

Лабораторная работа №1: Установка набора дистрибутивов Denwer

В данной лабораторной работе рассматриваются установка набора дистрибутивов, необходимых для разработки серверных приложений с помощью языка программирования PHP и настройка инструментальной среды Adobe Dreamweaver на работу с виртуальным сервером.

Упражнение 1: Установка набора дистрибутивов Denwer

В данном упражнении будет продемонстрирована установка набора дистрибутивов, в состав которого входят дистрибутивы Apache, PHP, MySQL, phpMyAdmin и других систем и приложений, необходимых для организации разработки серверных приложений с использованием средств языка программирования PHP.

1. Скачайте дистрибутив базового комплекта Denwer - Denwer_base (около 2Мб), самую последнюю версию комплекта Denwer можно взять с сайта - www.denwer.ru/dis/Base.
2. Запустите инсталлятор. Вначале архив будет автоматически распакован во временную директорию, а затем автоматически запустится инсталлятор.
3. По умолчанию для установки комплекса используется директория C:\WebServers, нажмите Enter, чтобы согласиться с этим выбором. В указанном каталоге будут расположены абсолютно все компоненты системы, и вне его никакие файлы в дальнейшем не создаются (исключая ярлыки на Рабочем столе).
4. Далее вам предложат ввести имя виртуального диска, который будет связан с только что указанной директорией. Рекомендуется согласиться со значением по умолчанию (Z:). Важно, что диска с этим именем еще не должно содержаться в системе — чаще всего так и происходит с диском Z:.
5. После этого начнется копирование файлов дистрибутива. В ходе установки вам будет предложена установка ярлыков на Рабочий стол – это необходимо сделать для дальнейшего удобства работы.
6. В конце установки будет задан вопрос, как именно вы собираетесь запускать и останавливать комплекс (2 варианта):

- Автоматически создавать виртуальный диск при загрузке машины (а при остановке серверов этот диск не отключать);
 - Создавать виртуальный диск только по явной команде старта комплекса (при щелчке по ярлыку запуска на Рабочем столе). И, соответственно, отключать диск от системы — при остановке серверов.
- Рекомендуемый вариант – второй.**

7. Установка завершена. На Рабочем столе Windows дважды щелкните на ярлычке Start Denwer (если вы не создавали ярлыки, то можно запустить Денвер по команде C:\WebServers\denwer\Run.exe).
8. Дождавшись, когда все консольные окна исчезнут, откройте браузер и наберите в нем адрес: <http://localhost>. В случае успешной установки откроется страница, оповещающая о этом (см. Рис. 1.1).

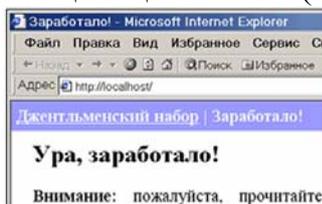


Рис.1.1. Окно страницы <http://localhost>

9. Изучите информацию страницы <http://localhost/>. Обратите особое внимание на разделы посвященные тестированию и работе с имеющимися утилитами.

Упражнение 2: Настройка Adobe Dreamweaver на работу с сервером Apache

В данном упражнении будет продемонстрирована настройка инструментальной среды разработки Adobe Dreamweaver для работы с виртуальным сервером Apache.

1. Запустите программу *Adobe Dreamweaver*.
2. Выберите пункты меню **Веб-сайт - Управление веб-сайтами**. Откроется диалоговое окно «Управление сайтами». В нем нажмите кнопку «Создать». Откроется окно настройки сайта. На вкладке **Веб-сайт** окна настройки сайта введите:
 - имя сайта: «*MyTravelNotes*»;
 - укажите путь к локальной папке его расположения:
C:\WebServers\home\localhost\www\

