

# Lập trình HĐT với PHP và PDO

## Nội dung thực hành:

- Sử dụng phong cách lập trình HĐT để xây dựng script PHP, bao gồm cả sử dụng chuẩn tự động nạp lớp PSR-4
- Sử dụng giao diện PDO để truy xuất (đọc, thêm, sửa, xóa) dữ liệu trong CSDL MySQL

## Cài đặt, cấu hình:

- Tạo CSDL *ct275\_lab3db* và import script *contacts.sql* đã cho CSDL này. Kiểm tra rằng sau khi import thành công, một bảng dữ liệu tên *contacts* được tạo ra trong CSDL *ct275\_lab3db* với các trường thông tin sau: id, name, phone, notes, created\_at, updated\_at
- Trong *C:/xampp* tạo thư mục *apps* và sao chép thư mục *lab3* vào thư mục *apps* này
- Cấu hình host ảo cho thư mục *C:/xampp/apps/lab3/www*:
  - Hiệu chỉnh tập tin *C:/xampp/apache/conf/httpd.conf*.

```
...
Listen 80
Listen 8080
...
```

- Hiệu chỉnh tập tin *C:/xampp/apache/conf/extra/httpd-vhosts.conf*.

```
<VirtualHost *:8080>
    DocumentRoot "C:/xampp/apps/lab3/www"
    ServerName localhost
    # Set access permission
    <Directory "C:/xampp/apps/lab3/www">
        AllowOverride None
        Require all granted
    </Directory>
</VirtualHost>
```

- Restart lại server Apache
- Mở trình duyệt bất kì, kiểm tra đường dẫn <http://localhost:8080/list-contacts.php> hoạt động

## Sử dụng chuẩn autoload PSR-4 và tạo lớp sinh kết nối đến CSDL

- Trong thư mục *lab3*, tạo tập tin *autoload.php* với nội dung như sau:

```
<?php

spl_autoload_register(function ($class) {
    // Tiếp đầu ngữ không gian tên.
    // Các lớp trong dự án sẽ sử dụng tiếp đầu ngữ này cho không gian tên
    $prefix = 'CT275\\Lab3\\';
    // Thư mục cơ sở ứng với tiếp đầu ngữ không gian tên
    $base_dir = __DIR__ . '/src/';

    $len = strlen($prefix);
    if (strncmp($prefix, $class, $len) !== 0) {
        return;
    }

    $relative_class = substr($class, $len);

    $file = $base_dir . str_replace('\\', '/', $relative_class) . '.php';

    if (file_exists($file)) {
        require $file;
    }
});
```

- Trong thư mục *lab3*, tạo tập tin *bootstrap.php* có nội dung sau:

```
<?php

define('DEBUG', true);
if (DEBUG) {
    ini_set('display_errors', 1);
    ini_set('display_startup_errors', 1);
    error_reporting(E_ALL);
}

session_start();

require __DIR__ . "/autoload.php";
require __DIR__ . "/src/helpers.php";
```

3. Trong thư mục *lab3/src*, tạo tập tin *Db.php* có nội dung như sau (chú ý thay đổi các tham số kết nối CSDL):

```
<?php
namespace CT275\Lab3;

class Db {
    private static $instance = NULL;

    private function __construct() {}
    private function __clone() {}

    public static function getInstance() {
        if (!isset(self::$instance)) {
            $driver = 'mysql';
            $host = 'localhost';
            $name = 'ct275_lab3db';
            $options[\PDO::ATTR_ERRMODE] = \PDO::ERRMODE_EXCEPTION;
            $dsn = "$driver:host=$host;dbname=$name;charset=utf8";
            self::$instance = new \PDO($dsn, 'root', '', $options);
        }

        return self::$instance;
    }
}
```

