

ПІДГОТОВКА ВЧИТЕЛІВ ІНФОРМАТИКИ ДО НАВЧАННЯ УЧНІВ ОСНОВ ВЕБ-ДИЗАЙНУ

Літош Ю.М.

Основним засобом для всесвітньої та міжгалузевої інтеграції є комп'ютерні мережі. У віртуальному просторі відкриваються можливості для різних видів діяльності. Представники багатьох спеціальностей, які раніше практично не потребували вмінь працювати в мережах, тепер починають активно використовувати всі засоби, що надаються мережевими технологіями. Тому значно зростає попит на професії, що пов'язані з роботою в комп'ютерних мережах та їх обслуговуванням.

З огляду на це, серед головних завдань шкільного курсу інформатики є формування в учнів розуміння принципів організації локальної та глобальної комп'ютерних мереж, вироблення вмінь та навичок роботи із сучасними програмними засобами, призначеними для розробки веб-сайтів з елементами інтерактивності. Цю мету можна реалізувати проведенням курсу за вибором «Основи веб-дизайну» профільного навчання або факультативів, гуртків.

У статті розглянемо створення таких елементів інтерактивності, як HTML-форм, їх опрацювання і способи передавання даних з форм за допомогою мови програмування PHP. Мова програмування PHP створена для генерації HTML-сторінок на веб-сервері та роботи з базами даних, підтримується переважною більшістю представників хостингів.

Як показує практика, для організації занять достатньо в комп'ютерному класі налагодити мережу Intranet. Усе програмне забезпечення (ПЗ), необхідне для її організації, можна встановити на виділеному комп'ютері локальної мережі. До складу ПЗ повинні входити:

1. Програма для установки веб-серверу, наприклад Apache. Переписати файл інсталяції можна із сайту <http://apache.org>, а установка описана в [1].

2. Програма для установки і налагодження на веб-сервері підтримки мови програмування PHP. Дистрибутив програми можна завантажити із сайту <http://php.su>, а опис налагодження наведений в [1].

3. Програма для установки сервера баз даних MySQL. Однією з причин популярності цього серверу можна вважати його інтеграцію з PHP. До стандартного складу PHP входять бібліотеки для роботи з MySQL, надаючи розробникам усі необхідні інструменти для взаємодії із сервером баз даних.

4. Інтерфейсом баз даних MySQL може слугувати програма phpMyAdmin — PHP додаток, що працює на веб-сервері. Ця програма може керувати як MySQL-сервером, так і окремою базою даних.

Вона забезпечує:

- створення і видалення баз даних;
- створення, копіювання, видалення та зміну таблиць;

- видалення, зміну і додавання полів.

Для створення, модифікації та керування даними в реляційних базах даних програма phpMyAdmin використовує мову запитів SQL. Завантажити інсталяційний файл програми й опис установки можна із [2].

5. Для передавання файлів на веб-сервер слід встановити FTP-сервер, який можна переписати за адресою <http://typsoft.ru>.

Створимо файл `forma.php`, що містить html-форму. У цій html-формі найважливішими атрибутами є: `action`, який вказує шлях до іншого файлу, що буде обробляти дані форми, імена текстових полів (`lastname`, `firstname`, `patronym`, `email`, `inforinfouser`). Також у формі є кнопка **submit**, після натискання на яку відбудеться передавання даних. Для написання програм зручно використовувати редактор PHP Expert Editor (рис. 1).

