

THỰC HÀNH CÔNG NGHỆ WEB

YÊU CẦU CHUNG:

- **Sinh viên tạo một báo cáo duy nhất cho TẤT CẢ các buổi thực hành (4 buổi).** Sau mỗi buổi thực hành sinh viên nộp báo cáo (chứa bài làm của buổi hiện thời và các buổi trước đó) dưới dạng tập tin PDF xem như báo cáo tiến độ, không tính điểm. **Điểm thực hành sẽ là điểm của bài báo cáo PDF sinh viên nộp trong buổi thực hành cuối cùng (chứa tất cả bài làm của tất cả các buổi).** Không nộp bài báo cáo tiến độ đúng hạn buổi nào sẽ không tính điểm buổi đó.
- Trong báo cáo ghi thông tin MSSV, Họ tên, Nhóm học phần.
- Các sinh viên bị phát hiện sao chép bài làm của nhau sẽ bị điểm 0 toàn bộ phần thực hành.
- Điểm tối đa cho mỗi buổi thực hành là 10 điểm (tương đương 1 điểm của học phần).

BUỔI 2: JQUERY

Đối với mỗi bài làm trong buổi 2 sinh viên chụp toàn màn hình desktop các ảnh kết quả và phần mã nguồn chính.

Bài 1. Làm việc với thư viện jQuery (1 điểm).

- **Mục tiêu:** Biết cách làm việc với thư viện jQuery.
- **Yêu cầu:** Tập tin index.html đã cho (trong thư mục ex1) sử dụng JavaScript DOM API và các hàm từ thư viện youtube.js để chuyển đổi một danh sách các đường dẫn video dạng văn bản sang một danh sách các đường dẫn video dạng ảnh thumbnail. Thay vì dùng JavaScript DOM API, hãy dùng jQuery để thao tác với DOM.
- **Hướng dẫn:**

Thêm thư viện jQuery vào trang tài liệu HTML và hiệu chỉnh đoạn mã đã cho như sau:

```
<script>
    $("a").each(function () {
        const videoLink = $(this); // this → an <a> DOM element
        const linkUrl = videoLink.prop("href");
        const thumbnailUrl = youtube.generateThumbnailUrl(linkUrl);

        const thumbnailImg = $("<img>");
        thumbnailImg.prop("src", thumbnailUrl);
        videoLink.append(thumbnailImg);
    });
</script>
```

Bài 2. Xử lý sự kiện và tạo hiệu ứng với jQuery (2 điểm).

- **Mục tiêu:** Biết cách xử lý sự kiện và tạo hiệu ứng đơn giản với jQuery.
- **Yêu cầu:** Từ kết quả bài tập 1, thêm mã lệnh cần thiết để có thể xem các đoạn phim ngay bên trong trang web như hình minh họa bên dưới.

Các đoạn phim hay nhất

- Trip through the 80s



- Otters Holding Hands



- The Ooooh Catttttt



- **Hướng dẫn:**

```
<script>
  const thumbnailify = function (videoLink) {
    const linkUrl = videoLink.prop("href");
    const thumbnailUrl = youtube.generateThumbnailUrl(linkUrl);

    const thumbnailImg = $("");
    thumbnailImg.prop("src", thumbnailUrl);
    videoLink.append(thumbnailImg);

    videoLink.on("click", function (e) {
      e.preventDefault();

      const videoEmbed = $("
```

Bài 3. Làm việc với plugin của jQuery (3 điểm).

- **Mục tiêu:** Biết cách làm việc với plugin của jQuery.
- **Yêu cầu:** jQuery Validation Plugin (<http://jqueryvalidation.org/>) là một trong những plugin được sử dụng phổ biến để tra dữ liệu nhập vào các biểu mẫu. Trong bài tập này, sinh viên được cung cấp một biểu mẫu đăng ký thành viên đơn giản (thư mục ex3) và nhiệm vụ của sinh viên là sử dụng jQuery Validation Plugin để kiểm tra dữ liệu người dùng nhập vào biểu mẫu đã cho với các quy tắc:
 - Người dùng phải cung cấp tất cả các thông tin yêu cầu.
 - Tên đăng nhập có ít nhất 2 ký tự.
 - Mật khẩu có ít nhất 5 ký tự.
 - Trường hợp thư điện tử phải là một địa chỉ email hợp lệ.
 - Người dùng phải đồng ý với các quy định của tổ chức.
 - Trường nhập liệu nào của biểu mẫu không hợp lệ thì (1) làm nổi bật trường nhập liệu bị lỗi và (2) hiển thị một thông báo lỗi phía dưới trường nhập liệu đó.

