2012-10-21

**Group 25’s problems regarding fetching data from Web server database and structure on Server.**

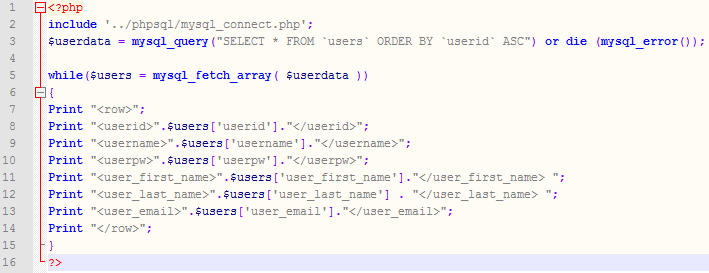
We have for some time had the web server up and running and been able to add data and retrieve data through PHP with MySQL queries.

We have decided that the server application (coded in Java) should get actions from the Android clients and then invoke methods required to do the actions and if information from the database (on the Web Server) is required, then go through the db\_handler.

The db\_handler should be able to access data from the database (on the Web Server).

Since we had no clue how to do it, if we should use Java and SQL or use HttpURLConnection through PHP with MySQL, as we used it already on the web server to test the database, we read about it and the HttpURLConnection seemed the easiest way to solve it, based on source 1 (listed in the end of this document).

After testing in a new separate project, writing the class for retrieving data, we successfully got a very long string consisting of whatever format we used to structure the data from the `users` table. Since we had output in HTML the first idea was to change all HTML tags into what we could use to parse the information in a good way. The following PHP with MySQL was used.

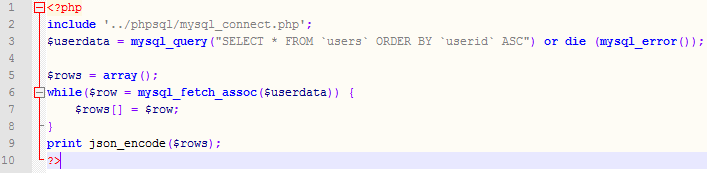


The output from the query to the Java application was:

<row><userid>4</userid><username>lillen</username><userpw>7896</userpw><user\_first\_name>Lillen</user\_first\_name> <user\_last\_name>Eklund</user\_last\_name> <user\_email>lillen@gmail.com</user\_email></row><row><userid>5</userid><username>musse</username><userpw>7865</userpw><user\_first\_name>Musse</user\_first\_name> <user\_last\_name>Pigg</user\_last\_name> <user\_email>musse@gmail.com</user\_email></row><row><userid>6</userid><username>långben</username><userpw>gillar\_majskolv</userpw><user\_first\_name>Jan</user\_first\_name> <user\_last\_name>Långben</user\_last\_name> <user\_email>långben@gmail.com</user\_email></row><row><userid>7</userid><username>nisse</username><userpw>nisse\_på\_Manpow</userpw><user\_first\_name>Nisse</user\_first\_name> <user\_last\_name>Manpower</user\_last\_name> <user\_email>nisse@manpower.se</user\_email></row><row><userid>8</userid><username>test2</username><userpw>bla bla</userpw><user\_first\_name>Test</user\_first\_name> <user\_last\_name>Testor</user\_last\_name> <user\_email>test@not4u.se</user\_email></row><row><userid>9</userid><username>fan\_vad\_jag\_är\_bra</username><userpw>eller\_hur?!</userpw><user\_first\_name>Ja det är jag!</user\_first\_name> <user\_last\_name>Jamen visst!</user\_last\_name> <user\_email>joråså@att.nu</user\_email></row><row><userid>10</userid><username>gösta</username><userpw>säkerhet</userpw><user\_first\_name>åäöstra</user\_first\_name> <user\_last\_name>glödhätaåsa</user\_last\_name> <user\_email>gödsel@bondgård.nu</user\_email></row><row><userid>11</userid><username>testigen</username><userpw>1234172487</userpw><user\_first\_name>testigen</user\_first\_name> <user\_last\_name>testigen\_lastname</user\_last\_name> <user\_email>testigen@föratt.nu</user\_email></row><row><userid>15</userid><username>malin</username><userpw>malle</userpw><user\_first\_name>malin</user\_first\_name> <user\_last\_name>malinder</user\_last\_name> <user\_email>malin@malinder.se</user\_email></row><row><userid>16</userid><username>anton</username><userpw>hemligt</userpw><user\_first\_name>anton</user\_first\_name> <user\_last\_name>anton</user\_last\_name> <user\_email>anton.anton</user\_email></row><row><userid>17</userid><username>Ser till och med</username><userpw>Bättre ut</userpw><user\_first\_name>På mobilen</user\_first\_name> <user\_last\_name>Än på datorn.</user\_last\_name> <user\_email>Dags@sova.nu</user\_email></row><row><userid>18</userid><username>Suspekt</username><userpw>bajsa1337</userpw><user\_first\_name>Harry</user\_first\_name> <user\_last\_name>Skaladuskitai</user\_last\_name> <user\_email>SuspektZ@gmail.com</user\_email></row><row><userid>19</userid><username>databaseExpert</username><userpw>isene</userpw><user\_first\_name>mattias</user\_first\_name> <user\_last\_name>isene</user\_last\_name> <user\_email></user\_email></row><row><userid>20</userid><username>sdfg</username><userpw>sdfg</userpw><user\_first\_name>sdfg</user\_first\_name> <user\_last\_name>sdfg</user\_last\_name> <user\_email>sdfg</user\_email></row><row><userid>22</userid><username>kkk</username><userpw>kkk</userpw><user\_first\_name>kkk</user\_first\_name> <user\_last\_name>kkk</user\_last\_name> <user\_email>kkk</user\_email></row>

