

## CS1116/CS5018

### Web Development 2

Dr Derek Bridge

School of Computer Science & Information Technology  
University College Cork

### What we will do

- We will create a database where we can store the comments
  - We will replace the Web page (page .html) by a server-side Python program (page .py)
    - It will be a self-processing page
    - Its form will use the HTTP post method, not the get method
- Q: Why?

### Case study 1A: a Web page comments facility

- Suppose you have a Web page, page .html
- Suppose you want to allow visitors to the page to
  - post comments
  - view the comments that have been posted

### MySQL database for the Web page comments facility

```
CREATE TABLE comments_table
(
  comment_id INT AUTO_INCREMENT,
  username VARCHAR(255),
  comment TEXT,
  PRIMARY KEY (comment_id)
);
```

## The Web page/Python program, page .py

```
#!/usr/local/bin/python3
from cgi import enable
enable()
from cgi import FieldStorage
import pymysql as db
print('Content-Type: text/html')
print()
comments = ''
try:
    connection = db.connect('localhost', 'userid', 'password', 'database_name')
    cursor = connection.cursor(db.cursors.DictCursor)
    form_data = FieldStorage()
    if len(form_data) != 0:
        username = form_data.getfirst('username')
        new_comment = form_data.getfirst('new_comment')
        cursor.execute("""INSERT INTO comments_table (username, comment)
            VALUES (%s, %s)""", (username, new_comment))
        connection.commit()
    cursor.execute("""SELECT * FROM comments_table
        ORDER BY comment_id DESC""")
    for row in cursor.fetchall():
        comments += '<article><h1><p>%s</p></articles>' % (row['username'], row['comment'])
    cursor.close()
    connection.close()
except db.Error:
    comments = '<p>Sorry! We are experiencing problems at the moment. Please call back later.</p>'
print("""
<doctype html>
<html lang="en">
<head>
<meta charset="utf-8" />
<title>My Web Page</title>
</head>
<body>
<p>
Hello! This is my Web page!
</p>
</section>
<h1>Comments</h1>
<form action="page.py" method="post">
<fieldset>
<legend>Post a new comment</legend>
<label for="username">Name:</label>
<input type="text" name="username" id="username" />
<label for="new_comment">Comment:</label>
<textarea name="new_comment" id="new_comment" rows="5" cols="50">
</textarea>
<input type="submit" />
</fieldset>
</form>
</section>
</body>
</html>""") % (comments))
```

## Case study 1B: a Web site comments facility

- Suppose you have a Web site with lots of pages (e.g. page1.html, page2.html, etc.)
- Suppose you want to allow visitors to
  - post comments on more than one page in a Web site
  - view the comments that have been posted

## What we will do

- We will create a slightly different database
- We will write a separate Python module, comment.s.py
- We will replace every Web page (page1.html, page2.html) by server-side Python programs (page1.py, page2.py, etc.)
- We will use import to load in our new module

## MySQL database for the Web site comments facility

```
CREATE TABLE comments_table
(
    comment_id INT AUTO_INCREMENT,
    username VARCHAR(255),
    url VARCHAR(255) NOT NULL,
    comment TEXT,
    PRIMARY KEY (comment_id)
);
```

## The new module, comments .py

```
#!/usr/local/bin/python3
from cgi import enable
enable()

def get_comments():
    from cgi import FieldStorage
    import pymysql as db
    from os import environ

    comments = ''
    url = environ.get('SCRIPT_NAME')
    try:
        connection = db.connect('localhost', 'userid', 'password', 'database_name')
        cursor = connection.cursor(db.cursors.DictCursor)
        form_data = FieldStorage()
        if len(form_data) != 0:
            username = form_data.getfirst('username')
            new_comment = form_data.getfirst('new comment')
            cursor.execute("""INSERT INTO comments_table (username, url, comment)
                VALUES (%s, %s, %s)""", (username, url, new_comment))
            connection.commit()
        cursor.execute("""SELECT * FROM comments_table
            WHERE url = %s
            ORDER BY comment_id DESC""", (url))
        for row in cursor.fetchall():
            comments += '<article><h1>%s</h1><p>%s</p></article>' % (row['username'], row['comment'])
        cursor.close()
        connection.close()
    except db.Error:
        comments = '<p>Sorry! We are experiencing problems at the moment. Please call back later.</p>'
    return ""

<section>
<h1>Comments</h1>
<form action="%s" method="post">
    <fieldset>
        <legend>Post a new comment</legend>
        <label for="username">Name:</label>
        <input type="text" name="username" id="username" />
        <label for="new_comment">Comment:</label>
        <textarea name="new_comment" id="new_comment" rows="5" cols="50">
        </textarea>
        <input type="submit" />
    </fieldset>
</form>
%s""" % (url, comments)
```

## A typical Web page/Python program, page1 .py

```
#!/usr/local/bin/python3
from cgi import enable
enable()

from comments import get_comments
print('Content-Type: text/html')
print()

print("""
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8" />
<title>One Of My Web Pages</title>
</head>
<body>
<p>
    Hello! This is one of my Web pages!
</p>
</body>
</html>""") % (get_comments())
```