

Раздаточный материал по предмету “Язык программирования PHP”. Лекция 8.

Работа с датой и временем

Пример: Установка голландской локали

```
setlocale(LC_ALL, 'nl_NL');
```

Некоторые ключи используемые для формирования даты:

<i>B</i>	Время в стандарте Swatch Internet	От 000 до 999
<i>s</i>	Дата в формате ISO 8601 (добавлено в PHP 5)	2004-02-12T15:19:21+00:00
<i>d</i>	День месяца, 2 цифры с ведущими нулями	от 01 до 31
<i>D</i>	Сокращенное наименование дня недели, 3 символа	от Mon до Sun
<i>F</i>	Полное наименование месяца, например January или March	от January до December
<i>L</i>	Признак високосного года	1, если год високосный, иначе 0.
<i>m</i>	Порядковый номер месяца с ведущими нулями	От 01 до 12
<i>U</i>	Количество секунд, прошедших с начала Эпохи Unix (The Unix Epoch, 1 января 1970, 00:00:00 GMT)	
<i>Y</i>	Порядковый номер года, 4 цифры	Примеры: 1999, 2003
<i>y</i>	Номер года, 2 цифры	Примеры: 99, 03
<i>z</i>	Порядковый номер дня в году (нумерация с 0)	От 0 до 365

Пример: Использование функции date()

```
<?php
// вывод дня недели, например Вторник
echo date("l");

// вывод даты в формате: Вторник 32-ое Января 2008 02:38:38
echo date("l d-oe F Y h:i:s A");
?>
```

Пример: Получение дат, отличных от текущей

```
<?php
$tomorrow = mktime(0, 0, 0, date("m"), date("d")+1, date("Y"));
$lastmonth = mktime(0, 0, 0, date("m")-1, date("d"), date("Y"));
$nextyear = mktime(0, 0, 0, date("m"), date("d"), date("Y")+1);
?>
```

- Функция `getdate()` вернёт информацию о текущих дате и времени

Имеет следующий прототип

```
array getdate ( [int $timestamp] )
```

Если задан необязательный аргумент, то массив будет сформирован для времени, хранящемся в нём.

Функция вернёт ассоциативный массив со следующими ключами:

"seconds"	Секунды	От 0 до 59
"minutes"	Минуты	От 0 до 59
"hours"	Часы	От 0 до 23
"mday"	Порядковый номер дня месяца	От 1 до 31
"wday"	Порядковый номер дня	От 0 (воскресенье) до 6 (суббота)
"mon"	Порядковый номер месяца	От 1 до 12
"year"	Порядковый номер года, 4 цифры	Примеры: 1999, 2003
"yday"	Порядковый номер дня в году (нумерация с 0)	От 0 до 365
"weekday"	Полное наименование дня недели	От Sunday до Saturday
"month"	Полное наименование месяца, например January или March	от January до December

- Функция `gettimeofday()` также вернёт ассоциативный массив, но только с ключами ответственными за представление времени.
- Функция `localtime()` возвращает массив, аналогичный по структуре одноимённой функции C. Имеет следующие поля:
 - "tm_sec" - секунды
 - "tm_min" - минуты
 - "tm_hour" - часы
 - "tm_mday" - день месяца
 - "tm_mon" - месяц года, 0 соответствует январю
 - "tm_year" - Количество лет, прошедших с 1900 г.
 - "tm_wday" - день недели
 - "tm_yday" - порядковый номер дня в году
 - "tm_isdst" - признак летнего времени

Кроме получения даты в нужном формате, с датой можно производить ещё некоторые манипуляции.

- Функция `gmdate()` форматирует дату/время по Гринвичу.

Имеет следующий прототип:

```
string gmdate ( string $format [, int $timestamp] )
```

Список параметров форматирования аналогичен функции `date()`, разница лишь в том, что возвращается время по Гринвичу:

К примеру, если эту функцию вызвать в Латвии (GMT +0200), то строка

```
echo date("M d Y H:i:s", mktime(0, 0, 0, 1, 1, 1998));
```

выведет "Jan 01 1998 00:00:00",

```
а строка echo gmdate("M d Y H:i:s", mktime(0, 0, 0, 1, 1, 1998));
```

выведет "Dec 31 1997 22:00:00".

