

**USD WEB SERVICES
ADOBE DREAMWEAVER
DATABASE DEVELOPMENT**



INFORMATION TECHNOLOGY SERVICES
UNIVERSITY OF SAN DIEGO

DEVELOPED BY JOY BRUNETTI
BRUNETTI@SANDIEGO.EDU
x8772

OCTOBER 2006

TABLE OF CONTENTS
DREAMWEAVER DATABASE DEVELOPMENT

FIRST THINGS FIRST	2
USING DYNAMIC DATA	2
CONNECTING TO A DATABASE	3
CREATING A RECORDSET	4
DISPLAYING DATA	5
INSERTING, UPDATING, AND DELETING RECORDS	6
INSERT RECORD	6
UPDATE RECORD	7
DELETE RECORD	8
MAKING PAGES	9

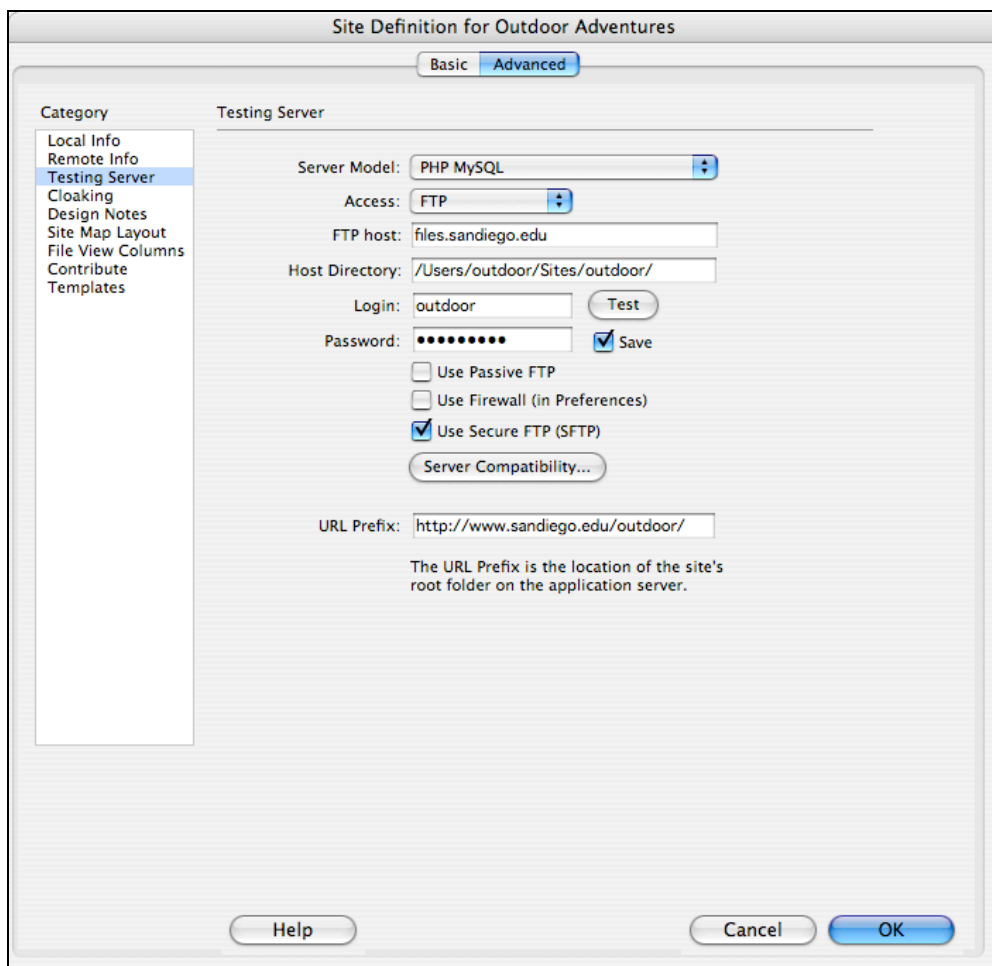
First Things First

Connect a database in Dreamweaver once you have created the corresponding database in MySQL using CocoaMySQL. Establish the actual MySQL database first since Dreamweaver is simply the mechanism for transferring data in/out of it via an easy and accessible Web interface.

