SQL injection

Many talk of this topic The major problem is that you if you have some knowledge of sql querres - you can cheat the system by e.g. logging in.

See http://www.w3schools.com/sql/sql_injection.asp

One page describe some steps to take to prevent sql-injection (https://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet)

They descibe three approaches:

1. Prepared statements
2. Stored procedures
3. 'clean' input

To try this you is to set up this environment.

Step 1: Create a databsase for a login

Make a table Login having username and password

e.g

```
create table login(
    username varchar(20) not null primary key,
    password varchar(30) not null
)
```

Step 2 Make a simple project to check username+password

Implement a simple java program which can check a username and password to the database.

(evt. see this netbeans example (with out a database): SQLInjectionTest.zip)

Change the database properties to match your database.

Step 3 Prevent by prepared statement

Make a new method (e.g 'checklogin2') to check username and password - but this time use a prepared statement

How does this goes - can you still cheat?

Step 4 Prevent by stored procedure

Create a stored procedure in your database to make the check with two parameters in (username - password) and one parameter out (the result).

Make a new method (e.g 'checklogin3') to check username and password - but this time call the stored procedure

How does this goes - can you still cheat?

Help:

Stored procedure http://www.mysqltutorial.org/stored-procedures-parameters.aspx
call stored procedures http://docs.oracle.com/javase/tutorial/jdbc/basics/storedprocedures.html

Step 5 Prevent by Cleaning input

Make a new method (e.g 'checklogin4') to check username and password - but this time 'clean' the input
1) use regular expression or 2) ESAPI (from owasp) see https://www.owasp.org/index.php/ESAPI

How does this goes - can you still cheat?