



Practice Homework #5

React.js 練習 #2 — Calculator

Due : 9pm, Monday, 2021-11-08

作業規定與建議步驟：

1. 這個作業是讓大家再次練習一次 React。關於 Calculator 的詳細規格，請參考你自己手機上的 APP，原則上只要合理、直覺就好。請注意：定義規格也是這個作業希望大家練習的地方。
2. 這個作業規定一定要用 **React.js** 來寫，如果你 **HW#4** 是使用 **class**，那這個作業建議你學習使用 **Hooks** 來寫寫看！這次作業除了 **Calculator** 這個 **container** 本身不要直接去使用網路上可能可以找得到的 **open-source code** 之外，底層的 **Components** (如：**Buttons**) 開放使用第三方的 **UI** 套件，如：**MaterialUI**, **Ant Design** 等。我們不另外提供參考的 **CSS** 檔案。
3. As convention, React APP 的入口檔案是 "src/index.js"，你需要在它裡面呼叫 `ReactDOM.render(...)`。至於 Calculator 的各個 container classes 以及 components, 請按照講義的建議把對應的 JavaScript 檔案放在 `src/Containers` 以及 `src/Components` 目錄裡。
4. Calculator 的基本要求功能如下：
 - 請參考任意一個 Calculator APP (手機、電腦)，並根據他的行為在 `hw5/README.md` 寫清楚你的 Calculator 的 Spec。在 Spec 的一開始請註明你是參考哪個平台上的哪個 Calculator APP。請注意，你實現的功能可能會跟實際的 APP 不太一樣 => 沒有關係，你自己在 Spec 寫清楚就好，甚至你的 UI 也可以因為功能不同而有自己的客製化。
 - 可進行基本的加減乘除，至於有沒有支援先加減後乘除、是否按了下一個 operator 螢幕就會先算出目前的結果、operator 連續按是什麼效果... 等，請以你參考的 APP 為準，然後再 `README.md` 定義清楚。
 - 除了整數，也支援浮點數 (i.e. 小數)
 - 如果計算結果產生 exception, 例如：Divided by 0, overflow 等，請在計算結果的 `<div>` 顯示錯誤訊息
5. Calculator 的進階要求功能如下：
 - 支援輸入錯誤可以有修正鍵
 - 支援如 "MR"/"MC" 等可以記錄目前計算結果的按鍵
 - 支援必要時切換成科學符號表示數字
 - 至少實現三項加減乘除以外的計算功能 (請記得在 `README.md` 說明)
6. 請注意，你的 `hw5` 目錄底下應該會有一個叫做 ".gitignore" 的檔案，它是由 create-react-app 自動產生，定義了「不會」被 push 到 GitHub 上面的檔案，請不要移除 .gitignore。此外，如果你 `hw5` 資料夾中還有其他的檔案是沒有需要 push 到 GitHub 上去的，請記得把他們的檔名也加到 .gitignore 去。
7. 承上，當你在 **review 別人的作業的時候**，由於你 pull 下來的檔案不會有那些定義在 .gitignore 的檔案，你直接 "yarn start" 是不會 work 的 (請不要因此把別人的作業判斷成屍體)！請先輸入 **"yarn" or "yarn install"**，它會根據 "package.json" 把一些檔案下載並安裝好，然後再來 "yarn start"。
8. 請注意，這只是個純前端的 App, 你不用把 Calculator 的計算存在資料庫或是檔案裡頭。換句話說，當你重新整理頁面後，Calculator 內容就會不見了。

9. 再次聲明，practice homework 不會有詳細的規格，「定義規格」也是 practice 的一部分，只要 make sure 你交上來的成品不要讓別人覺得是「屍體」就好，請不用再來問我們「這樣可不可以」、「這個要不要做」之類的問題了。

繳交

請在 deadline (9pm, Monday, 2021-11-08) 之前把你的作業 push 到你 GitHub 上 wp1101 這個 repo 上面 (Note: 我們不接受遲交哦！)。你應該會用到的指令有 `git add hw5`, `git commit -m "你想留的 comment"`, 以及 `git push`. 你也可以用 `git status` 來檢查一下檔案是否都已經成功 push 到 GitHub 上面，或者是登入 GitHub 去檢查。

如有任何的問題，請到 FB 社團的 "HW#5 Q&As" 貼文底下發問，或者是寫信到老師/助教信箱：
eewebprogramming@googlegroups.com。