

ICE4016 데이터베이스설계

<Week 5 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 10 월 06 일

학부 정보통신공학

학년 4

성명 이재영

학번 12181816



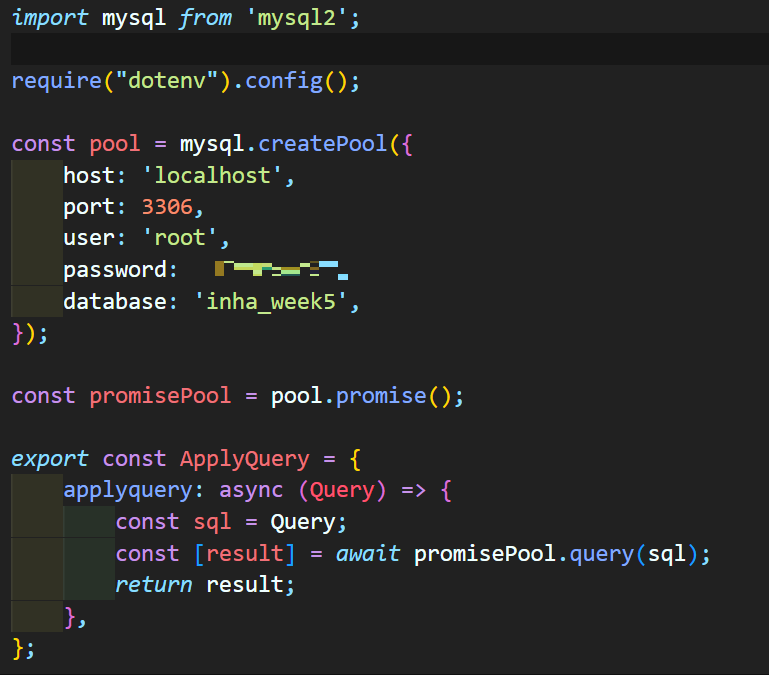
1. 개요

* 4주차 과제에서 만든 데이터베이스에 Class, Club, Employee 테이블 추가
* 기존 DB와의 관계 구성
* 각 테이블에 5개 이상의 데이터 추가
* 확장한 DB를 Express와 연동하여 DB 확인

1. 상세 설계 내용

* 텍스트, 스크린샷, 폰트이(가) 표시된 사진

  자동 생성된 설명추가한 Entity: Class, Club, Employee: 각 테이블의 Id를 Primary Key로 설정
* 텍스트, 스크린샷, 폰트이(가) 표시된 사진

  자동 생성된 설명추가된 relationship table:
  + student와 class 관계: 수업이 없는 학생이 존재할 수도 있기 때문에 비 식별관계로 설정, 또한 학생은 여러 수업을 들을 수도 있고, 수업에는 여러 학생들이 존재하기 때문에 N:M 관계로 구성
  + student와 club의 관계: 동아리에 가입하지 않은 학생이 존재할 수도 있기 때문에 비 식별관계로 설정, 학생은 여러 동아리에 가입할 수 있으며, 동아리에는 여러 학생들이 소속될 수 있기 때문에 N:M 관계로 구성
  + employee와 class의 관계: employee를 고용인(직원)으로 이해하여 관계 테이블 형성, 고용인에는 교수가 포함되며 교수가 아닌 수업을 진행하지 않는 직원도 포함되므로 비 식별관계로 설정, 교수라면 여러 수업을 담당할 수도 있고, 한 수업을 여러 교수님이 담당할 수 있어서 N:M 관계로 구성
* Node.js의 express로 웹 어플리케이션 구현
  + 텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

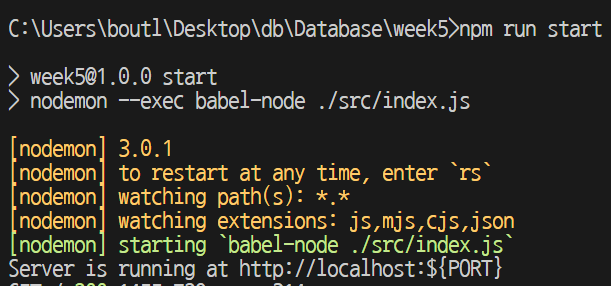
    자동 생성된 설명텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