Файл `forma.php`:

```
<html>
<head>
<meta http-equiv=content-type content=text/html>
<title>База даних учнів школи</title>
</head>
<body text=#000000 bgcolor=#52fa90>
<br><center><table width=90% border=1
bgcolor=green cols=1>
<tr><td><p style=text-align:justify;
margin-left: 50px; margin-right: 50px>
<h3>Інформація про наступного учня</h3>
```

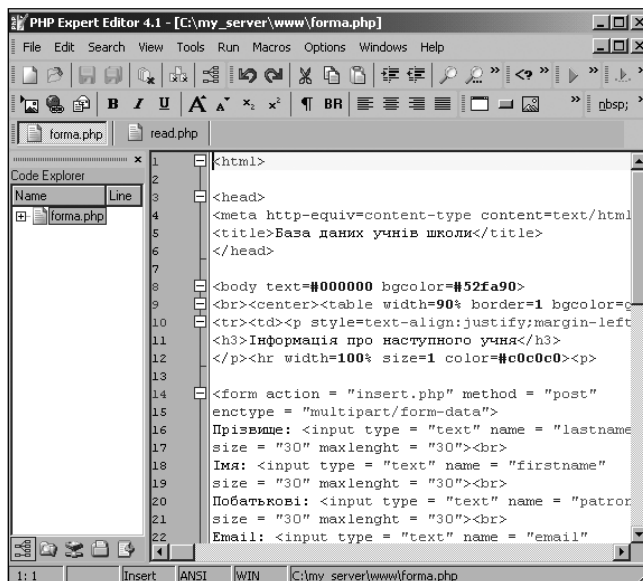


Рис. 1

```

</p><hr width=100% size=1 color=#c0c0c0><p>
<form action = "insert.php" method = "post"
enctype = "multipart/form-data">
Прізвище: <input type = "text" name = "lastname"
size = "30" maxlength = "30"><br>
Ім'я: <input type = "text" name = "firstname"
size = "30" maxlength = "30"><br>
Побатькові: <input type = "text" name = "patronym"
size = "30" maxlength = "30"><br>
Email: <input type = "text" name = "email"
size = "30" maxlength = "30"><br>
Адреса: <br>
<textarea name = "inforefouser" cols = "40"
rows = "5"></textarea>
<br>
<input type = "submit" name = "submit"
value = "Відправити дані"><br>
</form>
</td></tr></table></center>
</body>
</html>

```

Після того, як будуть введені будь-які значення в текстові поля і натиснута кнопка **Відправити дані**, html-форма відправить значення файлу insert.php.

Файл insert.php:

```

<html>
<body text=#000000 bgcolor=#52fa90>
<?php
$dbhost="localhost";
$dbuser="root";
$dbpassword="";
$dbname="schooldb";
$link=mysql_connect($dbhost,$dbuser,$dbpassword);
mysql_query("set character_set_client='cp1251'");
mysql_query("set character_set_results='cp1251'");
mysql_query("set collation_connection=
'cp1251_general_ci'");
mysql_select_db($dbname,$link);
$query="INSERT INTO `infouser` (`lastname`,
`firstname`, `patronym`, `email`, `inforefouser`)
VALUES ('$lastname', '$firstname', '$patronym',
'$email', '$inforefouser')";
mysql_query($query,$link);
//echo "Інформація занесена до бази даних";
mysql_close($link);
?>
<br><center><table width=90% border=1
bgcolor=green cols=1>
<tr><td><p style=text-align:justify;
margin-left: 50px; margin-right: 50px>
<h3>Інформація про наступного учня</h3>
</p><hr width=100% size=1 color=#c0c0c0><p>
<form action = "insert.php" method = "post"
enctype = "multipart/form-data">
Прізвище: <input type = "text" name = "lastname"
size = "30" maxlength = "30"><br>
Ім'я: <input type = "text" name = "firstname"
size = "30" maxlength = "30"><br>
Побатькові: <input type = "text" name = "patronym"
size = "30" maxlength = "30"><br>

```

```

Email: <input type = "text" name = "email"
size = "30" maxlength = "30"><br>
Адреса: <br>
<textarea name = "inforefouser" cols = "40"
rows = "5"></textarea>
<br>
<input type = "submit" name = "submit"
value = "Відправити дані"><br>
</form>
</td></tr></table></center>
</body>
</html>

```

Значення текстових полів, що містяться в змінних \$lastname, \$firstname, \$patronym, \$email, \$inforefouser, програма-сценарій insert.php передає базі даних schooldb і заповнює ними таблицю з іменем infouser. Для цього потрібно створювати програму за таким алгоритмом:

1. Налаштувати з'єднання із сервером MySQL.
2. Вибрати потрібну базу даних.
3. Зробити запит до серверу баз даних (sql запит).
4. Опрацювати результат запиту.
5. Закрити з'єднання (від'єднатися від серверу баз даних).