3. В том же окне перейдите на вкладку **Серверы**, где необходимо задать параметры виртуального сервера. Для этого нажмите «+». Откроется окно настроек сервера.
4. На вкладке **Базовый** укажите:
 - Имя сервера: *localhost*
 - Подключение с помощью: *Локальный / Сетевой*
 - Папка сервера: *C:\WebServers\home\localhost\www*
 - URL-адрес: *http://localhost*
5. В том же окне перейдите на вкладку **Дополнительно**:
 - В разделе **Удаленный сервер** установите галочку напротив пункта «*Сохранить сведения о синхронизации*».
 - В разделе **Тестовый сервер** из выпадающего меню **Серверная модель** выберите *PHP MySQL*.
6. Сохраните параметры. Обратите внимание, что в окне «Настройка сайта» на вкладке **Серверы** должны стоять два флага для localhost: **Удаленный** и **Тестовый** (если их нет - поставьте их самостоятельно).
7. В окне **Управление сайтами** появится имя созданного проекта сайта «*MyTravelNotes*». Нажмите кнопку **Готово**. Для корректировки, удаления и других операций с проектом сайта используйте соответствующие кнопки управления в данном окне (рис. 1.2).

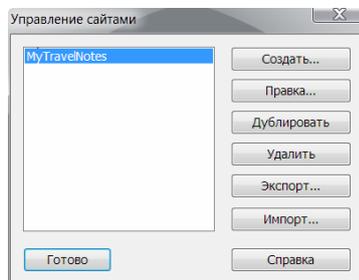


Рис. 1.2. Окно управления проектом сайта.

Упражнение 3: Первая программа на PHP

В этом упражнении Вы напишите программу “Hello, world!” на языке PHP, используя среду разработки Adobe Dreamweaver.

1. Создайте в программе Adobe Dreamweaver новую страницу (**Файл – Создать**. Выберите тип страницы PHP).
2. Введите следующий PHP-код:

```
<BODY>
<?PHP
echo "Hello, world!";
?>
</BODY>
```

3. Сохраните сценарий в папку сайта под именем hello.php.
4. Проверьте результат в браузере (F12).

При просмотре результата выполнения файла hello.php в браузере просмотрите html-код (*например, в IE меню Вид – Просмотр HTML-кода*). Обратите внимание на то, что php-кода на странице нет – это значит, что php-сценарий был обработан сервером, после чего сервер передал в браузер результат обработанного php-сценария.

Упражнение 4: Простейшие программы на PHP

Далее представлены дополнительные упражнения для закрепления навыков создания простейших программ в среде Adobe Dreamweaver на языке программирования PHP. Для их выполнения в Adobe Dreamweaver создайте новый .php файл (**Файл – Создать – Выбрать тип файла .php**) `examples.php` и выполните предложенные далее задания.

1. Переменной \$a необходимо присвоить значение 10, переменной \$b присвоить значение 20. Выведите значения переменных на экран.
2. Затем переменной \$c присвойте значение суммы этих переменных (переменной \$a и переменной \$b). Выведите значение переменной \$c на экран.
3. Далее увеличьте значение переменной \$c в три раза и выведите полученный результат на экран.
4. Разделите переменную \$c на разность переменных \$b и \$a, выведите результат на экран.
5. Введите новые переменные \$p и \$b. Присвойте переменной \$p значение «Программа», а переменной \$b значение «работает».
6. Затем сложите переменные, содержащие эти слова («Программа» и «работает»), при этом слова должны быть разделены пробелом (' '). Результат необходимо присвоить переменной \$result.
7. Далее с помощью оператора «.=» необходимо к строке «Программа работает» добавит слово «хорошо». Результат необходимо присвоить переменной \$result.
8. Есть две переменные: \$q = 5 и \$w = 7. Создайте скрипт, в результате выполнения которого эти две переменные «обмениваются» значениями –

переменная \$q получает значение 7, переменная \$w получает значение 5, при этом не создавая новых переменных (вариант \$q = 7 и \$w = 5 не рассматривается).

