

Lập trình HĐT với PHP: Lớp, PSR-4 và PDO

Nội dung thực hành:

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

Yêu cầu: File báo cáo cần nộp là file PDF chứa hình ảnh minh họa kết quả thực hiện (tại các bước yêu cầu kiểm tra kết quả). Trong file báo cáo ghi rõ thông mã sinh viên và họ tên sinh viên

Cấu hình MySQL và Apache:

- Tạo CSDL *ct275_lab3* và import *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_lab3* với các trường thông tin sau: id, name, phone, notes, created_at, updated_at (nếu import bị lỗi sinh viên có thể tự tạo lại bảng dữ liệu dựa vào thông tin trong file *contacts.sql*)
- Tạo thư mục *C:/xampp/apps/lab3* và sao chép nội dung thư mục *lab3/start* đã cho vào thư mục này
- Cấu hình host ảo cho thư mục *C:/xampp/apps/lab3/www*:
 - Hiệu chỉnh tập tin *C:/xampp/apache/conf/extra/httpd-vhosts.conf*:

```
<VirtualHost *:8080>
    DocumentRoot "C:/xampp/apps/lab3/www"
    ServerName ct275-lab3.localhost
    # Set access permission
    <Directory "C:/xampp/apps/lab3/www">
        AllowOverride None
        Require all granted
    </Directory>
</VirtualHost>
```

3.2 Restart lại server Apache

- Mở trình duyệt bất kì, kiểm tra đường dẫn <http://ct275-lab3.localhost/> 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;
    }
});

```

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

```

<?php

ini_set('display_errors', '1');
ini_set('display_startup_errors', '1');
error_reporting(E_ALL);

session_start();

require_once __DIR__ . '/autoload.php';
require_once __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;

use PDO;

class Db {
    private static $instance = NULL;

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

    public static function getInstance() {
        if (!isset(static::$instance)) {
            $dsn = 'mysql:host=localhost;dbname=ct275_lab3;charset=utf8';
            $options[PDO::ATTR_ERRMODE] = PDO::ERRMODE_EXCEPTION;
            static::$instance = new PDO($dsn, 'root', '', $options);
        }
        return static::$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 *index.php* (*lab3/www/index.php*):

```

<?php
    require_once '../bootstrap.php';

    use CT275\Lab3\Contact;

    $contacts = Contact::all();
?>
<!DOCTYPE html>
<html lang="en">
<head>
...
<tbody>
    <?php foreach($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.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://ct275-lab3.localhost/> 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 >= 0) {
        $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.php (lab3/www/add.php):

```
<?php
    require_once '../bootstrap.php';

    use CT275\Lab3\Contact;

    $errors = [];

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

        if (empty($errors)) {
            $contact->save();
            redirect('/');
        }
    }
?>
<!DOCTYPE html>
<html lang="en">
<head>
...

```

3. Mở trình duyệt bất kì, truy cập đến <http://ct275-lab3.localhost/add.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.php* (*lab3/www/edit.php*):

```
<?php
    require_once '../bootstrap.php';

    use CT275\Lab3\Contact;

    $id = isset($_REQUEST['id']) ?
        filter_var($_REQUEST['id'], FILTER_VALIDATE_INT) : -1;
    if ( $id < 0 || ! ($edited_contact = Contact::find($id)) ) {
        redirect('/');
    }

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

    // Contact được hiển thị trong form
```

```

        $contact = $edited_contact->toArray();
    ?>
<!DOCTYPE html>
<html lang="en">
<head>
...

```

3. Trong script index.php, thay đổi đường dẫn đến script edit.php:

```

...
<tbody>
    <?php foreach($contacts as $contact): ?>
        <tr>
            ...
            <td><a href="/edit.php?id=<?=htmlspecialchars($contact->getId())?>"
                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 delete.php:

```

<?php
    require_once '../bootstrap.php';

    use CT275\Lab3\Contact;

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

    redirect('/');

```

3. Hiệu chỉnh script index.php:

```

...

```

```

<tbody>
  <?php foreach($contacts as $contact): ?>
    <tr>
      ...
      <td><a href="/edit.php?id=<?=htmlspecialchars($contact->getId())?>"
        class="btn btn-xs btn-warning">
        <i alt="Edit" class="fa fa-pencil"> Edit</i></a>
        <form class="delete" action="/delete.php"
          method="POST" style="display: inline;">
          <input type="hidden" name="id"
            value="<?=htmlspecialchars($contact->getId())?>">
          <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>
</div>

<?php include('../parts/footer.php') ?>
...
<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>

```

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