Для того щоб під'єднатися до серверу MySQL, треба передати йому чотири параметри: назву хостингу, на якому розташований сервер баз даних (localhost); ім'я користувача, який може працювати з цією таблицею (root); пароль користувача (за замовчуванням пустий рядок ""); ім'я бази даних (schooldb).

Під'єднатися до серверу можна за допомогою команди: \$link=mysql_connect("Хост", "Користувач", "Пароль"). Цей ідентифікатор з'єднання необхідно вказувати у всіх mysql функціях. Вибрати базу даних: mysql_select_db("Ім'я бази даних", \$link). Закрити підключення треба командою mysql_close(\$link).

Керування базою даних (БД) в MySQL здійснюється за допомогою мови запитів SQL. Запит до серверу MySQL здійснюється за допомогою функції mysql_query(). У параметрах вказується запит \$query та з'єднання \$link. У програмі insert.php представлений запит, за яким виконується додавання записів до таблиці infouser.

Далі розглянемо створення бази даних schooldb та таблиці infouser, за допомогою інтерфейсу адміністрування баз даних phpMyAdmin та php-скрипту.

У вікні **Создать новую БД** програми phpMyAdmin треба ввести ім'я бази даних schooldb і виконати команду **Создать** (рис. 2).

Наступне вікно повідомляє про створення БД (рис. 3).

Для створення таблиці можна використовувати вкладку **Структура** на сторінці перегляду бази даних. Щоб створити таблицю, можна скористатись полем **Створити таблицю**, у якому треба задати ім'я infouser та кількість полів (шість). Потім вказуємо список стовпців. Кожному стовпцю надається ім'я, вказується тип даних, атрибут null або not null, значення за замовчуванням.

Стовпець id_infouser є первинним ключем (primary key) й унікально ідентифікує кожний запис. Цьому стовпцю надані тип даних int (ціле число) і

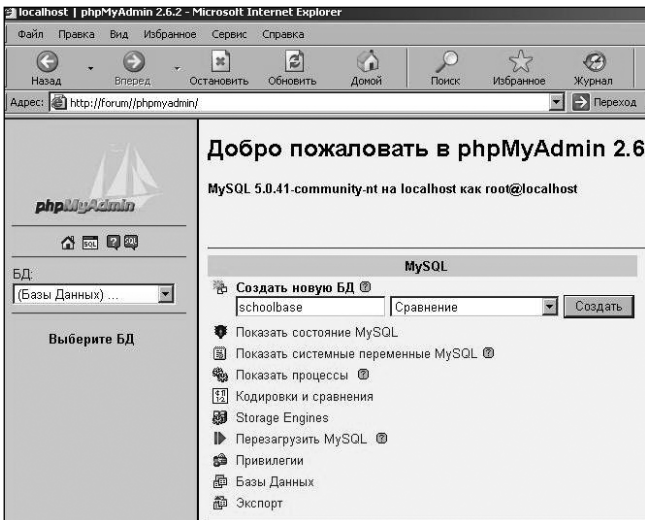


Рис. 2

атрибут `auto_increment`, завдяки чому він є лічильником. Для інших стовпців типами даних є рядки змінної довжини (`varchar`), для яких вказується максимальна довжина. Далі треба натиснути кнопку **Зберегти** (рис. 4).