Лабораторная работа №2: Создание статического каркаса сайта. Работа с инструментарием среды разработки Adobe Dreamweaver

В данной лабораторной работе иллюстрируется создание двух первых страниц сайта – **blog.html** и **inform.html** в статическом виде с помощью средств разработки инструментальной среды Adobe Dreamweaver. Страница **blog.html** является первой страницей сайта, должна загружаться в браузере и содержать собственно заметки автора блога. Страница статистики **inform.html** будет вспомогательной страницей, содержащей статистическую информацию о размещенных на сайте заметках и комментариях.

Упражнение 1: Настройка Adobe Dreamweaver

В данном упражнении продемонстрирован процесс создания web-проекта в инструментальной среде Adobe Dreamweaver.

1. Запустите Denwer.
2. Запустите Adobe Dreamweaver, откройте многофункциональное окно **Настройки (Правка – Настройки)**, категория **Создать документ**. Установите тип документа по умолчанию (DTD) – HTML5 и кодировку по умолчанию – кириллица (Windows).

Упражнение 2: Создание статической основы web-страниц

В данном упражнении необходимо создать две первые статические страницы проекта, которые станут основой для дальнейшей разработки. Страница **blog.html** является первой страницей сайта, должна загружаться в браузере и содержать собственно заметки автора блога. Страница статистики **inform.html** будет вспомогательной страницей, содержащей статистическую информацию о размещенных на сайте заметках и комментариях.

1. В Dreamweaver перейдите в меню **Файл – Создать**.
2. Создайте статичный документ в формате HTML для будущей страницы с названием **blog.html**. На странице должно располагаться меню переходов

между страницами и место размещения основного контента сайта (рис.2.1):

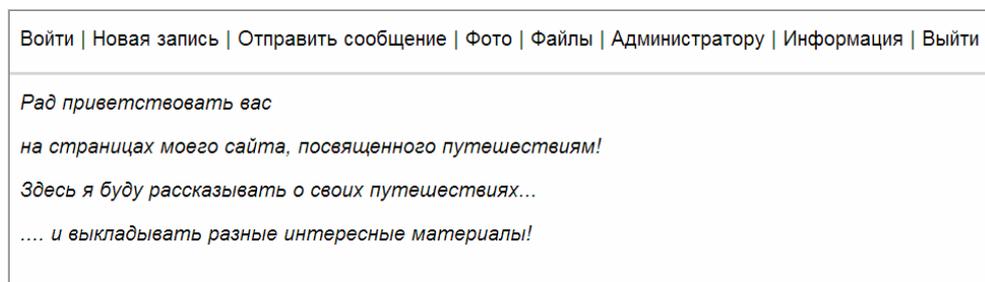


Рис. 2.1. Пример страницы blog.html

3. Сохраните страницу под именем blog.html.
4. Создайте страницу статистики inform.html. Схема страницы (рис. 2.2):

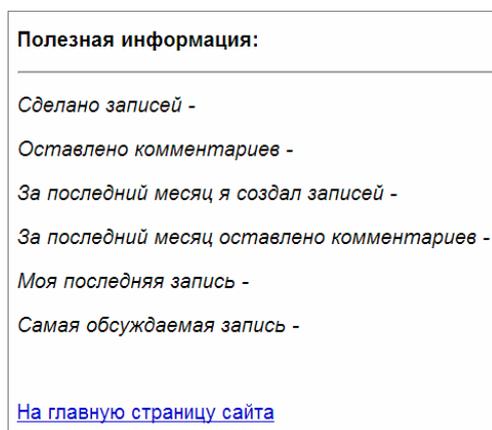


Рис. 2.2. Пример страницы inform.html

5. Сохраните таблицу под именем inform.html.
6. Свяжите гиперссылками созданные страницы (меню **Изменить – Создать ссылку** или вручную средствами языка разметки HTML).

