**SQL Injection problem**

1. “--” Symbol acts as comment of SQL query if it is part of SQL query, then Database engine stops executing the portion of SQL query that is commented.

Example:

SELECT COUNT(\*) FROM LOGIN WHERE USERNAME=’sankar’--AND PASSWORD=’12345’";

1. Supplying special SQL instructions along with input values of the application like (--) and making then part of SQL query to change SQL query behaviour and application behaviour is called SQL Injection Problem.
2. SQL instruction like (--) that is participated in SQL query compilation will be recognized as SQL instruction during execution.
3. Simple Statement Object makes Database software to compile SQL query having input values.
4. PreparedStatement Object, the Software compiles then SQL query without input values, means PreparedStatement is precompiled.

Example:



Example Simple Statement Problem:

package com.nt.jdbc;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.Scanner;

public class LoginAppTestProblem {

public static void main(String[] args) {

Connection con=null;

Statement st=null;

String query=null;

ResultSet rs=null;

int count=0;

Scanner sc=null;

try {

//read Inputs

sc=new Scanner(System.in);

System.out.println("Enter UserName: ");

String user=sc.nextLine();

System.out.println("Enter Password: ");

String pass=sc.nextLine();

//Convert input values as required for the SQl Queries

user="'"+user+"'";

pass="'"+pass+"'";

//1.Register type-4 jdbc driver software

Class.forName("oracle.jdbc.driver.OracleDriver");

//2.Establish the connection

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger");

//3.create statement object

st=con.createStatement();

//4.prepare sql Query

query= "SELECT COUNT(\*) FROM LOGIN WHERE USERNAME=+"+user+"AND PASSWORD="+pass;

//5.send and execute SQl query in Database software

rs=st.executeQuery(query);

//6.process the ResultSet Object

if(rs!=null) {

rs.next();

count=rs.getInt(1);

}

if(count==0)

System.out.println("Invalid Credentials");

else

System.out.println("Valid Credentials");

sc.close();

} catch (ClassNotFoundException e) {

e.printStackTrace();

}

catch (SQLException e) {

e.printStackTrace();

}

}

}

Example PreparedStatement Solution

**package** com.nt.jdbc;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.util.Scanner;

**public** **class** LoginAppTestSolution {

**public** **static** **void** main(String[] args) {

Connection con=**null**;

PreparedStatement ps=**null**;

String query=**null**;

ResultSet rs=**null**;

**int** count=0;

Scanner sc=**null**;

**try** {

//read Inputs

sc=**new** Scanner(System.***in***);

System.***out***.println("Enter UserName: ");

String user=sc.nextLine();

System.***out***.println("Enter Password: ");

String pass=sc.nextLine();

//Convert input values as required for the SQl Queries

user="'"+user+"'";

pass="'"+pass+"'";

//1.Register type-4 jdbc driver software

Class.*forName*("oracle.jdbc.driver.OracleDriver");

//2.Establish the connection

con=DriverManager.*getConnection*("jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger");

//2.prepare sql Query

query= "SELECT COUNT(\*) FROM LOGIN WHERE USERNAME=? AND PASSWORD=?";

//3.create statement object

ps=con.prepareStatement(query);

//5. set query param values

ps.setString(1,user);

ps.setString(2,pass);

//6.send and execute SQl query in Database software

rs=ps.executeQuery();

//7.process the ResultSet Object

**if**(rs!=**null**) {

rs.next();

count=rs.getInt(1);

}

**if**(count==0)

System.***out***.println("Invalid Credentials");

**else**

System.***out***.println("Valid Credentials");

sc.close();

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

}

**catch** (SQLException e) {

e.printStackTrace();

}

}

}