

## [ Spring JPA 실습 2 ]

메일제목 : 스프링실습13-XXX

압축 소스명 : 소스48.zip

3월 27일에 구현했던 소스를 가져다가 JPA 에 알맞게 변경하고 추가하여 구현한다.

(1) vo 패키지에 Student 라는 엔티티 클래스를 생성한다.(setter, getter, toString 필수)  
student 라는 DB 테이블은 name 과 score 라는 컬럼을 가지고 있는데 name 이 PK 이다.

(2) dao 패키지에 StudentDAO 클래스를 구현한다.

모든 DB 연동은 JPA 기술을 사용한다.

boolean insertStudent(Student entity)

List<Student> getAllStudent() → 데이터가 없으면 비어있는 리스트 리턴

Student getScore(String name) → 존재하지 않는 학생명으로 호출하면 null 을 리턴

boolean updateStudent(String entity)

boolean deleteStudent(String name)

(2) controller 패키지에 StudentController 클래스를 구현한다.

void printAll() --> StudentDAO의 getAllStudent()를 호출하고 결과를 화면에 출력한다.

void printScore(String name) --> StudentDAO의 getScore()를 호출하고

리턴 결과가 null 이면 "XX 학생은 존재하지 않습니다."를 출력한다.

아니면 "XXX 학생의 점수는 XX 입니다"를 출력한다.

void insert(String name, int score) --> StudentDAO의 insertStudent()를 호출한다.

리턴결과가 true 이면 "입력 성공"을 출력한다.

리턴결과가 false 이면 "입력 실패"를 출력한다.

void update(String name, int score) --> StudentDAO의 updateStudent()를 호출한다.

리턴결과가 true 이면 "XX 학생의 점수를 변경했습니다."을 출력한다.

리턴결과가 false 이면 "XX 학생은 존재하지 않습니다."를 출력한다.

void delete(String name) -----> StudentDAO의 deleteStudent()를 호출한다.

리턴결과가 true 이면 "XX 학생의 데이터를 삭제했습니다."을 출력한다.

리턴결과가 false 이면 "XX 학생은 존재하지 않습니다."를 출력한다.

(3) mainview 패키지에 StudentApp 클래스를 구현한다.  
수행을 시작하면 다음 메시지를 출력한다.

처리하려는 기능을 선택하세요.

1. 학생 정보 출력
2. 학생 정보 입력
3. 학생 정보 삭제
4. 학생 정보 수정
5. 학생 점수 확인
6. 종료

입력 :

1을 입력하면 StudentController 의 printAll() 호출

2를 입력하면 학생의 이름과 점수를 표준 입력 받아 StudentController의 insert() 호출

3을 입력하면 삭제하려는 학생 이름을 입력받아 StudentController의 delete() 호출

4를 입력하면 수정하려는 학생 이름과 점수를 입력받아 StudentController의 update() 호출

5를 입력하면 점수를 확인하고 싶은 학생의 이름을 입력받아 StudentController 의  
printScore() 호출

6을 입력하면 프로그램 수행으로 종료한다.

6을 제외하고 1번부터 5번의 경우엔 기능을 수행한 다음 다시 메뉴리스트를 출력하여  
반복 처리한다.

소스 구현 순서 : Entity -> DAO -> Controller -> MainView