Assignment 4 – INLS 623

Build Remote Database. Build a remote database connection using PHP to list out the contents of tables and schemas from your MySQL database from Assignment 1. You can base your work on the example we cover in class, i.e. see files

list-dbs.php
list-records.php
list-tables.php

in our "php1" directory (link to this directory is in our class schedule for this module), or you can write your own. You are responsible for making your own copy of PHP code that is runnable from the web. The easiest way to do this is using the class example is to copy all the files from this directory to your own web directory on ruby.ils.unc.edu (you could also use isis.unc.edu, or anywhere you have webpages). On ruby.ils.unc.edu issue the following commands:

```
cd
cd public_html
chmod 711 .
cp -r /fs1/htdocs/courses/2014_fall/inls623_001/php1  .
```

the first command makes sure you are in your home directory, the second one changes to your "public_html" directory (where your webpages are served from on ruby). The third command (chmod) may be necessary if you when you try to access the URL is gives you a message like

You don't have permission to access the requested directory. There is either no index document or the directory is read-protected. You don't have permission to access the requested object. It is either read-protected or not readable by the server.

For more information see

http://sils.unc.edu/it-services/servers/creating-webpages

The fourth command copies the folder (-r is recursive) php1 from the fall 2014 course folder to your own webspace (make sure you have "." at end of command).

You can then access your copy of the PHP files by pointing your browser to

http://ils.unc.edu/~ONYEN/php1/list-dbs.php

where ONYEN is your ONYEN.

If you use our class PHP example you must customize it at least a little bit (say to change fonts, colors—see the file *list-records.php*). Additionally for it to work you will need to change the connection to use your database "db2_X" where X is your own database, and your mysql password (right now it's set to use my personal test database ID, db2_29). So in each location it provides the database and password, i.e.

```
$conn = mysql_connect("pearl.ils.unc.edu","db1_29","db1_29");
```

You should change this to yours (i.e. if you were db2_18 and your password was X12345):

```
$conn = mysql_connect("pearl.ils.unc.edu","db2_18","X12345");
```

Just search and change all occurrences in each of the *list-*.php* files. To turn in this portion of the assignment you should produce a printout of the webpage showing the result form your version of the PHP code, listing your tables and schema from your db2_X database, and a listing your PHP source code (only the files you modified, and please highlight the parts you changed). Doing more than one or two simple changes to the class example, or doing your own different example, will result in top credit for this part of the assignment. See the other files (hello, example*) for further examples of PHP code.

Ramakrishnan Chapter 6, Exercises 6.1 (Page 217)

Your answers should not be one sentence in length. They do not have to be overly long, but they should be detailed and precise. Answer the following questions:

1. Explain the following terms: Cursor, embedded SQL, JDBC, SQLJ.
2. What are the differences between JDBC and SQLJ? Why do they both exist?
3. Explain the term *stored procedure*, and give examples why stored procedures are useful.

Ramakrishnan review questions (not numbered) that refer to the following mentioned sections (Page 264)

Answer the following questions:

- What are URIs and URLs? **(section 7.2.1)**
- What are some shortcomings of HTML and how does XML address them? **(section 7.4)**
- Why do we have XML DTDs? What is a well formed XML document? What is a valid XML document? Give an example (not a description) of an XML document that is (a) well formed but not valid and (b) and an XML document that is valid but not well formed **(section 7.4.2)**
- What is a three-tier architecture? What advantages does it offer over single-tier and two-tier architectures? Give a short overview of the functionality at each of the three tiers. **(section 7.5)**
- What problem do style sheets address? What are the advantages of using style sheets? **(section 7.6.3)**
- Why do we need to maintain state at the middle tier? What are cookies? How does a browser handle cookies? How can we access the data in cookies from servlets? **(section 7.7.5)**