Лабораторная работа №3: Создание базы данных MySQL

В ходе выполнения данной лабораторной работы необходимо создать в MySQL новую базу данных с названием «*MySiteDB*» и добавить в нее две таблицы: **notes** и **comments**. **Notes** содержит заметки блога; **comments** – комментарии к этим заметкам. Схема данных (рис.3.1):

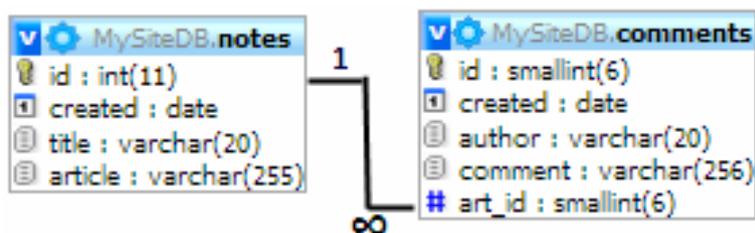


Рис.3.1. Схема базы данных “MySiteDB”

В Приложении 4 представлена информация об основных понятиях, необходимых для работы с базой данных.

Упражнение 1: Создание БД «MySiteDB»

В этом упражнении реализуется запрос на создание новой базы данных.

1. Создайте новый php документ, который будет называться **create_db.php**.
2. Создайте соединение с сервером localhost. Имя сервера *localhost*, пользователь *root*, пароля нет.
3. Создайте базу данных:
 - 3.1. Сформируйте запрос на создание базы *MySiteDB* с использованием SQL;
 - 3.2. Реализуйте запрос на создание БД с помощью функции `mysqli_query()`.
4. Сохранить документ, выполнить запрос.
5. С помощью утилиты **PhpMyAdmin** убедитесь, что создана новая база данных. Для этого запустите утилиту: `http://localhost/tools/phpmyadmin` (или `http://localhost` и выберите PhpMyAdmin из списка утилит).
6. Вторично выполните запрос, чтобы убедиться, что соединение есть, а база не создается (т.к. она была уже создана ранее, в ходе предыдущего выполнения скрипта).
7. Желательно добавить цикл `if` для обнаружения неполадок в работе.

Вариант реализации создания БД MySiteDB

```

<?php
//Создать соединение с сервером
$link = mysqli_connect ("localhost", "root", "");
if ($link) {
    echo "Соединение с сервером установлено", "<br>";
} else {
    echo "Нет соединения с сервером";
}

```

```

//Создать БД MySiteDB
//Сначала формирование запроса на создание
$db = "MySiteDB";
$query = "CREATE DATABASE $db";

//Затем реализация запроса на создание. Важна последовательность
аргументов функции: соединение с сервером, SQL-запрос.
$create_db = mysqli_query($link, $query);
if ($create_db) {
    echo "База данных $db успешно создана";
} else {
    echo "База не создана";
}
?>

```

Упражнение 2: Создание пользователя admin

В этом упражнении Необходимо создать нового пользователя базы данных с именем **admin** и паролем **admin** с правами администратора. Пользователей можно добавлять двумя способами:

- при помощи SQL-запроса GRANT
- в таблице назначения привилегий MySQL (Privileges) с помощью утилиты PhpMyAdmin.

Выберите один из двух приведенных далее способов.

Способ 1: создание нового пользователя с помощью SQL-запроса GRANT

1. Создайте новый php-документ, который будет называться create_user.php;
2. Создайте соединение с сервером;
3. Сформируйте SQL-запрос на создание нового пользователя базы данных:

```

$query = "GRANT ALL PRIVILEGES ON *.* TO 'admin'@'localhost'
IDENTIFIED BY 'admin'
WITH GRANT OPTION";

```

//. * - глобальный уровень привилегий, применяется ко всем базам на сервере.*

4. Реализуйте запрос.

- Проверка создания пользователя. С помощью утилиты PhpMyAdmin убедитесь, что создан новый пользователь. Для этого запустите утилиту PhpMyAdmin и перейдите на вкладку Privileges. Изучите список пользователей.

Способ 2: создание нового пользователя с помощью утилиты PhpMyAdmin

1. Запустите утилиту PhpMyAdmin и перейдите на вкладку Privileges. Нажмите кнопку «Add a new user».
2. Введите имя пользователя (**admin**), имя сервера (**localhost**), пароль с подтверждением (**admin**). Предоставьте новому пользователю все права (**global privileges – Check All**).
3. Убедитесь, что новый пользователь создан корректно.
4. Все дальнейшие действия с базой данных будут проводиться под пользователем **admin** с паролем **admin** и соответствующими правами, если иное не указано в задании.

