Department of Computer Science
City University of Hong Kong
CS2204 Fundamentals of Internet Application Development
Course Work No. 2 (CW2)

Learning Outcomes:

- Able to write Javascript and execute them in Web pages;
- Able to enhance visual interest, access data and manipulate elements of Web pages by applying Javascript to appropriate HMTL components and CSS

1. Overview

You will use the Web pages you have prepared in CW1. Enhance their interactivities and make all specified features work. Javascript will be added and you could also change the HTML/CSS to facilitate the functions of Javascript. Detail requirements are listed in the following sections.

2. External JS Library

An external js file - http://courses.cs.cityu.edu.hk/cs2204/cs2204.js should be used in all pages. Functions in the library will be used to simulate login and data access. This library works with all videos and images provided in CW1, however, if you use your own images then editing is required. You have to submit your version of cs2204.js if you make changes to it.

3. Index Page

- the log-in form is hidden originally after the page is initially loaded. After 3 seconds the log-in form "move" in from any border and stays on top the video
- when the log-in form is submitted, call the function *checkLogin(user, password)* in the external library cs2204.js where user and password are obtained from the form input; a true return means user and password are correct and the form submission is sent to *Main* page; display alert error message for a false return and stop the submission

  [Note : the login checking is a simulation, there are 1/4 of chances of getting a false return; you may use your own function to return deterministic true or false for initial testing]

4. Main Page

- use 2 more photos such that on loading/reloading of the page, a random photo is used in the Spotlight block
- call the function *getVideo( )* in the external library to get a random pick of video objects; base on the return video objects, generate the necessary video information under the "Must Watch" title
- the return of *getVideo( )* is an array with the following example JSON format:

```
[{ "title":"Wonders",
   "poster":"wonders.jpg",
   "video":["wonders.mp4", "wonders.ogg"],
    "desc":"See the wonders of mother nature",
    "director":"Adam Smith",
    "cast":["Voluntees from Nature Association"],
    "plot":"Seven wonders of the world ... "
  }, {...}, {...}, {...} ]
```

- set up and write event handlers for all numbers in the table and change the cursor to a "hand" style when the numbers are mouse-over; when table number is clicked, get a new random set of video objects and generate the video information again
- set up the image link to carry forward the video information to *Play* page; the title of a video can be used as the key to identify a video object

5. Play Page

- this page will only be loaded from *Main* page
- call the function *getVideo("all")* in the library to return all video objects available in the library
- use the video identification carried forward to search for the video object
- use the information in the video object to generate the video title, video tag, cast, director and plot HTML
- the duration should actually be obtained from the video when it is loaded and ready to play
- the set of recommended video information (title and image) remains static same as in CW1 but you may need to modify your HTML to include video identification for each of the images
- set up and write event handlers for the images; when they are clicked the video tag and all other video information will be regenerate

6. Assessment

You will be assessed by how much and how well you can apply what have been learnt from the course, some considerations are:-

- the requirements are met
- the Javascript should be clear, tight without redundant code and appropriate techniques are used;
- no Javascript libraries nor frameworks such as jQuery or YUI, etc. should be used;
- sensible use of external, embedded or inline scripts
- show a clear separation of structure, presentation and behavior
- use appropriate HTML tags, classes, ids or CSS to help your Javascript
- your pages should work in more than one browser
- arrange your web site directories properly, e.g. HMTL, style sheets, images, and scripts etc.

7. Due Date and submission

- The due date has been announced in Canvas, the deadline time is 11:55 pm
- Submit a zip file of your web site with appropriate folders set up so that it could be used directly by unzip

8. Miscellaneous information

- All techniques to achieve the requirements can be found in the lecture slides or examples in Canvas
  - Some common techniques include but not limited to: loops; use Array/JSON to store information; get hold of elements by id/tag/CSS selectors; loops to scan through selected tags; change elements' properties (especially styles or innerHTML); use document.write( ) method; create HTML, show or hide elements dynamically and properly set up event handlers, etc.
  - There may be more than one way to write Javascript for a problem
  - It is important to first think of a way to solve the problem before writing your codes

~ End ~