На основі наших дій через інтерфейс програма phpMyAdmin згенерувала наступний sql-запит.

```
id_infouser int(7) not null auto_increment
lastname varchar(255) not null
firstname varchar(255) not null
patronym varchar(255) not null
email varchar(100) not null
infoinfouser varchar(255) not null
```

Також створити базу даних та таблицю можна за допомогою php-скрипту `create.php`, відкритого у браузері. Ця програма використовує sql-запити, згенеровані в програмі phpMyAdmin (рис. 5). Тому phpMyAdmin може бути інструментом для вивчення мови SQL.

Файл `create.php`:

```
<?php
$dbhost="localhost";
$dbuser="root";
$dbpassword="";
$link=mysql_connect($dbhost,$dbuser,$dbpassword);
mysql_query("CREATE DATABASE `schooldb`");
mysql_select_db("schooldb");
$query=mysql_query("CREATE TABLE `infouser`
(`id_infouser` int(7) not null auto_increment,
`lastname` varchar(255) not null,
`firstname` varchar(255) not null,
`patronym` varchar(255) not null,
```

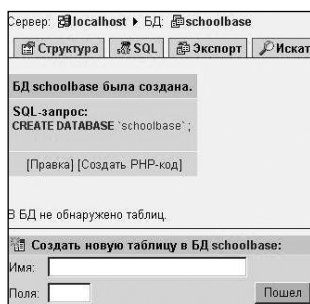


Рис. 3

Поле	Тип	Довжини/Значення
id_infouser	INT	7
lastname	VARCHAR	255
firstname	VARCHAR	255
patronym	VARCHAR	255
email	VARCHAR	100
infoinfouser	VARCHAR	255

Рис. 4

```
`email` varchar(100) not null,
`infoinfouser` varchar(255) not null,
PRIMARY KEY (`id_infouser`));");
mysql_close($link);
echo "База даних створена";
?>
```

Далі перейдемо до процедури запити даних з бази даних. Результати запити відтворимо у вигляді звичайної html-таблиці. Наступний файл `vivid.php` дозволяє підключитися до серверу MySQL, зробити вибірку даних із таблиці `infouser`, що відповідають запити користувача, переданого файлом `zapit.php`.

Файл `zapit.php`:

```
<html>
<head>
<title>Робота з базою даних</title>
<meta http-equiv=content-type content=text/html;
charset=windows-1251>
</head>
<body>
<table width=75% border=1 align=center
bgcolor=#ccccff>
<tr><td colspan=3 align=center>
Вивід інформації з бази даних<hr>
<form action="vivid.php" method="post">
Введіть прізвище: <input type="text"
name="fam" value="">
<input type="submit" name="submit"
value="Знайти">
</form></td></tr></table>
</body>
</html>
```

Файл `vivid.php`:

```
<?php
$dbhost="localhost";
$dbuser="root";
$dbpassword="";
$dbname="schooldb";
$link=mysql_connect($dbhost,$dbuser,$dbpassword);
mysql_query("set character_set_client='cp1251'");
mysql_query("set character_set_results='cp1251'");
mysql_query("set collation_connection='cp1251_
general_ci'");
mysql_select_db($dbname,$link);
echo "<table border='1' width='100%' \
\"bgcolor='\" #ffffe1\">";
echo "<tr><td>Прізвище</td><td>ім'я </td><td>
побатькові</td>";
echo "<td>email</td></tr>";
$query="SELECT * FROM `infouser` where `last-
name`='`$fam`";
$query=mysql_query($query,$link);
for($c=0;$c<mysql_num_rows($query);$c++)
{
echo "<tr>";
```

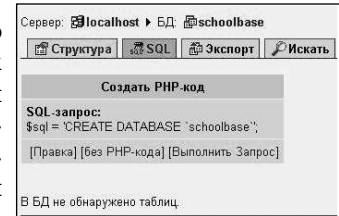


Рис. 5

```
$f=mysql_fetch_array($q);
echo "<td>${f[lastname]}</td><td>${f[firstname]}
</td><td>${f[patronym]}</td>";
echo "<td>${f[email]}</td>";
echo "</tr>";
}
echo "</table>";
mysql_free_result($q);
mysql_close($link);
?>
```