Упражнение 3: Создание таблицы notes

В данном упражнении будет продемонстрирован один из способов создания таблиц в ранее созданной базе данных на примере создания таблицы **notes**. Таблица **notes** содержит заметки автора блога. Данная таблица будет создана средствами PHP. Информацию о полях таблицы см. в *Приложении 3*.

1. Создайте новый php-документ, который будет называться **create_tbl.php**;
2. Создайте соединение с сервером уже под созданным ранее пользователем **admin** с паролем **admin**.
3. Подключитесь к базе данных MySiteDB.
4. Сформируйте запрос на создание таблицы **notes** с полями, указанными в *Приложении 3*.

//Формирование запроса

```
$query = "CREATE TABLE notes
        (id INT NOT NULL AUTO_INCREMENT,
        PRIMARY KEY (id),
        created DATE,
        title VARCHAR (20),
        article VARCHAR (255))";
```

5. Реализуйте запрос на создание таблицы.

6. С помощью утилиты PhpMyAdmin убедитесь, что создана новая таблица. Для этого запустите утилиту, перейдите к базе данных MySiteDB и просмотрите ее структуру. В ней должна появиться соответствующая таблица.

Вариант реализации создания таблицы notes

```
<?php
//Соединение с сервером
$link = mysqli_connect ('localhost', 'admin', 'admin');

//Выбор БД
$db = "mySiteDB";
$select = mysqli_select_db($link, $db);
if ($select){
    echo "База успешно выбрана", "<br>";
} else {
    echo "База не выбрана";
}
//Создание таблицы
//Формирование запроса
$query = "CREATE TABLE notes
        (id INT NOT NULL AUTO_INCREMENT,
        PRIMARY KEY (id),
        created DATE,
        title VARCHAR (20),
        article VARCHAR (255))";

//Реализация запроса
$create_tbl = mysqli_query ($link, $query);
if ($create_tbl){
    echo "Таблица успешно создана", "<br>";
} else {
    echo "Таблица не создана";
}
?>
```

Упражнение 4: Создание таблицы comments

В данном упражнении будет продемонстрирован другой способ создания таблиц в ранее созданной базе данных на примере создания таблицы **comments**. Таблица **comments** содержит комментарии пользователей к

заметкам автора блога. Таблица будет создана с помощью утилиты PhpMyAdmin. Информацию о полях таблицы см. в *Приложении 3*.

1. Запустите браузер.
2. Запустите утилиту *phpMyAdmin*. В главном окне *PHPMysqlAdmin* выберите БД *MySiteDB*.
3. В поле “Create new table”, присвойте имя таблице – *comments*; количество полей - 5, нажмите кнопку «Go» (*рис. 3.2*).

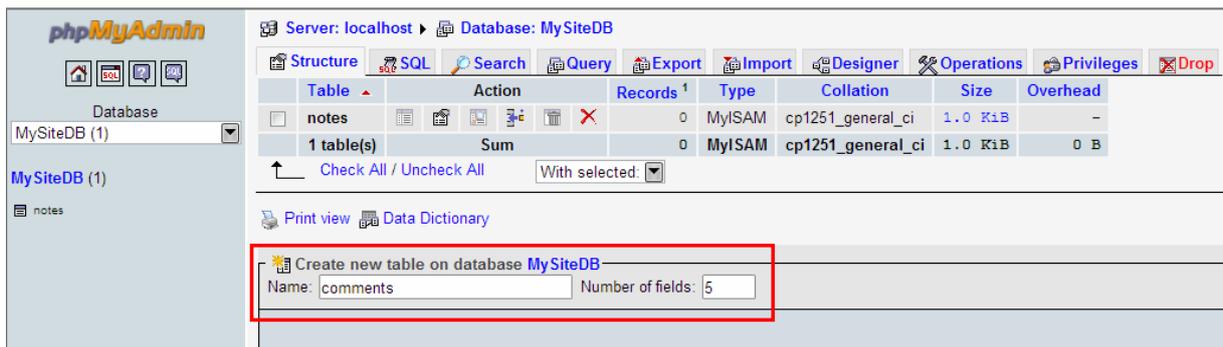


Рис. 3.2. Создание новой таблицы с помощью утилиты phpMyAdmin.

