



PHP – part 4

By Ned Lowe

ned.lowe@ic.ac.uk

<http://www.iesoc.com/>

Text Manipulation

- `$new = str_replace($search, $replace, $original);`
 - Searches for substring `$search` in `$original`
 - Replaces all instances with `$replace`
 - Returns new string into `$new`
- Cleaning up user input
 - `$var = trim($var);`
 - Removes whitespace from beginning and end of `$var`
- Substrings
 - `$newstring = substr($originalstring,0,3);`
 - starts from position '0' and takes '3' characters

```
<html>
  <head>
    <title>Create Page</title>
  </head>
  <body>
    <form action='temp2.php' method='post'>
      <textarea name='userinput'></textarea><br>
      <input type='submit'>
    </form>
  </body>
</html>
```

```
<?php
$userinput = trim($userinput);
$userinput = str_replace("joe", "anon", $userinput);
echo nl2br($userinput);
?>
```

SQL part 2

- Deleting a table
 - drop table *name*;
- Removing a column from a table
 - alter table *name* drop *column*;
- Adding a column to a table
 - alter table *name* add *column properties*;
 - alter table people add email varchar(255);
- Modifying an existing column
 - alter table *name* modify *column new_properties*;
 - alter table name modify name varchar(255);

Time and Dates

- `date(parameters)` function
 - prints the current date and time
 - a – Morning or Afternoon (am or pm)
 - A – Morning or Afternoon (AM or PM)
 - d – day of the month with leading zero (03, 31)
 - D – day of the week in 3 char abbreviated format (Mon, Sun)
 - F – month of the year (January, November)
 - g – Hour of the day in 12 hour format (1,10)
 - G – Hour of the day in 24 hour format (0,22)
 - h – Hour of the day in 12 hour format (01,10)
 - H – Hour of the day in 24 hour format (00,22)

- i – minutes past the hour (01,45)
- j – Day of the month (1, 26)
- l – Day of the week (Monday, Tuesday)
- L – leap year, 1 if leap year, 0 if not
- m – month of year (01,11)
- M – month of year in 3 char abbreviated format (Jul, Dec)
- n – month of year (1,11)
- s – seconds past the minute (00,52)
- S – Ordinal suffix (st, nd, th)
- t – total number of days in month
- U – Unix timestamp (number of seconds since 1st Jan 1970)
- w – day of the week as a single digit (Sunday = 0, Wednesday=3)
- y – Year in 2 digit format (00, 45)
- Y – Year in 4 digit format (2000, 2045)
- z – Day of the year as a number (0, 365)

Using date() function

- `echo date("jS F Y");`
- example 27th May 2003
- Any character that is not one of the above list can be used
- If a variable exists that stores a date then it can be passed to the function to print
- `echo date("jS F Y",$var);`

Creating times

- `$var = mktime(hour,minute,second,month,day,year);`
- `$var = mktime(5,34,00,10,23,02);`
- `echo time("js F Y",$var);`
- Unix Timestamps allow sophisticated date manipulation

```
<?php
$result = mysql_query("select * from calendar where
    entry_id='$calname'", $db);
$masterdatestring = mysql_result($result, 0, "entry_date");
$yearstring = substr($masterdatestring, 0, 3);
$monthstring = substr($masterdatestring, 3, 3);
$daystring = substr($masterdatestring, 6, 3);
$datestring = mktime("", "", "", $monthstring, $daystring, $yearstring);
echo date("jS F Y", $datestring);
?>
```

- Stored as a string for ease then converted back to a date
- If no time is required then just leave it blank “ ”

.htaccess files

- Only works for Apache web server
 - htpasswd is an Apache program to create passwords
- Create a file in the directory you wish to password protect, saved as .htaccess

```
AuthUserFile /usr/home/path/to/directory/.htpass
```

```
AuthName "Area Name – up to you"
```

```
AuthType Basic
```

```
require valid-user
```

.htaccess part 2

- Within the folder you wish to password protect type:
 - `htpasswd -c .htpass username`
- It will prompt you for the password – enter it
- That directory is now protected by password
- For multiple user simply run the above command many times

Passwords

- One way cryptographic hashing function
 - `crypt(stringtoencrypt, 'key');`
 - `crypt($password, 'iesoc');`
 - Returns a string that cannot be decrypted
- Don't compare against the string, compare against the encrypted string

```
if (crypt($password, 'iesoc')==“xsdgsksvd356”) {  
    echo “Correct password”;  
}
```

```
<form action='create.php' method='post'>
  <input type='text' name='username'><br>
  <input type='text' name='password'><br>
  <input type='submit'>
</form>
```

```
<?php
$password = crypt($password,'iesoc');
$db = mysql_pconnect("localhost","username","password");
mysql_select_db("tablename",$db);
mysql_query("insert into accounts(username,password)
  values('$username','$password');
?>
```

```
<?php
$db = mysql_pconnect("localhost","username","password");
mysql_select_db("tablename",$db);

$password = crypt($password,'iesoc');
$result = mysql_query("select * from people where
    username='$username' and password='$password'", $db);
if (mysql_num_rows($result) == 1) {
echo "You are logged in";
}
else {
echo "Please try again";
}
?>
```