Для відображення даних усієї таблиці infouser треба в програмі vivid.php рядок \$query = "SELECT * FROM `infouser` where `lastname`='\${fam}'"; замінити рядком \$query = "SELECT * FROM `infouser`";

Зберігати дані, передані користувачем за допомогою форм, можна не тільки в базах даних, а й у файлах. Далі розглянемо створення програми write.php для запису даних із форми file.php у файл. Робота з файлами розділяється на три етапи:

1. Відкриття файлу в потрібному режимі.
2. Маніпуляції з даними (запис, зчитування).
3. Закриття файлу.

Відкриття файлу здійснюється за допомогою команди fopen("filename", "mode"). Під час відкривання файлу велике значення має режим роботи з ним (параметр mode). У програмі використаний режим «a+». Він дозволяє записувати дані у файл і зчитувати з нього. За цього режиму створюється файл, якщо був відсутнім, та автоматично додаються дані в кінець файлу. Функція fputs() записує дані у файл відповідно до вибраного режиму.

Файл file.php:

```
<html>
<head>
<meta http-equiv=content-type content=text/html>
<title>База даних учнів школи</title>
</head>
<body text=#000000 bgcolor=#52fa90>
<br><center><table width=90% border=1
bgcolor=green cols=1>
<tr><td><p style=text-align:justify;
margin-left: 50px; margin-right: 50px>
<h3>Інформація про наступного учня</h3>
</p><hr width=100% size=1 color=#c0c0c0><p>
<form action = "write.php" method="post"
enctype = "multipart/form-data">
Прізвище: <input type = "text" name = "str"
size = "50" maxlength = "50"><br>
Ім'я:<input type="text" name="st"
size="50" maxlength="50"><br>
<input type = "submit" name="submit"
value = "Ввести інформацію про нового учня">
<br>
</form>
</td></tr></table></center>
</body>
</html>
```

Файл write.php

```
<?
$file = fopen ("file.txt", "a+");
```

```
if (!$file)
{
echo("Помилка відкриття файла");
}
else
{
fputs ($file, "$str | $st\r\n");
}
fclose ($file);
?>
```

За допомогою наступної програми read.php можна зчитувати інформацію з файлу і відобразити на веб-сторінці. За допомогою функції file() дані з файлу зчитуються у масив. Кожний елемент масиву складається з окремого рядка файлу, а номер рядка стає індексом у масиві. Потім дані масиву виводяться у браузері.

Файл read.php:

```
<html>
<head>
<title>Робота з базою даних</title>
<meta http-equiv=content-type content=text/html;
charset=windows-1251>
</head>
<body>
<?
$file_array = file("file.txt");
if(!$file_array)
{
echo("Помилка відкриття файла");
}
else
{
for($i=0; $i < count($file_array); $i++)
{
printf("%s<br>", $file_array[$i]);
}
}
?>
</body>
</html>
```

Розглянуті приклади стають основою для постановки більш складних задач. Розроблений автором спецкурс уже декілька років викладався на курсах підвищення кваліфікації вчителів інформатики в системі післядипломної педагогічної освіти. Результати проведення таких занять дозволяють зробити такі висновки:

- навчання має завершуватися виконанням проекту з розробки тематичного веб-сайту, який може брати участь в конкурсах, зокрема з веб-дизайну.
- проведення таких занять уможлиблюють впровадження в навчальний процес навчально-дослідницької праці як важливого засобу формування в учнів стійкого інтересу і готовності до творчої діяльності.

Література

1. Кухарчик А. PHP: обучение на примерах. — Мн.: Новое знание, 2004. — 237 с.
2. Використання phpMyAdmin // www.php-myadmin.ru.