

데이터베이스설계 과제 #1

제 목

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.

2020년 9월 26일

학부 정보통신공학과

학년 3

성명 서혜빈

학번 1218178



1. 개요

* 간략한 설계 예정 내용

<isert>

우선 insert.html 에서 학번, 이름, 학과, 학년, 복수전공, 입학일, 이메일을 입력하고, 입력 버튼을 클릭한다

그러면 창이 넘어가며 insert 가 제대로 수행되었을 때, insert succss 문을 출력하고, table의 전체 내용을 출력한다.

<select>

학번, 이름, 학과, 학년, 복수전공, 입학일, 이메일 중 옵션을 선택 후, 검색할 내용을 입력한다.

만일 학번 옵션 선택 후, 12181785 을 입력하고 확인을 누르면, 학번이 12181785 인 column의 내용을 출력한다.

아무것도 입력하지 않고, 확인을 클릭할 경우 전체 column이 출력되도록 한다.

1. 상세 설계내용

<INSERT>

* Insert.html

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert</title>

</head>

<body>

<form action = *"/project1/InsertStudent"* accept-charset=*"utf-8"* name = *"person\_info"* method = *"get"*>

<fieldset style = "width:*150*">

<legend> Insert </legend>

학번 : <input type = *"text"* name = *"studentnumber"* required/><br>

이름 : <input type = *"text"* name = *"name"* required/><br>

학과 : <input type = *"text"* name = *"major"*/><br>

학년 : <input type = *"text"* name = *"grade"*/><br>

복수전공 : <input type = *"text"* name = *"doublemajor"*/><br>

입학일 : <input type = *"text"* name = *"admissiondate"*/><br>

이메일 : <input type = *"text"* name = *"email"*/><br>

<input type = *"submit"* value = *"입력"*/><br>

</fieldset>

</form>

</body>

</html>

학번은 studentnumber, 이름은 name, 학과는 major, 학년은 grade, 복수전공은 doublemajor, 입학일은 admissiondate, 이메일은 email 로 설정하였다.

학번과 이름은 table에서도 not null 로 선언하여 반드시 입력을 받아야한다. 따라서 여기서도 required 를 선언해줌으로써 반드시 입력을 받아야만 하도록 명령어를 썼다.

* InsertStudent.java

우선 가장 처음에는, studesign database에 username과 user passwd 를 줌으로써 연결했다.



Int studentnum = Integer.parseInt(request.getParamter(“studentnumber”));

에서 request.getParameter(“studentnumber”) 는 html 에서 선언한 name이 studentnumber인 것을 가져온다는 것이다.

그리고 Integer.parseInt는 가져온 값을 int로 변환하는 것이다.

Html 에서의 studentnumber는 text로, int형이 아니므로 int형으로 변환해주었다.

그리고 이 값을 studentnum 에 저장한다.

나머지 변수들도 마찬가지로 설계하였다.



String sql은 mysql에 입력해줄 문들을 저장하는 변수이다.

Insert를 할 것임으로 insert into tablename values ( … ) 을 저장한다.

위에서 저장한 변수들을 sql에 넣어준다.

그리고 executeUpdate(sql); 을 통해, mysql 에 sql 변수에 저장한 내용을 입력해주고, table에 insert 한다.

executeUpdate는 insert를 위한 명령어이다.



그리고 sql 에 select \* from studentinf 를 넣어주고, executeQuery 명령어를 줌으로써, 모든 column을 출력한다.

while문은 table 에서 다음 column이 없을 때까지 출력을 반복하도록 하였다.

(mysql 에서의 select \* from studentinf 와 같은 명령어 및 실행)

<SELECT>

* Select.html

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Select</title>

</head>

<body>

<form action = *"/project1/SelectStudent"* accept-charset=*"utf-8"* name = *"person\_info"* method = *"get"*>

<fieldset style = "width:*150*">

<legend> Select </legend>

<select id = *"search"* name = *"search"*>

<optgroup label = *"Select"*>

<option value = *""*>선택하세요</option>

<option value = *"studentnum"*>학번</option>

<option value = *"name"*>이름</option>

<option value = *"major"*>학과</option>

<option value = *"studentgrade"*>학년</option>

<option value = *"doublemajor"*>복수전공</option>

<option value = *"admissiondate"*>입학일</option>

<option value = *"email"*>이메일</option>

</optgroup>

</select>

<input type = *"text"* name = *"textarea"*>

<input type = *"submit"* value = *"검색"*/>

</fieldset>

</form>

</body>

</html>

Select.html 의 경우, 옵션을 주었다. 각 옵션의 value값은 table과 같은 name을 주었다. 그리고 select의 id 및 name 은 search로 주었다. 이는 아래의 SelectStudent.java 에서 사용될 것이다.

그리고 옵션 옆에는 text를 입력할 수 있는 공간을 주고, 이 공간에 입력되는 내용을 textarea로 선언하였다.

예를 들어, 이름이라는 옵션을 선택하고 mouse를 입력한다고 해보자.

그러면 mysql에는 name = mouse 인 column을 select 하라는 명령어가 갈 것이고, 명령어에 따라 name = mouse 인 column 만을 출력할 것이다.

* SelectStudent.java



가장 먼저, null 인 str 변수를 선언한다.

그리고 area 변수에는 textarea를 가져와 넣어준다. 즉 html 에서 입력된 내용을 area에 저장한다.

