

ScoreCard DB

CLEAR

JohnMath62

ADD

☒ Name ☐ Subject

Query string...

QUERY

Adding (Ric, Math, 100)
Adding (Ric, English, 83)
Adding (Mary, English, 87)
Adding (John, Math, 62)

Practice Homework #6

React + Axios + Express + Mongoose

Due : 9pm, Monday, 2022-11-07

作業規定與建議步驟：

1. 這個作業是要大家使用 React/Axios/Express/Mongoose 框架，寫出一個簡單的全端應用 “Score Card DB”。
2. 這個作業會提供完整的前端參考程式碼 ([tgz](#) | [zip](#))，但後端請同學自行開發，一些規範條列如後 (也請參考 **Lecture Note #7 “More on Backend + Database”**)。
3. 如同 Homework #5 一樣，請先到 “hw6” 目錄用 “yarn init -y” 來建立一個新專案，然後將 “package.json” 加入：

```
"scripts": {  
  "start": "cd frontend && yarn start",  
  "server": "cd backend && yarn server"  
}
```

換句話說，請將前端與後端的 code 分別放在 “frontend” 與 “backend” 目錄底下，屆時在 “hw6” 底下用 “yarn start” 以及 “yarn server” 來開啟前後端的 apps。

4. **[Frontend]** 如同講義所說，請到 “wp1111/hw6” 目錄下，利用 create-react-app 創建一個 frontend 的 React project，然後到該目錄，移除原先的 “src” 目錄，再將前端的參考程式碼解壓縮，你應該就會看到新的 “src” 目錄了。

```
cd wp1111/hw6  
create-react-app frontend  
cd frontend  
rm -rf src  
tar xzvf hw6-frontend-src.tgz or unzip hw6-frontend-src.zip
```

- 請注意，解壓縮之後應該只會看到新增了一個 “src” 的目錄，如果你解壓縮之後發現被多插入了一層目錄，請將 “src” 搬至 “hw6” 底下，並且移除該多出來的目錄。

5. **[Backend]** 請參考講義 p27 的說明，到 hw6 目錄底下，執行底下事項：

- 建立一個新的 “backend” 目錄
- 到 “backend” 目錄用 “yarn init -y” 建立一個新的專案
- 安裝 cors express nodemon babel dotenv-defaults mongoose 等套件
- 增加一個 .babelrc
- 修改 package.json, 新增：

```
"scripts": {  
  "server": "nodemon src/server.js --ext js --watch src --exec babel-node"  
},
```

- 申請 MongoDB Atlas 帳號 ([link](#))，把 connection string 複製下來 (參考講義 p80)，貼到一個 .env 的檔案，內容為 “MONGO_URL=你複製下來的連結” (see p83)
- 將 .env 複製到一個 .env.defaults 的檔案，然後把連結移除，內容變成 “MONGO_URL=” (see p84)
- 新增一個 .gitignore 的檔案，並將 /node_modules 以及 .env 加入

6. 請參考講義 pp. 90-94 backend 的檔案架構與程式碼 (under "src") :

- **server.js**: Server 的主程式 (entry point), 在 package.json 裡頭應該要有對應的 scripts "nodemon src/server.js..."
- **db.js**: 定義 Mongoose 的連線
- **models/ScoreCard.js**: 定義 ScoreCard DB schema, 並且建立一個 ScoreCard model
- **routes/index.js**: 定義 Express/Router middleware, 作為與前端串接的 API endpoint
- **routes/scoreCard.js**: 實現 ScoreCard 的 RESTful APIs

7. 這次作業的基本要求功能如下 (請參考封面示意圖) :

- 當 "Clear" 按鈕被按下去後，會送出 DELETE request 把 DB 清空，然後在下方空白處印出 "Database cleared"。
- 當 "Name", "Subject" 以及 "Score" 三個輸入框都被輸入值之後，按下 "Add" 按鈕，會把資料寫到 DB :
 - ✓ 如果 {Name, Subject} paired value 已經存在 DB，則取代原先在 DB 的這筆資料，然後在下方空白處印出 "Updating (Name, Subject, Score)" // Name, Subject, Score 請代換成輸入的值
 - ✓ 否則，在 DB 新增一筆資料，就印出 "Adding (Name, Subject, Score)"。
- 在 Query string 不為空字串的情況下，當 Query 按鈕按下去後，會根據 Name/Subject 二選一的 radio button 選擇 Name or Subject 作為 query 條件，然後根據輸入 Query String 的內容去 DB 把符合條件的資料搜尋回來，在下方空白處印出搜尋出來的結果(一筆一行)，"Found card with name: (Name, Subject, Score)", or "Found card with subject: (Name, Subject, Score)"。如果找不到，就印出 "QueryType (QueryString) not found!" // 例如: "Name (Ric) not found!"
- Query Input 下方為 message console。在印出訊息/資料時請繼續印在之前的訊息之下，而當 "Clear" 動作被執行之後，請清除 console 中的所有訊息並印出 "Database cleared"。如果在 query 時有任何的錯誤訊息，也是一樣印在這邊。

8. 這次作業的進階要求功能建議如下 (只要做到 3 項即可認定為 Perfect) :

- ✓ 將 "Add" 與 "Query" 分成兩個 tabs, 各自按下時會切換不同的 views (取代原來的 console)
- 在 "Add" 的 view, 跟原來的基礎功能一樣可以輸入 { Name, Subject, Score }, 然後下方的 console 除了顯示 "Updating (Name, Subject, Score)" 之外，也顯示出新增完之後這個人 (Name) 的所有資料，以表格 (table) 的方式顯示
- 在 "Query" 的 view, 跟原來的基礎功能一樣可以輸入 { QueryType, QueryString }, 然後下方以表格 (table) 的方式顯示搜尋結果。如果搜尋不到，一樣顯示 "QueryType (QueryString) not found!"
- ✓ 在上述顯示的 table, 新增可以根據某個欄位來排序的功能 (遞增 or 遞減)，這部分的 UI/UX 請自行設計，但此排序應該是前端的功能，不用也不要將排序的結果寫回後端
- 任何 UI/UX 的改進 (請在 README.md 中說明) 結果我做錯了...我做成整個data的grid
好好笑

9. 如果你 hw6 資料夾中有些檔案是沒有需要 push 到 GitHub 上去的，請記得把他們的檔名加到 .gitignore 去。

10. 以上說明，如有不夠詳盡之處，請自行判斷與定義，不用問我們「這樣可不可以」、「這個要不要做」之類的問題，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，但為了讓 reviewer 能夠正確地使用你的 app, 請儘可能地在 README.md 中提供必要的說明。

繳交

請在 deadline (9pm, Monday, 2022-11-07) 之前把你的作業 (i.e. hw6 整個目錄) push 到你 GitHub 上 wp1111 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 hw6` (i.e. 把整個 hw6 一次加進去), `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。請注意，你不用建立 "review" 這個資料夾，助教會在指派 review 作業時再幫你加進去你的 repo。

如有任何的問題，請到 FB 社團的 "HW#6 Q&As" 貼文底下發問，或者是寫信到老師/助教信箱：eewebprogramming@googlegroups.com。