



Homework #2

更多 HTML/JS 練習

Due : 5pm, 2018-03-28

WEB PROGRAMMING 106-2

Programming Assignment #1

完成講義 Lecture Node #2, p18-19 的 JS Practice #2.

基本上就是設計一個跟 RegBall 不一樣的 EvilBall 物件(但兩者應該都要繼承相同的 Ball prototype), 然後可以用鍵盤控制 EvilBall 上下左右的移動(自行選擇四個鍵), 碰到 RegBall 就把它吃掉(i.e. RegBall 不見), 並且在右上角的分數加一。當然, 你可以自行設計讓他變得更好玩一些, 但是請量力而為, 不要最後自己做不出來。

【註】這個作業是參考 Mozilla Developer Network (MDN) 的一個範例: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/Object_building_practice, 大家如果有任何不清楚的地方, 歡迎自行前往參考, 但請自重, 不要去抄襲其他人寫過的 code, 否則就失去 homework 練習的意義了~

Programming Assignment #2

完成交換 wireframe 拿到的網頁設計, 如果看不懂, 請聯絡 wireframe 上的原作者。

這個作業的目的是讓大家體會寫 spec 以及開發別人委託的專案的滋味, 並且可以互相參考彼此的 code。Hope you enjoy it!

Programming Assignment #3

寫一個 Web Programming 的經典 Beginner 範例: TODO List.

底下是基本要求:

1. 可建立數個 **ToDoList**
2. 可修改單一 **ToDoList title** (例如: School, Work, Dance Club, etc)
3. 可刪除單一 **ToDoList**
4. 每個 **ToDoList** 內有數個 **TodoItems**
5. 可新增刪除任意 **TodoItem**
6. 可勾選已完成的 **TodoItem**
7. 統計並顯示單一以及所有 **ToDoList** 上未完成的 **TodoItem** 數量
8. 統計並顯示所有 **ToDoList** 上已完成的 **TodoItem** 數量

只要符合上述基本要求, 網頁的樣貌與功能可自行設計, 但請自行斟酌難度, 以自己可以寫得出來為原則。

其他規定 (for these 3 assignments)

1. 這個作業跟上個作業一樣, 請不要使用現成的框架或是 library (e.g. Angular, React, Bootstrap)。我們的目的是希望大家再練習一次 HTML 以及 JavaScript 的基本語法。
 2. 歡迎使用 ES-6 或是以後的語法, 包含上課沒有教到的部分。
 3. CSS 樣式可以參考(i.e. copy)其他類似專案的樣式(e.g. <http://todomvc.com/>), 但 HTML 以及 JavaScript 請自己寫。
-

4. 以上作業皆為前端網頁設計，大家可以不用管後端，如果有需要向伺服器讀取資料的功能，就先把資料寫死在 code 裡面就好，如果有要填入資料 (e.g. TodoItem)，則直接寫在前端網頁就好(i.e. 不用存到後端資料庫，所以網頁重新整理就不見了)。
5. 一般作業的評分標準是只要有達到基本要求就好，並不會因為特別做很多而分數超過滿分。當然，有自我挑戰收穫必然是自己獲得的，但請量力而為，注意版本控制，繳交上來請務必是可以展示的網頁。

繳交

請開一個母目錄 (yourStudentID_hw2)，並在其下開設三個子目錄，命名為 hw2-1, hw2-2, hw2-3，分別將三個作業的原始碼及相關檔案放置其中，壓縮整個母目錄 (tar zcvf yourStudentID_hw2.tgz yourStudentID_hw2)，再將壓縮檔上傳到 Ceiba (只開放有修課同學)。

檔案超過 20MB 無法上傳的同學，請找一個網路空間 (建議：github) 照上面的目錄安排放置，然後寫一個文字檔將 github repo 的 link 寫在上面，至 Ceiba 交此文字檔即可。當然，檔案沒有超過 20MB 但想要這樣做的同學也可以這樣做。我們在教完 git 之後就會請同學開始使用 github 上傳作業。

請勿遲交，遲交一天扣 1/3 的分數，兩天扣 2/3，三天就可以不用交了。

03/28 課前 Reading/TODOs (請務必事先閱讀，避免上課聽不懂)

1. 安裝 Git, 申請 Github 帳號
 - 請安裝 command line, 避免使用 GUI 工具：<https://git-scm.com/book/zh-tw/v2> (chap 1.4 開始)
2. Git tutorial
 - (30 mins video) <https://www.youtube.com/watch?v=HVsySz-h9r4>
 - Codeschool Try Git: <https://www.codeschool.com/courses/try-git>
3. 安裝 ESLint，確認你習慣的編輯器上可以使用。
4. 預習 React
 - [Getting Started](#)
 - [Tutorial](#)
 - [Thinking in React](#)
 - Tutorial 如果看起來很茫然，建議改成看線上教學影片 (長度都只有 1 - 3 分鐘)
 - [React Fundamentals: Hello World - First Component](#)
 - [React Fundamentals: The Render Method](#)
 - [React Fundamentals: Introduction to Properties](#)
 - [React Fundamentals: State Basics](#)
 - [React Fundamentals: Owner Ownee Relationship](#)
 - [React Fundamentals: Using Refs to Access Components](#)
 - [React Fundamentals: Accessing Child Properties](#)
 - [React Fundamentals: Component Lifecycle - Mounting Basics](#)

- [React Fundamentals: Component Lifecycle - Mounting Usage](#)
- [React Fundamentals: Component Lifecycle - Updating](#)