Using Dynamic Data

To use dynamic data on a page, you will first need to tell Dreamweaver that you are working with PHP and MySQL on your site. In the Application panel in Dreamweaver, do the following:

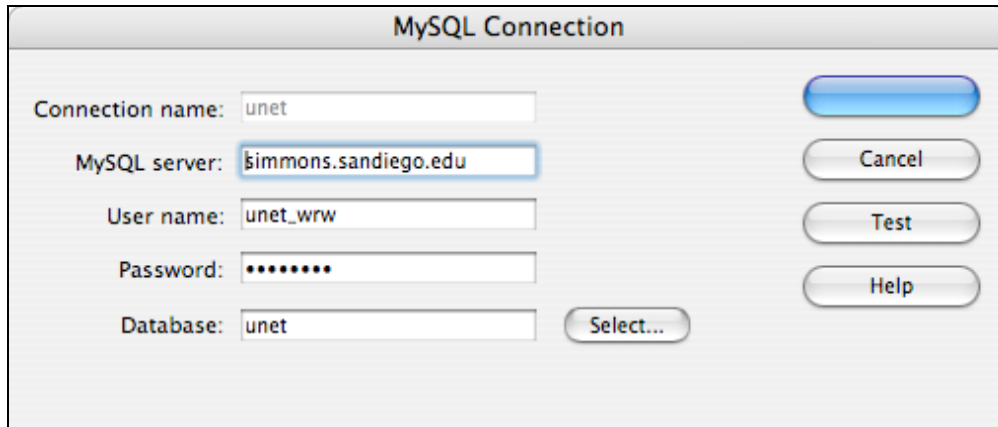
1. Create a site for this file: already done as long as the site is created in Dreamweaver
2. Choose a document type: open a Web page so that DW sees the *.php extension
3. Set up the site's testing server: click on the link to be taken to a window where you will need to select "PHP MySQL" as the server model. The rest of the information will be filled in once the selection is made. **You will need to change the URL prefix to be the URL of the Web site you are working on.**



4. Create a connection by clicking on the + button above. Connect to the actual database using the next steps.

Connecting to a Database

After you click on the + button in the Application panel a window will come up where you will need to fill in the database information:



1. Connection name: a brief one-word name to call the database connection (e.g. unet, computerinv)
2. MySQL server: simmons.sandiego.edu
3. User name: this will either be database_wr or database_wrw depending on if you need to just view information or if you will need to insert and modify records. For our purposes you will normally connect to wrw.
4. Password: check with Karen for database passwords
5. Database: use the select button to browse for the appropriate database. Again this is something like unet, computerinv, etc.

Click OK and the database connection will now be displayed in the Application panel. You can expand the entry to view tables within that database.

Creating a Recordset

The screenshot shows a dialog box titled "Recordset" with the following fields and controls:

- Name:** A text input field containing "rsPR".
- Connection:** A dropdown menu showing "unet" and a "Define..." button.
- Table:** A dropdown menu showing "epicpr".
- Columns:** Radio buttons for "All" (selected) and "Selected:". Below is a list box containing "id", "title", "date", "contact", and "description".
- Filter:** A dropdown menu showing "None", followed by an equals sign "=" and another dropdown menu.
- URL Parameter:** A dropdown menu showing "URL Parameter" and an empty text input field.
- Sort:** A dropdown menu showing "date" and another dropdown menu showing "Descending".

On the right side of the dialog, there are several buttons: "OK", "Cancel", "Test", "Advanced...", and "Help".

A recordset is a grouping of fields that you will use within a form or a display of data. To create a recordset you will need to be in the Bindings tab of the Application panel:

1. Name: give the recordset a name with the naming scheme of rsName
2. Connection: select the database connection currently on the site
3. Table: select the table you created in MySQL with the fields that you will be inserting on this Web page.
4. Columns: select the columns you will need in this recordset (normally "All")
5. Filter: normally none except on detail pages where you will need to create an "id" filter. In this case URL parameter will equal "id", otherwise leave it blank.
6. Sort: sort the information in whatever way makes sense, for example by a date in descending order.

Displaying Data

To display data from a database, simply drag the fields from your recordset onto the page in the layout that makes sense.

1. Go to the Bindings tab in the Application panel
2. Drag the fields over from the recordset

If you need for the data to repeat, do the following:

1. Go to the Server Behaviors tab in the Application panel
2. Select the fields on your Web page that you need repeated
3. Click on the + button and select Repeat Region
 - a. Recordset: select the recordset you are using on your page
 - b. Show: select how many records should repeat. Unless there are a high volume of entries in the database, normally you would just show all records.