    자동 생성된 설명sql.js: MySQL과 연결하기 위한 pool을 생성하고, javascript에서 비동기 작업을 다루기 위해 promise 객체를 활용한다. 웹 서버상에서 입력된 쿼리를 MySQL로 전달하여 작업을 진행한다.
  + home.js: express의 장점은 router 모듈로 분리하여 가독성 향상 및 유지 보수에 용이하다는 것이다. 위 코드는 해당 부분이며, router를 선언하고 post 메소드를 통해 웹 어플리케이션에 내가 입력한 쿼리를 MySQL로 전달하고 해당 쿼리에 대한 작업 결과를 출력하거나 잘못된 쿼리일 경우에는 에러를 발생하도록 한다. (라우터 레벨 미들웨어)
  + 텍스트, 스크린샷, 폰트이(가) 표시된 사진

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

    자동 생성된 설명index.js: 지정된 경로에 마운트 하여 어플리케이션 레벨 미들웨어를 작동하는 코드이다. hbs 파일형식의 템플릿 엔진을 사용하도록 지정한다. 이후 app.listen 메소드로 포트번호 3000번의 웹 어플리케이션을 실행한다.
  + home.hbs: hbs 확장하는 handlebars.js라는 자바스크립트의 템플릿 엔진 중 하나이며, 템플릿 엔진이라는 것은 프로그램 로직과 프레젠테이션 계층을 분리하기 위함이다. 분리함으로써 코드의 가독성이 더 좋아진다. 웹 어플리케이션이 실행되면 웹 창에서 어떠한 프레젠테이션을 보이는 지 구성하는 코드이다. 아래 실행 화면에서 확인할 수 있다.

1. 실행 화면

* npm run start를 이용해 express 실행 🡪 Server is.. 구문 확인 후 <http://localhost:3000> 접속
* 텍스트, 스크린샷, 폰트, 대수학이(가) 표시된 사진

  자동 생성된 설명텍스트, 스크린샷, 폰트이(가) 표시된 사진

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

  자동 생성된 설명기존 엔티티를 제외한 추가된 엔티티의 속성 확인
* 텍스트, 폰트, 스크린샷, 대수학이(가) 표시된 사진

  자동 생성된 설명텍스트, 폰트, 스크린샷이(가) 표시된 사진

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

  자동 생성된 설명select문을 활용해 각 엔티티에 삽입된 데이터 확인
* 텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

  자동 생성된 설명student 엔티티에 외래키 속성으로 department의 id를 추가하고 department.id가 2인 학생의 이름과 이메일 출력
* 텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

  자동 생성된 설명room 엔티티에 외래키 속성으로 building의 id를 추가하고 building.id가 100인 room의 이름과 수용 인원 수 출력

1. 결론

* Worch bench로 도형과 선을 이용해서 Schema를 구성하는 것은 직관적이어서 쉬웠지만, 스크립트를 이용해 코드를 작성하는 부분에서 아직까지 조금 헷갈리는 것 같습니다. 특히 1대1, 1대N, N대M 관계에서 각각의 관계를 가질 때 외래키를 설정하는 것에 대해 많이 찾아보았습니다.
  + 1대1 관계: 자식 테이블은 별도의 primary key를 설정하지 않고 부모 테이블의 primary key를 참조한 foreign key로 자식 테이블의 primary key
  + 1대N 관계: 자식 테이블은 해당 테이블만의 primary key를 설정하고 부모 테이블의 primary key를 foreign key로 사용
  + N대M 관계: 관계 테이블을 생성하여 두 테이블 간의 primary key를 foreign key로 불러와 두 key를 primary key로 사용
* JOIN 키워드가 결합된 SELECT문을 사용해보면서 두 테이블 간의 관계에서 원하는 데이터를 확인해볼 수 있었습니다.