Tên của bạn	<input type="text" value="Tên của bạn"/>
	<i>Bạn chưa nhập vào họ của bạn</i>
Họ của bạn	<input type="text" value="Họ của bạn"/>
	<i>Bạn chưa nhập vào tên của bạn</i>
Tên đăng nhập	<input type="text" value="Tên đăng nhập"/>
	<i>Bạn chưa nhập vào tên đăng nhập</i>
Hộp thư điện tử	<input type="text" value="Hộp thư điện tử"/>
	<i>Hộp thư điện tử không hợp lệ</i>
Mật khẩu	<input type="text" value="Mật khẩu"/>
	<i>Bạn chưa nhập mật khẩu</i>
Nhập lại mật khẩu	<input type="text" value="Nhập lại mật khẩu"/>
	<i>Bạn chưa nhập mật khẩu</i>
	<input type="checkbox"/> Đồng ý các quy định của chúng tôi
	<i>Bạn phải đồng ý với các quy định của chúng tôi</i>
	<input type="button" value="Đăng ký"/>

Hiển thị thông báo lỗi phía dưới trường thông tin không hợp lệ.

- **Hướng dẫn:**

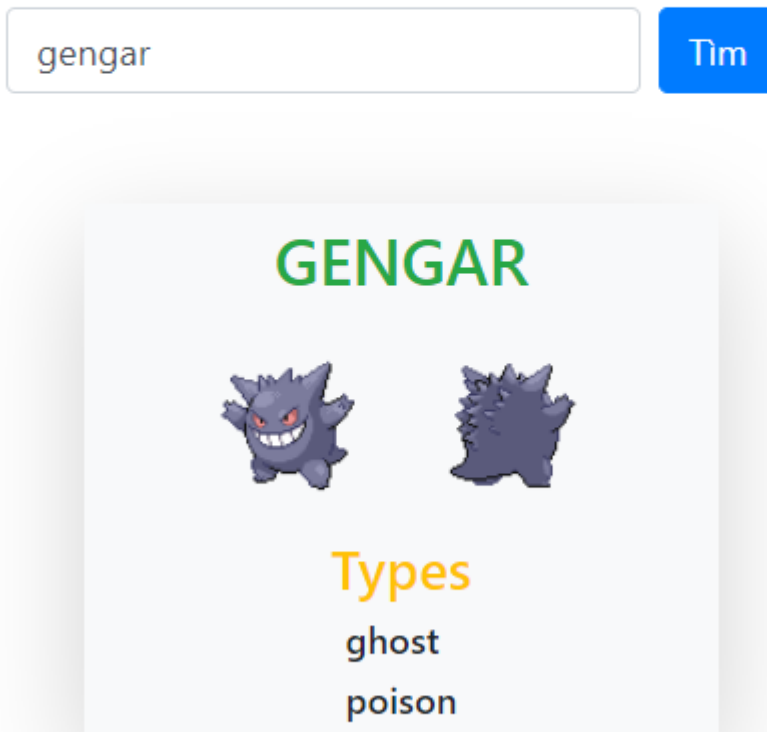
Tài liệu tham khảo cho jQuery Validation Plugin có sẵn tại: <http://jqueryvalidation.org/>

```
<script type="text/javascript">
$.validator.setDefaults({
    submitHandler: function () { alert("submitted!"); }
});
$(document).ready(function () {
    $("#signupForm").validate({
        rules: {
            firstname: "required",
            lastname: "required",
            username: { required: true, minlength: 2 },
            password: { required: true, minlength: 5 },
            confirm_password: { required: true, minlength: 5, equalTo: "#password" },
            email: { required: true, email: true },
            agree: "required"
        },
        messages: {
            firstname: "Bạn chưa nhập vào họ của bạn",
            lastname: "Bạn chưa nhập vào tên của bạn",
            username: {
                required: "Bạn chưa nhập vào tên đăng nhập",
                minlength: "Tên đăng nhập phải có ít nhất 2 ký tự"
            },
            password: {
                required: "Bạn chưa nhập mật khẩu",
                minlength: "Mật khẩu phải có ít nhất 5 ký tự"
            },
            confirm_password: {
                required: "Bạn chưa nhập mật khẩu",
                minlength: "Mật khẩu phải có ít nhất 5 ký tự",
                equalTo: "Mật khẩu không trùng khớp với mật khẩu đã nhập"
            },
            email: "Hộp thư điện tử không hợp lệ",
            agree: "Bạn phải đồng ý với các quy định của chúng tôi"
        },
        errorElement: "div",
        errorPlacement: function (error, element) {
            error.addClass("invalid-feedback");
            if (element.prop("type") === "checkbox") {
                error.insertAfter(element.siblings("label"));
            } else {
                error.insertAfter(element);
            }
        },
        highlight: function (element, errorClass, validClass) {
            $(element).addClass("is-invalid").removeClass("is-valid");
        },
        unhighlight: function (element, errorClass, validClass) {
            $(element).addClass("is-valid").removeClass("is-invalid");
        }
    });
});
</script>
```