Inserting, Updating, and Deleting Records

You will need to create a form on your Web page before using the insert, update (modify), or delete record behaviors.

Once you have your form ready, you can find Insert Record, Update Record, or Delete Record in the Server Behaviors tab of the Application panel and do the following:

Insert Record

The screenshot shows the 'Insert Record' dialog box with the following configuration:

- Submit values from: alerts
- Connection: unet
- Insert table: alerts
- Columns:
 - 'id' Is an Unused Primary Key.
 - 'title' Gets Value From 'FORM.title' as 'Text'
 - 'date' Gets Value From 'FORM.date' as 'Date'
 - 'contact' Gets Value From 'FORM.contact' as 'Text'
 - 'description' Gets Value From 'FORM.description' as 'Text'
 - 'email' Gets Value From 'FORM.email' as 'Text'
- Value: None
- Submit as: Text
- After inserting, go to: ./index.php

1. Submit values from: select the table where your data should be inserted
2. Connection: select the database you are working with
3. Insert table: select the table where the data should be inserted
4. Columns: match the fields in your form with the fields in the MySQL database. If you have named the fields the same in both places, then DW will automatically match them up for you.
5. Value: if you do need to match up the columns manually, do so by selecting the appropriate corresponding item here for each item in your form listed above it
6. Submit as: select if something is text, integer, etc. Normally this is already filled in for you.
7. After inserting, go to: select the page where you should be taken after clicking "submit" on your form on the Web

Update Record

Update Record

Submit values from: alerts

Connection: unet

Update table: alerts

Columns:

- 'id' Is an Unused Primary Key.
- 'title' Gets Value From 'FORM.title' as 'Text'
- 'date' Gets Value From 'FORM.date' as 'Date'
- 'contact' Gets Value From 'FORM.contact' as 'Text'
- 'description' Gets Value From 'FORM.description' as 'Text'
- 'email' Gets Value From 'FORM.email' as 'Text'

Value: None

Submit as: Text Primary key

After updating, go to: ../index.php Browse...

1. Submit values from: select the table where your data should be inserted
2. Connection: select the database you are working with
3. Update table: select the table where the data should be updated
4. Columns: match the fields in your form with the fields in the MySQL database. If you have named the fields the same in both places, then DW will automatically match them up for you.
5. Value: if you do need to match up the columns manually, do so by selecting the appropriate corresponding item here for each item in your form listed above it
6. Submit as: select if something is text, integer, etc. Normally this is already filled in for you.
7. After inserting, go to: select the page where you should be taken after clicking "submit" on your form on the Web
8. Click OK
9. In your form, go to each field and select an initial value by clicking on the field and then clicking on the lightning bolt in "init val" down in the Properties panel. This is necessary so that the form displays the information currently in a record before being updated.

Delete Record

Delete Record

First check if variable is defined: Primary key value

Connection: unet

Table: epicevent

Primary key column: id Numeric

Primary key value: URL Parameter eventdid

After deleting, go to: Browse...

OK
Cancel
Help

1. First check if variable is defined: set to Primary key value
2. Connection: select the database you are working on
3. Table: select the table from which you would like records deleted
4. Primary key column: should be "id" and Numeric should be checked
5. Primary key value: select URL Parameter and then give the key a one-word name such as eventid, computerid, or alerted
6. After deleting, go to: leave this blank

Making Pages

As you've learned you'll need a few pages for each step of the input process. Some general rules of thumb:

- Normally you'll want to create an add.php page where you will have a form inserting a record.
- Then you'll want to create listing.php which simply lists (repeats) the entries being submitted into the database. You should display information such as titles, dates, and names if possible.
- Lastly, detail.php will be your display page where each record is detailed in its entirety. This is basically a copy of your add page, but with the fields displaying instead of inserting.
- You might also need a page called modify.php which is a copy of add.php, but with the update record behavior rather than the insert record behavior.
- You'll want to create a file called manage.php with links to each of these pages to give the user an easy interface for adding, modifying, and deleting pages.

For an example, look at how our Key database works at <http://www.sandiego.edu/its/keys>. There are examples of all of the above in an accessible and easy to use format. You will need to log in to access the site.

Happy database work ahead!