Search 변수에는 search를 가져와 넣어준다. 즉 선택된 옵션의 value를 가져와 넣어준다. 예를 들어 학번이 선택되었다면 search에는 studentnum을 가져와 넣어준다.

그리고 html에서 아무것도 입력이 되지 않았다면 table의 모든 column이 출력되도록 했다.

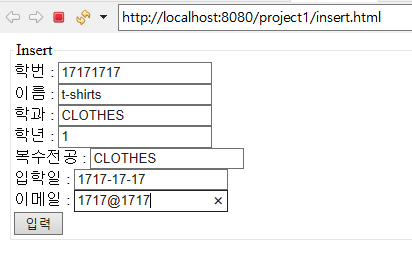


다음 column이 비었을 때까지 값들을 출력해주는 문이다.

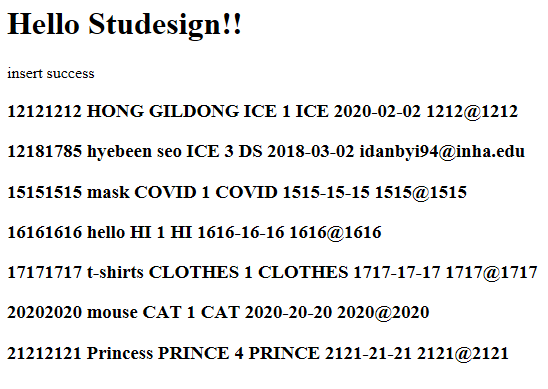
Select \* from studentinf where name like ‘%mouse%’ 가 mysql 에 입력되어 name이 mouse 인 column만 내보냈다면, while 문도 mouse column만 돌아가 mouse만 출력한다.

1. 실행 화면

<INSERT>



각 text 에 내용을 입력한 후, 입력을 클릭한다. 단, 학번과 이름은 반드시 입력해야한다.

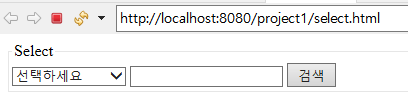


위에서 입력한 17171717이라는 학번을 가진 column이 제대로 insert 되었음을 확인할 수 있다.



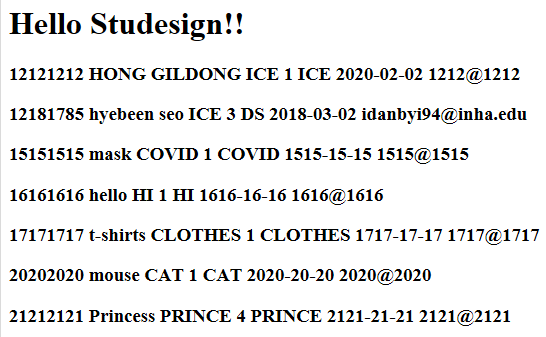
Mysql 에서도 확인한 결과, 잘 insert 되었다.

<SELECT>

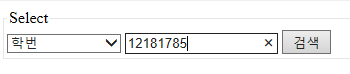


Select.html 첫 실행화면이다.

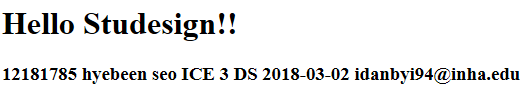
아무것도 입력하지 않고, 검색을 클릭한다.



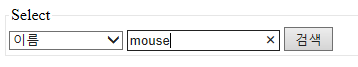
위 사진과 같이 studentinf table의 모든 column이 select 되었음을 확인할 수 있다.



이번에는 학번 옵션을 선택한 후, 12181785 를 검색한다.



12181785 라는 학번을 가진 column의 전체 내용이 select 되었음을 확인할 수 있다.



이번에는 이름 옵션을 선택한 뒤, mouse를 검색한다.



Mouse라는 이름을 가진 column의 전체 내용이 잘 select 되었다.

1. 분석 및 결론

Insert 를 하면서, insert.html에서 입력되는 studentnumber, 즉 학번은 text 이므로, int 형이 아니다. 따라서 parseInt 명령어로 int로 바꿔주는 과정이 중요함을 알 수 있었다.

그리고 table에서 not null 인 것은, html 에서도 required로 반드시 입력하도록 해야함을 알게 되었다.

executeUpdate는 insert, executeQuery는 select 를 위한 명령어임을 알게 되었다.

또한 select 를 하면서, option 의 value들을 가져오는 방법에 대해 조금 헤맸다.

처음 select.html 을 작성했을 때, 아래와 같이 선언했다.

<select name = *"search"*>

그리고 selectStudent.java 를 작성하며 option 을 가져올 때 search를 가져왔지만, 오류가 발생했다. Search 값을 출력해본 결과, null이 나왔다.

이 결과를 통해 html 에서 name = “search” 만 선언하고, search를 가져오는 것은 안된다는 것을 알게 되었다. 그래서 아래와 같이 id를 추가해주었다.

<select id = *"search"* name = *"search"*>

그 결과, SelectStudent.java에서 option의 value들을 잘 인식했다.

Select 할 때 모든 column을 출력하는 것 뿐만 아니라, 사용자가 찾고 싶은 column만 찾을 수 있도록 하고 싶었다. 이는 select \* from tablename where ? like ‘%?’ 명령어를 사용하면 된다. 첫 번째에는 name, major 등과 같이 사용자가 선택한 option value를 주었다. 그리고 두 번째에는 사용자가 text 창에 입력한 내용을 주었다.