lớp

    }

    public function add($product, $quantity = 1){

        // lưu sản phẩm vào giỏ hàng

    }

    public function update($id, $quantity = 1){

        // cập nhật số lượng cho sản phẩm theo id

    }

    public function remove($id){

        // Xóa sản phẩm khỏi giỏ hàng theo id

    }

    public function clear(){

        // Xóa sạch sản phẩm khỏi giỏ hàng

    }

    //..... các phương thức bổ trợ khác nếu cần

}

?>