LOYOLA UNIVERSITY CHICAGO

CENTER FOR TEXTUAL STUDIES AND DIGITAL HUMANITIES

DIGH 402 - Introduction to Digital Humanities Design and Programming

Spring Semester 2014

Week 3

MySQL Database

- create a database called ‘402framework’

- add a table called ‘users’ using the outlined properties

- add some data to this new table

Databases - PHPMYADMIN

Create database and table using PHPMYADMIN - USERS

	Type	Collation	Attributes	Null	Default	Extra	Primary
userid	int(10)		unsigned	No	None	auto_increment	Yes
username	varchar(30)	utf8_unicode_ci		No	None		
firstname	varchar(30)	“ ”		No	None		
lastname	varchar(50)	“ “		No	None		
usercreated	timestamp	“ “		No	CURRENT_TIMESTAMP		

Databases - PHPMYADMIN

Example SELECT Queries

- SELECT * FROM users;
- SELECT username, usercreated FROM users;
- SELECT * FROM users WHERE userid=3;
- SELECT username, usercreated FROM users WHERE userid=3;
- SELECT userid FROM users WHERE username=”yvaine”;
- SELECT DISTINCT username FROM users;
- SELECT userid FROM users WHERE firstname=”yvaine” AND lastname=”wall”;
- SELECT userid FROM users WHERE firstname=”yvaine” OR lastname=”wall”;
- SELECT * FROM users WHERE username LIKE 'yvaine%';
- SELECT * FROM users WHERE lastname IN ('issit', 'stormhold');
- SELECT * FROM users WHERE userid BETWEEN 2 AND 5;
    - SELECT * FROM users WHERE userid NOT BETWEEN 2 AND 5;
    - SELECT * FROM users WHERE lastname BETWEEN 'd' and 'm';
    - SELECT * FROM users WHERE lastname BETWEEN 'd' and 'm' ORDER BY firstname;

...

PHP and MySQL

Expand 402framework database

- add new tables for
    - content, content_type, content_lookup

- consider each table as separate data except ‘content_lookup’

- consider required table properties for ‘content’ and ‘content_type’
    - data type, collation, attributes, null, default, and any extras and indices

- content_lookup properties
    - artificially replicating foreign keys in MyISAM
    - primary keys
    - combine ‘content’, ‘content_type’, and ‘users’

- why an artificial ‘lookup’ table?
    - speed
    - easier to read and reference

- alternatives such as
    - embed reference ID in ‘content’ table

    http://sqlfiddle.com/#!2/58892/20

Databases - PHPMYADMIN

Create database and table using PHPMYADMIN - CONTENT

	Type	Collation	Attributes	Null	Default	Extra	Primary
contentid	int(10)		unsigned	No	None	auto_increment	Yes
contentname	varchar(150)	utf8_unicode_ci		No	None		
contentdesc	text	“ ”		No	None		
contenttext	text	“ “		No	None	fulltext	
contentcreated	timestamp	“ “		No	CURRENT_TIMESTAMP		

Databases - PHPMYADMIN

Create database and table using PHPMYADMIN - CONTENT_TYPE

	Type	Collation	Attributes	Null	Default	Extra	Primary
content_type_id	int(10)		unsigned	No	None	auto_increment	Yes
content_type_name	varchar(150)	utf8_unicode_ci		No	None		
content_type_desc	text	“ ”		No	None		

Databases - PHPMYADMIN

Create database and table using PHPMYADMIN - CONTENT_TYPE_LOOKUP

	Type	Collation	Attributes	Null	Default	Extra	Primary
content_id	int(10)		unsigned	No	None		Yes
content_type_id	int(10)		unsigned	No	None		Yes
user_id	int(10)		unsigned	No	None		Yes

Databases - PHPMYADMIN

Using our new tables and DB structure

- manage, list, and provide work records for our users, such as
    - who is a user of the framework
    - content created and when
    - content types created per user (and again when)

- potential patterns?

- use the ‘content_type_lookup’ table to ask the following queries
    - current content by content_type
    - current content by user/users
    - current content by timestamp …

- how would this change if we added the following columns to our ‘users’ table
    - gender
    - age
    - country

and so on...

Databases - PHPMYADMIN

Selecting data from our new tables - queries against ‘content_type_lookup’

- find total number of content items of a given content type

```
SELECT COUNT(content_lookup.content_type_id) FROM content_lookup WHERE content_type_id=1;
```

- find all content with content name, content id, and username by user id

```
SELECT content.contentid, content.contentname, users.username FROM content_lookup, content, users WHERE content_lookup.content_id=content.contentid AND users.userid=content_lookup.user_id AND content_lookup.user_id=1
```

- find all content with content type name and content name by content type id

```
SELECT content_type.content_type_name, content.contentname FROM content, content_type, content_lookup WHERE content_lookup.content_id=content.contentid AND content_lookup.content_type_id=content_type.content_type_id AND content_lookup.content_type_id=1;
```

and many more...

PHP and MySQL

SELECT FROM database and table - USERS

- SELECT * FROM users;

    PHP Example

- add user privileges to 402framework database
    - admin user & basic query user
    - different privileges relative to user requirements and framework security

PHP and MySQL

Basic PHP MySQL connection - admin or basic user privileges

```
<?php
$con = mysqli_connect('localhost', 'admin402', 'digh402', '402framework');
if (!$con) {
    die('Could not connect: ' . mysql_error());
}

$result = mysqli_query($con,"SELECT * FROM users");

while($row = mysqli_fetch_array($result))
  {
  echo 'user = '.$row['username'] . " & created = " . $row['usercreated'];
  echo "<br>";
  }

mysql_close($con);
?>
```

PHP and MySQL

Test basic PHP MySQL connection & SELECT queries - against USERS table

- modify basic connection to query ‘users’ table
    - SELECT username, usercreated FROM users;
    - SELECT * FROM users WHERE userid=3;
    - SELECT DISTINCT username FROM users;

- output results in a tabular format

    PHP Example