

mySQL

MySQL in Flask

in 15 or so slides

but first ...

Principle of Least Privilege

A user (or process) should have the lowest level of privilege required in order to perform his/her assigned task.

Principle of Least Privilege

- ❖ not root
 - ❖ why? because as root we can do anything

```
CREATE DATABASE IF NOT EXISTS aliendb;
```

```
GRANT ALL PRIVILEGES ON aliendb.* to  
'alienUsername'@'localhost' identified by 'alienPassword';
```

Root have Password?

❖ if not ...

```
SET PASSWORD FOR 'root'@'localhost' = PASSWORD('f3jga908');
```

Back to MySQL & Flask

Follow along in the simpleSample folder under alienAbduction!!

4 Basic Steps

1. connect with `MySQLdb.connect`
2. assemble query string
3. execute the query with `cur.execute`
4. fetch the results

1) connect with MySQLdb.connect

```
# utils.py
import MySQLdb

DATABASE='microBlogDB'
DB_USER = 'blogUser'
DB_PASSWORD = 'blogPassword'
HOST = 'localhost'

def db_connect():
    return MySQLdb.connect(HOST, DB_USER, DB_PASSWORD, DATABASE)
```

1) connect with MySQLdb.connect

```
db = utils.db_connect()
```

```
cur = db.cursor()
```

In computer science, a database cursor is a control structure that enables traversal over the records in a database. Cursors facilitate subsequent processing in conjunction with the traversal, such as retrieval, addition and removal of database records. ...

http://en.wikipedia.org/wiki/Database_cursor

1) connect with MySQLdb.connect

```
db = utils.db_connect()
```

```
cur = db.cursor()
```

In computer science, a database cursor is a control structure that enables traversal over the records in a database. Cursors facilitate subsequent processing in conjunction with the traversal, such as retrieval, addition and removal of database records. ...

http://en.wikipedia.org/wiki/Database_cursor

It is a structure we will use to interact with the database

2) assemble query string

```
query = "SELECT population FROM Country WHERE name = 'Haiti'"
```

```
query = "SELECT * FROM users WHERE name = '" +  
        request.form['name'] + "'"
```

```
print query
```

3) execute the query with `cur.execute`

```
cur.execute(query)
```

4) fetch the results

```
rows = cur.fetchall()
```

```
return render_template('index.html', posts=rows)
```

```
{% for post in posts %}
```

```
<p>{{post[1]}} <b>{{post[0]}}</b></p>
```

```
{% endfor %}
```

simpleSample

All this in the demo

utils.py
server.py
index.html

The Task

- ❖ MySQL-ify alien abductions
- ❖ partner task