Since we had problems writing a parser in Java to interpret the result (within our time limit), we started to look up what the alternatives were. One alternative is using JSON encoder and decoder, based on source 2. (See links at the end of this document)

We made a new JSON version of the PHP/MySQL query which looks like this:



This formats the output like this:

[{"userid":"4","username":"lillen","userpw":"7896","user\_first\_name":"Lillen","user\_last\_name":"Eklund","user\_email":"lillen@gmail.com"},{"userid":"5","username":"musse","userpw":"7865","user\_first\_name":"Musse","user\_last\_name":"Pigg","user\_email":"musse@gmail.com"},{"userid":"6","username":null,"userpw":"gillar\_majskolv","user\_first\_name":"Jan","user\_last\_name":null,"user\_email":null},{"userid":"7","username":"nisse","userpw":null,"user\_first\_name":"Nisse","user\_last\_name":"Manpower","user\_email":"nisse@manpower.se"},{"userid":"8","username":"test2","userpw":"bla bla","user\_first\_name":"Test","user\_last\_name":"Testor","user\_email":"test@not4u.se"},{"userid":"9","username":null,"userpw":"eller\_hur?!","user\_first\_name":null,"user\_last\_name":"Jamen visst!","user\_email":null},{"userid":"10","username":null,"userpw":null,"user\_first\_name":null,"user\_last\_name":null,"user\_email":null},{"userid":"11","username":"testigen","userpw":"1234172487","user\_first\_name":"testigen","user\_last\_name":"testigen\_lastname","user\_email":null},{"userid":"15","username":"malin","userpw":"malle","user\_first\_name":"malin","user\_last\_name":"malinder","user\_email":"malin@malinder.se"},{"userid":"16","username":"anton","userpw":"hemligt","user\_first\_name":"anton","user\_last\_name":"anton","user\_email":"anton.anton"},{"userid":"17","username":"Ser till och med","userpw":null,"user\_first\_name":null,"user\_last\_name":null,"user\_email":"Dags@sova.nu"},{"userid":"18","username":"Suspekt","userpw":"bajsa1337","user\_first\_name":"Harry","user\_last\_name":"Skaladuskitai","user\_email":"SuspektZ@gmail.com"},{"userid":"19","username":"databaseExpert","userpw":"isene","user\_first\_name":"mattias","user\_last\_name":"isene","user\_email":""},{"userid":"20","username":"sdfg","userpw":"sdfg","user\_first\_name":"sdfg","user\_last\_name":"sdfg","user\_email":"sdfg"},{"userid":"22","username":"kkk","userpw":"kkk","user\_first\_name":"kkk","user\_last\_name":"kkk","user\_email":"kkk"}]

That is, as an array with each user as an element containing all column names and the user specific values and using colons and commas for separation; colon according to format [column name]:[user value] and commas to separate columns.

There should be a decoder for Java, which we have included in the Server project to be able to decode the data returned in JSON.

Learnings:

* …

Further work:

* …

Future possible issues:

* …

Future possible improvements:

* …

Sources

1. <http://stackoverflow.com/questions/12472449/get-a-http-request-to-a-java-application-whenever-the-mysql-database-changes>
2. <http://stackoverflow.com/questions/383631/json-encode-mysql-results>
3. <http://stackoverflow.com/questions/2793150/how-to-use-java-net-urlconnection-to-fire-and-handle-http-requests>