4. Создание полей таблицы *comments*:
 - 4.1. В открывшемся окне заполните необходимые поля таблицы (*рис. 3.3*) и нажмите кнопку «Save».
 - 4.2. Для поля *id* добавьте следующие атрибуты: обозначьте автоинкремент *A_I* и первичный ключ *PRIMARY* в поле со списком *INDEX*.

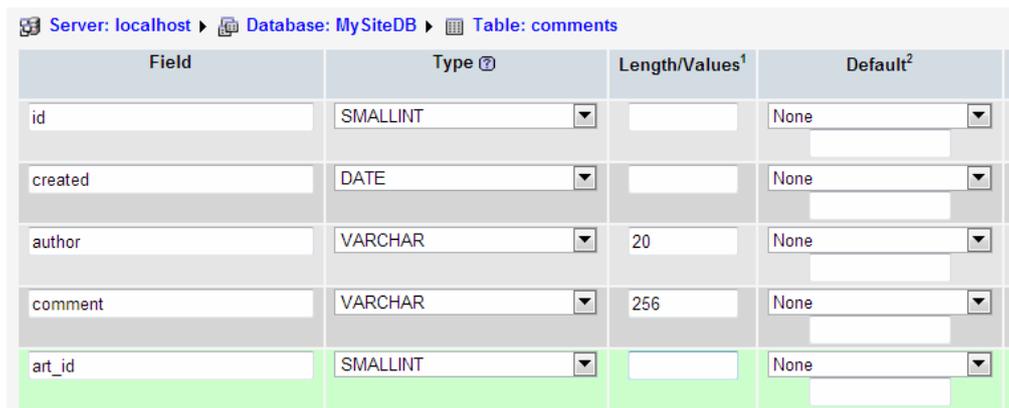


Рис. 3.3. Заполнение полей таблицы

5. Полученный результат должен выглядеть следующим образом (*рис. 3.4*):

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	id	smallint(6)			No	None	auto_increment	[Icons]
<input type="checkbox"/>	created	date			No	None		[Icons]
<input type="checkbox"/>	author	varchar(20)	cp1251_general_ci		No	None		[Icons]
<input type="checkbox"/>	comment	varchar(256)	cp1251_general_ci		No	None		[Icons]
<input type="checkbox"/>	art_id	smallint(6)			No	None		[Icons]

Рис. 3.4. Результат создания таблицы

Упражнение 5: Создание межтабличных связей

В данном упражнении необходимо создать связи между таблицами для поддержания целостности данных web-приложения.

1. Для организации межтабличных связей выберите БД MySiteDB, вкладку Designer. Откроется окно схемы данных.
2. С помощью инструментов окна Designer создайте связь «один ко многим» (рис. 3.5).

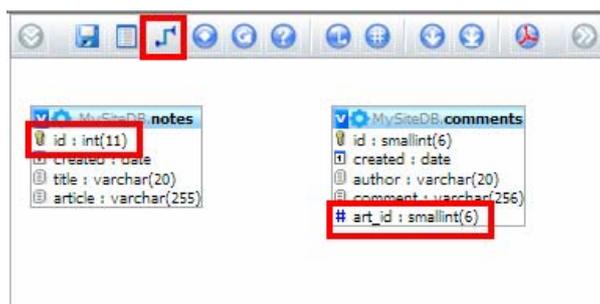


Рис. 3.5 Поле окна инструментов Designer

3. Введите в созданные таблицы несколько записей – для проверки их работы и для использования на будущих серверных страницах сайта. Для этого выберите нужную таблицу и нажмите кнопку **Insert** в группе **Action**. После заполнения соответствующих полей таблицы нажмите кнопку **Go**.