Bài 4. Ajax/JSON và jQuery (6 điểm).

- **Mục tiêu:** Biết cách làm việc với Ajax/Json sử dụng jQuery.
- **Yêu cầu:** Trong phần này, sinh viên sẽ sử dụng hàm \$.ajax hoặc \$.getJSON của jQuery để đọc nội dung JSON từ REST API <https://pokeapi.co/api/v2/pokemon/<tên-pokemon>> và chuyển dữ liệu JSON lên các đối tượng DOM. Ứng dụng web hoạt động như sau: người dùng nhập tên một Pokemon vào input và nhấn Tìm hoặc nhấn phím Enter thì ứng dụng sẽ thực hiện lời gọi \$.ajax hoặc \$.getJSON của jQuery để đọc nội dung JSON về pokemon cần tìm từ REST API. Nếu đọc thành công thì hiển thị thông tin pokemon như hình minh họa (sinh viên có thể có thiết kế giao diện khác).

Pokemon Index



- **Hướng dẫn:**

Các trường dữ liệu được gạch đỏ như hình bên dưới là các dữ liệu cần đọc:

<https://pokeapi.co/api/v2/pokemon/gengar>

```
{
  "abilities": [ ... ], // 1 item
  "base_experience": 250,
  "forms": [ ... ], // 1 item
  "game_indices": [ ... ], // 20 items
  "height": 15,
  "held_items": [ ... ], // 1 item
  "id": 94,
  "is_default": true,
  "location_area_encounters": "https://pokeapi.co/api/v2/pokemon/94/encounters",
  "moves": [ ... ], // 117 items
  "name": "gengar",
  "order": 149,
  "past_types": [],
  "species": { ... }, // 2 items
  "sprites": {
    "back_default": "https://raw.githubusercontent.com/PokeAPI/sprites/master/sprites/pokemon/back/94.png",
    "back_female": null,
    "back_shiny": "https://raw.githubusercontent.com/PokeAPI/sprites/master/sprites/pokemon/back/shiny/94.png",
    "back_shiny_female": null,
    "front_default": "https://raw.githubusercontent.com/PokeAPI/sprites/master/sprites/pokemon/94.png",
    "front_female": null,
    "front_shiny": "https://raw.githubusercontent.com/PokeAPI/sprites/master/sprites/pokemon/shiny/94.png",
    "front_shiny_female": null,
    "other": { ... }, // 3 items
    "versions": { ... } // 8 items
  },
  "stats": [ ... ], // 6 items
  "types": [
    {
      "slot": 1,
      "type": {
        "name": "ghost",

```

– Hết –