Today we will discuss Python database access.

These are the facts:

1. Every Python program we wrote starts with the line below, also needs to be made executable:

   ```
   #!/usr/bin/python
   ```

2. The routines that allow your Python program to access databases are in this module:

   ```
   import MySQLdb
   ```

3. To connect to a database one needs to invoke the right function, and pass the right params:

   ```python
   Con = MySQLdb.Connect(host="silo.cs.indiana.edu", \
   port=57862, user="dgerman", \
   passwd="sp00n", db="demoOne")
   ```

   What you see above is a connection to a fictitious MySQL server on silo. Adapt accordingly. The thing you get back is an object, we think of it as being “the connection.”

4. A connection can give you a cursor, with which you can issue SQL statements. To obtain it:

   ```python
   Cursor = Con.cursor()
   ```

5. To issue an SQL statement we first write it down. We need it as a string:

   ```python
   sql = "SELECT * FROM judges"
   ```

   Question: I see we’re working with a table called judges. What database are we in?

6. With that string we can invoke a special cursor method that makes things happen:

   ```python
   Cursor.execute(sql)
   ```

   This silently sends the query and collects the answer (from the database server) inside.
7. You can invoke a second magical method of the cursor to extract those results:

```python
Results = Cursor.fetchall()
```

8. The easiest thing to do at this point would be to just print it, see what you got.

```python
print Results
```

9. It’s always thoughtful to close the connection to the database (like hanging up):

```python
Con.close()
```

Now this is what we need to do:

- let’s get in our database server as root and create a new database

- let’s give to our already existing regular user access to it

- let’s get out as root and come back as the regular user

- let’s select the database, create a table, put some data in it

- let’s get out, then go to cgi-bin and create a Python script that extracts and prints the data from the table we just created
Reading assignment for next time: what does the following program do? How does it do it? Do you see any of the above in it?

#!/usr/bin/python
import cgi, random, MySQLdb, sys, os

(message, session_id, balance, action, id) = ("", ",", ",", ",")
print "Content-type: text/html\n\n"

q = cgi.FieldStorage()
if q.has_key("action") : action = q["action"].value

Con = MySQLdb.Connect(host="silo.cs.indiana.edu", port=8974, user="lbird", passwd="dribl", db="awards")
Cursor = Con.cursor()

if q.has_key("session_id") and action != "Reset":
    id = q["session_id"].value
    Cursor.execute("SELECT message, balance FROM example WHERE session_id = '%s'" % q["session_id"].value)
    Results = Cursor.fetchall()
    (message, balance) = Results[0]
    balance = str(int(balance) + 1)
    message = "You have clicked me %s times. " % balance
    Cursor.execute("update example set message = '%s', balance = '%s' where session_id = '%s'" %
        (message, balance, q["session_id"].value))
else:
    balance = 0
    message = "Welcome."
    for i in range(8): id += str(random.randrange(10))
    if not Cursor.execute("insert into example (session_id, balance, message) values ('%s', '%s', '%s')" %
        (id, balance, message)):
        print "Content-type: text/html\n\nError. Please reload."
sys.exit()

print ""
<form>
%s <input type=submit name=action value=Proceed> <input type=submit name=action value=Reset>
<input type=hidden name=session_id value=%s>
</form>
"" % (message, id)