#### Liệt kê các contact####

1. Trong lớp *Contact* (*lab3/src/Contact.php*), thêm vào các hàm sau:

```

public static function all()
{
    $contacts = [];

    $db = Db::getInstance();
    $stmt = $db->prepare("select * from contacts");
    $stmt->execute();
    while ($row = $stmt->fetch()) {
        $contact = static::createFromDb($row);
        $contacts[] = $contact;
    }

    return $contacts;
}

protected static function createFromDb(array $data)
{
    $contact = new Contact();
    $contact->id = $data['id'];
    $contact->name = $data['name'];
    $contact->phone = $data['phone'];
    $contact->notes = $data['notes'];
    $contact->created_at = $data['created_at'];
    $contact->updated_at = $data['updated_at'];
    return $contact;
}

```

2. Hiệu chỉnh script *list-contacts.php* (*lab3/www/list-contacts.php*):

```

<?php
    require "../bootstrap.php";
    $contacts = CT275\Lab3\Contact::all();
?>
<!DOCTYPE html>
<html lang="en">
<head>
...
<tbody>
    <?php foreach($contacts as $contact): ?>
        <tr>
            <td><?=htmlspecialchars($contact->name)?></td>
            <td><?=htmlspecialchars($contact->phone)?></td>
            <td><?=htmlspecialchars(date("d-m-Y",
                strtotime($contact->created_at)))?></td>
            <td><?=htmlspecialchars($contact->notes)?></td>
            <td><a href="/edit-contact.php" class="btn btn-xs btn-warning">
                <i alt="Edit" class="fa fa-pencil"> Edit</i></a>
            <a href="#" class="btn btn-xs btn-danger">
                <i alt="Delete" class="fa fa-trash"> Delete</i></a></td>
        </tr>
    <?php endforeach ?>
</tbody>
...

```

3. Mở trình duyệt bất kì, truy cập đến <http://localhost:8080/list-contacts.php> để kiểm tra kết quả

### Thêm mới contact:

1. Thêm hàm save vào lớp Contact như sau:

```

public function save()
{
    $result = false;

    $db = Db::getInstance();

    if ($this->id) {
        $stmt = $db->prepare("update contacts set name = :name, phone = :phone,
                                notes = :notes, updated_at = now() where id = :id");
        $result = $stmt->execute([
            'name' => $this->name,
            'phone' => $this->phone,
            'notes' => $this->notes,
            'id' => $this->id]);
    } else {
        $stmt = $db->prepare(
            "insert into contacts (name, phone, notes, created_at, updated_at)
            values (:name, :phone, :notes, now(), now())");
        $result = $stmt->execute([
            'name' => $this->name,
            'phone' => $this->phone,
            'notes' => $this->notes]);
        if ($result) {
            $this->id = $db->lastInsertId();
        }
    }

    return $result;
}

```

2. Thêm đoạn mã xử lý thêm contact vào script add-contact.php (lab3/www/add-contact.php):

```

<?php
    require "../bootstrap.php";

    use CT275\Lab3\Contact;

    $errors = [];

    if ($_SERVER['REQUEST_METHOD'] === 'POST') {
        $contact = new Contact($_POST);
        $errors = $contact->validate() ? [] : $contact->getValidationErrors();

        // Dữ liệu hợp lệ...
        if (count($errors) === 0) {
            $contact->save();
            redirect('/list-contacts.php');
        }
    }
?>
<!DOCTYPE html>
<html lang="en">
<head>
...

```

3. Mở trình duyệt bất kì, truy cập đến <http://localhost:8080/add-contact.php> và thử thêm một contact để kiểm tra

### Hiệu chỉnh contact

1. Thêm hàm find và update vào lớp Contact như sau:

```

public static function find($id)
{
    $contact = null;

    $db = Db::getInstance();
    $stmt = $db->prepare("select * from contacts where id = :id");
    $stmt->execute(['id' => $id]);

    if ($row = $stmt->fetch()) {
        $contact = static::createFromDb($row);
    }

    return $contact;
}

public function update(array $data)
{
    $this->fill($data);
    if ($this->validate()) {
        return $this->save();
    }
    return false;
}

