

ICE4016 데이터베이스설계

<Week 11 practice>

보고서 작성 서약서

1. 나는 타학생의 보고서를 베끼거나 여러 보고서의 내용을 짜집기하지 않겠습니다.

2. 나는 보고서의 주요 내용을 인터넷사이트 등을 통해 얻지 않겠습니다.

3. 나는 보고서의 내용을 조작하지 않겠습니다.

4. 나는 보고서 작성에 참고한 문헌의 출처를 밝히겠습니다.

5. 나는 나의 보고서를 제출 전에 타학생에게 보여주지 않겠습니다.

나는 보고서 작성시 윤리에 어긋난 행동을 하지 않고 정보통신공학인으로서 나의 명예를 지킬 것을 맹세합니다.

2023년 11 월 14 일

학부 정보통신공학

학년 4

성명 이재영

학번 12181816



1. 개요

* Inha DB를 활용해서 다음과 같은 기능을 가진 페이지 제작
* 로그인을 했을 때 ‘Enrolment’를 클릭하면 수강신청 페이지로 이동
* 수강신청 페이지에서 Apply로 수강신청 기능 구현
  + Apply 버튼을 누르면 DB에 수강신청 내용이 반영
  + Remaining participants = 0 이면 수강신청 불가
  + Course completion list (이미 수강신청한 과목)은 수강신청 불가

1. 상세 설계 내용

* 텍스트, 스크린샷, 폰트이(가) 표시된 사진

  자동 생성된 설명기존 사용하던 Inha DB의 Student entity 수정
  + 로그인을 위해 Password 속성 추가, Password 값은 각 학생의 전화번호 11자리
* 텍스트, 스크린샷, 폰트이(가) 표시된 사진

  자동 생성된 설명텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

  자동 생성된 설명로그인 성공 시 ‘Enrolment’ 클릭하면 수강신청 페이지로 이동
  + 텍스트, 스크린샷, 폰트이(가) 표시된 사진

    자동 생성된 설명입력된 아이디(학번)과 패스워드(전화번호)는 vars에 저장하고 student 테이블에 대한 정보를 users에 저장한다. Users를 user로 맵핑하여 user의 아이디와 패스워드가 같은 경우가 있으면 로그인 여부를 의미하는 checkLogin은 True를 가지고, whoAmI은 아이디 값을 가지게 된다. 해당 로그인 정보는 1시간동안 쿠키가 저장되며, 1시간동안은 로그아웃을 하지 않는 이상 웹페이지를 껐다 켜도 해당 아이디로 로그인이 유지된다.
  + 텍스트, 폰트, 스크린샷이(가) 표시된 사진

    자동 생성된 설명텍스트, 스크린샷, 폰트이(가) 표시된 사진

    자동 생성된 설명텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

    자동 생성된 설명쿠키에 user 값이 있으면 main.hbs가 렌더링되며 없다면 다시 로그인 페이지로 가게 된다.
  + 위 코드는 main.hbs의 일부이며 로그인이 성공했을 때 사용할 수 있는 기능이다. Logout 박스와 enrolment 아이콘을 클릭하면 onclick 키워드에 의해 각각 /logout, /sugang 페이지로 이동한다.
* 텍스트, 스크린샷, 폰트이(가) 표시된 사진

  자동 생성된 설명수강신청 페이지에서의 수강신청 기능
  + 로그인한 학생이 수강중인 과목들을 출력해주는 getClass 함수 구현,
    - 로그인한 학생의 id(학번)을 인자로 주어 웹 페이지이 출력해야 하는 해당 학생 수강 수업 정보를 가져옴
    - Student\_has\_Class (학생과 수업의 관계 테이블)에 join하여 해당 student\_id와 로그인 학생의 id와 동일한 튜플을 가져옴
    - 텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

      자동 생성된 설명각 수업은 department\_id를 속성으로 가지기 때문에 해당 속성을 활용하여 Department 테이블에 join하여 department 테이블의 id와 class 테이블의 department\_id가 동일한 튜플의 department name을 가져와서 Opening\_departments에 맵핑한다.
  + 로그인한 학생이 수강하는 수업을 제외한 나머지 수업을 출력하는 getCountParticipant 함수 구현
  + getClass 함수와 비슷해 보이나 where not exists절 (서브 쿼리)의 결과가 존재하면 해당 학생이 이미 수강중인 수업이기 때문에 제외된다.
  + COALESCE 키워드를 이용하는데 먼저 입력된 값이 NULL이 아니면 출력한다. 즉 등록된 학생 수가 없어서 NULL이 되면 뒤에 입력된 0이 출력되는 것이다. 이 부분이 필요한 이유는 후에 Remaining\_participants (남은 가능 인원)을 계산할 때 필요하기 때문이다.
  + Opening\_departments의 맵핑은 getClass와 동일하며 Remaining\_participants는 (Class 테이블의 최대 가능 인원 수 – 계산된 등록 인원 수)를 맵핑해준다. 그래서 맵핑된 부분을 포함하여 select된 부분을 반환한다.
  + 수강텍스트, 스크린샷, 폰트이(가) 표시된 사진