⇒ Помните, что поля **id** в таблицах заполнять не надо – они заполняются автоматически.

⇒ Поле **art_id** таблицы **comments** должно быть привязано к полю **id** таблицы **notes**, т.к. комментарии создаются только в привязке к конкретной заметке. MySQL автоматически в поле **art_id** таблицы **comments** подставляет выпадающий список **id** уже созданных заметок, вам необходимо лишь выбрать **id** заметки из выпадающего списка.

Упражнение 6: Регистрация базы данных в Adobe Dreamweaver для подключения к сайту

Регистрация включает задание имени и пароля пользователя, от имени которого ведется работа с базой (в нашем случае admin), а также адрес сервера данных (localhost) и имя базы (MySiteDB).

1. Запустите сервер и базу данных MySQL.
2. В Adobe Dreamweaver в меню **Базы данных** нажмите кнопку «плюс (+)». Появится единственный пункт – **Подключение MySQL**.
3. Далее появится диалоговое окно, которое необходимо заполнить (см. рис.3.6). Последний раздел «База данных» можно заполнить как самостоятельно, так и нажав **Выбрать**. Протестируйте соединение. Затем нажмите **ОК**.

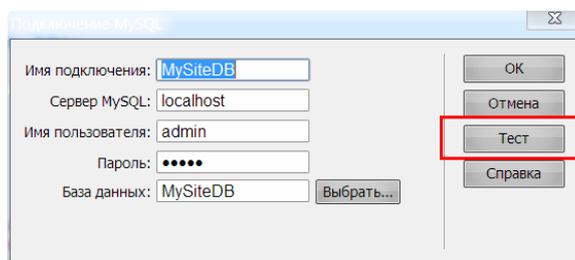


Рис. 3.6. Заполнение полей окна подключения базы данных к проекту

4. После подключения в меню **Базы данных** должна отображаться подключенная нами БД MySiteDB с двумя таблицами (рис. 3.7).

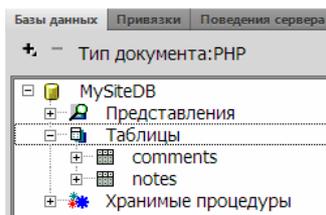


Рис. 3.7. Отображение подключенной к проекту базы данных

5. Убедитесь, что в меню **Файлы** появилась новая папка **Connections** и в ней файл **MySiteDB.php** (название файла совпадает с именем соединения).

Упражнение 7: Файл подключения базы данных

В ходе выполнения данного упражнения необходимо внести изменения в код автоматически созданного файла подключения базы данных для настройки корректной работы подключенной базы данных с кириллицей.

1. Откройте файл MySiteDB.php и внесите в него следующие изменения:
 - 1.1. Измените наименования переменных для удобства дальнейшей работы;
 - 1.2. Измените функцию *mysql_pconnect()* на *mysqli_connect()*;
 - 1.3. Внесите дополнения для корректной кодировки символов в базе данных:

```
<?php
# FileName="Connection_php_mysql.htm"
# Type="MYSQL"
# HTTP="true"
$localhost = "localhost";
$db = "MySiteDB";
$user = "admin";
$password = "admin";

$link = mysqli_connect($localhost, $user, $password) or
trigger_error(mysql_error(),E_USER_ERROR);
//trigger_error выводит на страницу сообщение об ошибке. Первый параметр
- сообщение об ошибке
//в строковом виде, в данном случае возвращается функция mysql_error(),
второй - числовой код //ошибки(почти всегда используется значение
константы E_USER_ERROR, равное 256)

//Следующие строки необходимы для того, чтобы MySQL воспринимал
кириллицу.
//Параметры функции mysqli_query(): идентификатор соединения с сервером
и запрос SQL

mysqli_query($link, "SET NAMES cp1251;") or die(mysql_error());
mysqli_query($link, "SET CHARACTER SET cp1251;") or die(mysql_error());
?>
```