```

2. Hiệu chỉnh script *edit-contact.php* (*lab3/www/edit-contact.php*):

```

<?php
    require "../bootstrap.php";

    use CT275\Lab3\Contact;

    $id = isset($_REQUEST['id']) ?
        filter_var($_REQUEST['id'], FILTER_VALIDATE_INT) : false;
    if (!$id || ($editedContact = Contact::find($id)) === null) {
        redirect('/list-contacts.php');
    }

    $errors = [];
    if ($_SERVER['REQUEST_METHOD'] === 'POST') {
        if ($editedContact->update($_POST)) {
            // Cập nhật dữ liệu thành công
            redirect('/list-contacts.php');
        }
        // Cập nhật dữ liệu không thành công
        $errors = $editedContact->getValidationErrors();
    }

    // Contact được hiển thị trong form
    $contact = $editedContact->toArray();
?>
<!DOCTYPE html>
<html lang="en">
<head>
...

```

3. Trong script *list-contacts.php*, thay đổi đường dẫn đến script *edit-contact.php*:

```

...
<tbody>
    <?php foreach($contacts as $contact): ?>
        <tr>
            ...
            <td><a href="/edit-contact.php?id=<?=htmlspecialchars($contact->id)?>"
                class="btn btn-xs btn-warning">
                <i alt="Edit" class="fa fa-pencil"> Edit</i></a>
                <a href="#" class="btn btn-xs btn-danger">
                <i alt="Delete" class="fa fa-trash"> Delete</i></a></td>
            </tr>
        <?php endforeach ?>
    </tbody>
...

```

#### 4. Mở trình duyệt web bất kì và kiểm tra cập nhật contact

### Xóa contact

#### 1. Thêm hàm delete vào lớp Contact:

```

public function delete()
{
    $db = Db::getInstance();
    $stmt = $db->prepare("delete from contacts where id = :id");
    return $stmt->execute(['id' => $this->id]);
}

```

#### 2. Trong thư mục *lab3/www*, tạo script *del-contact.php*:

```

<?php
    require "../bootstrap.php";

    use CT275\Lab3\Contact;

    if ($_SERVER['REQUEST_METHOD'] == 'POST'
        && isset($_POST['id'])
        && ($contact = Contact::find($_POST['id'])) != null) {
        $contact->delete();
    }

    redirect('/list-contacts.php');

```

#### 3. Hiệu chỉnh script list-contacts.php:

```

...
<tbody>
  <?php foreach($contacts as $contact): ?>
    <tr>
      ...
      <td><a href="/edit-contact.php?id=<?=htmlspecialchars($contact->id)?>"
        class="btn btn-xs btn-warning">
        <i alt="Edit" class="fa fa-pencil"> Edit</i></a>
        <form class="delete" action="/del-contact.php"
          method="POST" style="display: inline;">
          <input type="hidden" name="id"
            value="<?=htmlspecialchars($contact->id)?>">
          <button type="submit" class="btn btn-xs btn-danger"
            name="delete-contact">
            <i alt="Delete" class="fa fa-trash"> Delete</i></button>
          </form></td>
      </tr>
    <?php endforeach ?>
  </tbody>
...
<div id="delete-confirm" class="modal fade" role="dialog">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close"
          data-dismiss="modal">&times;</button>
        <h4 class="modal-title">Confirmation</h4>
      </div>
      <div class="modal-body">Do you want to delete this contact?</div>
      <div class="modal-footer">
        <button type="button" data-dismiss="modal"
          class="btn btn-danger" id="delete">Delete</button>
        <button type="button" data-dismiss="modal"
          class="btn btn-default">Cancel</button>
      </div>
    </div>
  </div>
</div>

<footer class="footer">
  <div class="container">
    <p class="text-muted">Copyright &copy; 2016 Web Development Course</p>
  </div>
</footer>
...
<script>
  $(document).ready(function(){
    new WOW().init();
    $('#contacts').DataTable();

    $('button[name="delete-contact"]').on('click', function(e){
      var $form=$(this).closest('form');
      e.preventDefault();
      $('#delete-confirm').modal({ backdrop: 'static', keyboard: false })
      .one('click', '#delete', function() {
        $form.trigger('submit');
      });
    });
  });
</script>

```

### 3. Mở trình duyệt web bất kì và kiểm tra xóa contact

Thêm các thông báo tác vụ thành công/thất bại

