

topics:

- MySQL and PHP
- on-line documentation
 - <http://www.php.net/manual/en/ref.mysql.php>

using mysql and php

- basically, there are four things you want to be able to do from within php:
 - connect to the mysql database
 - disconnect from the mysql database
 - execute mysql queries
 - check the status of your mysql commands
- queries can be any kind of mysql query, including SELECT, UPDATE, INSERT, etc.
- following SELECT queries, you can execute mysql/php functions to put the data read from the mysql database into your php variables

connecting to the mysql database

- here is an example of connecting to the database:

```
$conn = mysql_connect( $mysql_host, $mysql_user, $mysql_password )  
    or die( 'Could not connect: ' . mysql_error() );  
echo 'Connected successfully';  
mysql_select_db( $mysql_db ) or die( 'Could not select database' );
```
- replace the variables \$mysql_host, \$mysql_user, \$mysql_password and \$mysql_db with strings containing the values for your database
- notice that there are two functions invoked
 - one that logs into mysql: mysql_connect()
 - one that selects the database to use: mysql_select_db()

disconnecting from the mysql database

- to disconnect from mysql, there is one function needed:

```
mysql_close( $conn );
```

check the status of your mysql commands

- if errors occur, the functions return errors
- these errors can be read as strings using the function `mysql_error()`
- note the usage in this statement:

```
$conn = mysql_connect( $mysql_host, $mysql_user, $mysql_password )  
    or die( 'Could not connect: ' . mysql_error() );  
echo 'Connected successfully';
```

execute mysql queries

- here is an example of executing a SELECT query:

```
// set up and execute the MySQL query  
$query = 'SELECT * FROM my_table';  
$result = mysql_query( $query )  
    or die( 'Query failed: ' . mysql_error() );  
// print the results as an HTML table  
echo "<table>\n";  
while ( $row = mysql_fetch_array( $result, MYSQL_ASSOC ) ) {  
    echo "\t<tr>\n";  
    foreach ( $row as $item ) {  
        echo "\t\t<td>$item</td>\n";  
    }  
    echo "\t</tr>\n";  
}  
echo "</table>\n";  
// free result  
mysql_free_result( $result );
```

- there are three functions used here
 - one to execute the query and store the result in a local variable: `mysql_query()`
 - one to parse the data read returned from the query as an array: `mysql_fetch_array()`
 - one to free the memory used by the query result: `mysql_free_result()`
- NOTE that if the result returned is a scalar and not an array, then only `mysql_query()` needs to be called and does not need to be followed by a call to `mysql_fetch_array()`
- finally, note the use of `mysql_error()` in the query function

other handy functions

- there is a long list of php/mysql functions
- the whole list is available on the web page listed on the first page of these notes
- here are a few of the more handy functions:
 - `int mysql_num_rows ($result)`
this function returns the number of rows contained in a `$result`; this is relevant when the result returned from `mysql_query()` is an array and not a scalar
 - `int mysql_affected_rows ($conn)`
this function returns the number of rows that were affected by the most recently executed INSERT, UPDATE, REPLACE or DELETE query; this is useful for checking if what you expected to happen with these commands actually happened (e.g., if the row you expected to insert was inserted, etc.)

- `int mysql_insert_id ($conn)`
this function is used when inserting a row into a table that has an `AUTO_INCREMENT` ID field; the function returns the ID number that was generated
- `string mysql_stat ($conn)`
this function returns a string containing information about the status of the current database connection