    자동 생성된 설명 신청하는 addClass 함수
    - 텍스트, 스크린샷, 폰트이(가) 표시된 사진

      자동 생성된 설명텍스트, 스크린샷이(가) 표시된 사진

      자동 생성된 설명텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

      자동 생성된 설명현재 로그인한 학생의 id와 수업의 id를 이용해 insert문으로 student\_has\_class 테이블에 데이터를 추가한다.
  + /sugang 페이지로 들어가면 진행되는 부분
    - req.cookies.user가 존재한다는 것은 로그인되어 있다는 것과 동일한 의미를 가진다고 생각할 수 있다. 해당 정보를 이용해서 getClass와 getCountParticipant 함수에 인자로 넣어 호출해주면, 학생이 이미 수강중인 수업과 수강 중인 수업을 제외한 수업(남은 가능 인원 연산)에 대한 정보를 classInformation, allClass 각각의 변수로 저장해준다. 해당 변수들을 select.hbs로 보내 렌더링한다. (hbs에서의 #each 뒤 이름과 맞춰주는 것이 중요하다)

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

* + Apply 버튼 클릭 후 실행되는 부분
    - 버튼으로 보내지는 Id 값은 class의 id값이며 req.body.applyBtn에 저장된다. 먼저 수강중인 수업을 제외한 나머지 수업들에 대한 정보 (getCountParticipant)와 현재 수강중인 수업에 대한 정보를 가져온다.
    - 텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

      자동 생성된 설명나머지 수업의 데이터에서 class의 id를 가져와서 나머지 수업들이 존재하고 해당 수업들의 수강 가능 인원이 존재하면 addClass 함수를 호출해서 student\_has\_class 테이블에 insert하도록 한다. 이후 추가된 수업을 반영해서 해당 학생이 수강중인 수업에 대한 정보를 구하고 (getClass), 나머지 수업에 대한 정보를 구한다. (getCountParticipant) 해당 정보들을 select.hbs로 렌더링한다.
    - 만약 Apply 버튼을 클릭했을 때 해당 수업의 가능인원이 0이어서 더 이상 수강 불가능 하다면 “수강 가능 인원을 초과했습니다! “의 메시지 창이 뜨게 된다. 위 코드는 select.hbs에 추가한 코드이며 message의 값이 존재하는 경우에 alert 메소드를 이용해 창이 뜨게 한다.

1. 실행 화면

* 현재 Inha DB 상의 정보
  + 텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

    자동 생성된 설명텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

    자동 생성된 설명텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

    자동 생성된 설명스크린샷, 텍스트, 라인, 폰트이(가) 표시된 사진

    자동 생성된 설명student 테이블의 Id, Password 속성이 각 학생의 아이디와 패스워드를 나타내는 값이며, 학생과 수업 관계 테이블인 student\_has\_class에는 위와 같이 튜플이 존재한다. class 테이블에 존재하는 수업은 위와 같다.
* 텍스트, 스크린샷, 폰트이(가) 표시된 사진

  자동 생성된 설명텍스트, 스크린샷, 폰트, 로고이(가) 표시된 사진

  자동 생성된 설명텍스트, 스크린샷, 폰트, 로고이(가) 표시된 사진

  자동 생성된 설명npm run start로 웹앱을 실행하면 위와 같이 접속된다.
* Student 테이블의 Id와 Password를 이용해 정상적으로 로그인되는 것을 확인할 수 있다.
* 텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

  자동 생성된 설명텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

  자동 생성된 설명12181816, 01020226965로 로그인 후 ‘Enrolment’를 클릭하면 Student\_has\_Class에 존재하는 튜플처럼 수강중인 수업은 위 표에, 나머지 수업들에 대한 정보와 남은 수강 가능 인원은 아래 표에 출력되는 것을 볼 수 있다.
* 텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

  자동 생성된 설명“Internet Protocol”과목을 Apply한 결과이다. 수강중인 수업 표에 출력되고, 아래의 나머지 수업에 대한 정보에는 삭제되어 출력되는 것을 볼 수 있다.
* 이후 12180000, 01000010001으로 로그인하여 수강신청 페이지에 접속해보았다. 이전에 12181816 학생이 수강중인 두 개의 수업에는 남은 가능 인원이 1씩 감소되어 나머지 스크린샷, 소프트웨어, 멀티미디어 소프트웨어, 그래픽 소프트웨어이(가) 표시된 사진

  자동 생성된 설명수업 표에 출력되는 것을 확인할 수 있다.
* 텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

  자동 생성된 설명남은 수강 가능 인원이 0인 과목을 Apply하면 위와 같은 메시지창이 뜨며 확인 클릭 시에 다시 수강신청 페이지로 돌아가게 된다.
* 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

  자동 생성된 설명수강 가능한 두 과목을 Apply해주었다.
* MySQL을 실행하여 inha\_week11의 학생 수업 관계 테이블을 확인해보니 두 학생의 아이디의 Apply에 따라 데이터가 잘 적용된 것을 확인할 수 있다.

1. 결론

* Javascript상에서 연산한 데이터를 hbs에 명시된 데이터에 맵핑하여 활용할 수 있었습니다.
* hbs에서의 onclick, alert 등의 새로운 키워드에 대해 공부해보면 웹 페이지 상에서 다양한 기능을 추가할 수 있었습니다.
* 목표에 따른 결과를 얻을 수 있었지만 어떤 부분에서는 복잡한 쿼리를 사용하여 가독성이나 최적화 시점에서는 부족하다고 느꼈습니다. Query Optimization을 통해 최적화된 코드로 수정이 필요하다는 생각이 들었습니다.