- Функция `strtotime()` пытается текстовое представление даты на английском языке в метку времени Unix. В случае невозможности перевести полученную дату в нормальный формат, функция вернёт -1.

Пример: Допустимые значения формирования даты

```
<?php
echo strtotime("now"), "\n";
echo strtotime("10 September 2000"), "\n";
echo strtotime("+1 day"), "\n";
echo strtotime("+1 week"), "\n";
echo strtotime("+1 week 2 days 4 hours 2 seconds"), "\n";
echo strtotime("next Thursday"), "\n";
echo strtotime("last Monday"), "\n";
?>
```

Работа с базами данных MySQL

Для MySQL подключение и авторизация объединены в одной функции.

```
$link=mysql_connect("host","user","password");
```

Получив идентификатор соединения с базой мы можем с ним работать, к примеру, можно создать новую базу используя следующую функцию:

```
mysql_create_db("databasename",$link);
```

После завершения работы с базой, соединение нужно закрыть. Это делается с использованием следующей функции:

```
mysql_close($link);
```

Для выбора базы данных, с которой предстоит работать используется функция

```
bool mysql_select_db ( string $database_name [, resource $link_identifier] )
```

Для того, чтобы выполнить запрос к таблице, можно использовать функцию

```
resource mysql_query ( string $query [, resource $link_identifier] )
```

Тут, `$query` строка с запросом, если параметр `link_identifier` опущен, используется последнее открытое соединение. Если открытые соединения отсутствуют, функция пытается соединиться с СУБД, аналогично функции `mysql_connect()` без параметров.

Получение списка всех пользователей в системе

Логика этого действия простая, нужно получить из базы данных все записи и вывести их в браузер, в цикле формируя таблицу, по одной строке на запись.

Создадим для этого скрипт `get_all_users_form.php`:

```
<?php
header( "Content-Type: text/html; charset=utf-8;" );
header( "Content-Encoding: utf-8" );

$link=mysql_connect("localhost","root","") or die("Unable to connect to database");
mysql_select_db("tsi_db") or die("Unable to select database");

$query_results      =      mysql_query("SELECT * FROM `users` WHERE 1");

mysql_close($link);

$colorOne      =      "#80FFFF";
$colorTwo      =      "#FFFF80";
$curColor      =      $colorOne;

?>
<a href="8_lekcija.php">
    &lt;&lt;&lt;НАЗАД
</a>

<h1>Пользователи в системе</h1>
<table border="1" width="70%">

    <tr bgcolor="<?=$curColor?>" >
```

<pre> <td width="20%" align="center" > ID </td> <td align="center" > Редактировать </td> </tr> <? while(\$user_data = mysql_fetch_array(\$query_results)){ ?> <tr bgcolor="<?=\$curColor?>" > <td width="20%" align="center" > <?=\$user_data['uID']?> </td> <td > <a href="show_user_data.php?uID=<?=\$user_data['uID']?>" ><? =\$user_data['name']?> </td> </tr> <? \$curColor = (\$curColor == \$colorOne) ? \$colorTwo : \$colorOne; } ?> </table> </pre>

Дополнительно рассмотрим некоторые полезные функции для работы с MySQL

mysql_num_rows	Функция вернёт количество рядов полученных в ответе на запрос. Замечание: Функция работает только с SQL оператором выборки SELECT
mysql_affected_rows	Функция вернёт количество рядов полученных в ответе на запрос. Замечание: Функция работает с операторами SQL INSERT, UPDATE, DELETE
mysql_escape_string	Экранирует спецсимволы в строке запроса к базе данных
mysql_fetch_object	Обрабатывает результат запроса аналогично функции mysql_fetch_array(), но вместо массива возвращает объект.
mysql_get_server_info	Вернёт информацию о MySQL сервере
mysql_ping	Проверяет статус соединения с сервером, если оно утеряно, то происходит дополнительное соединение.