**Практическая работа Разработка веб-приложения «Заметки»**

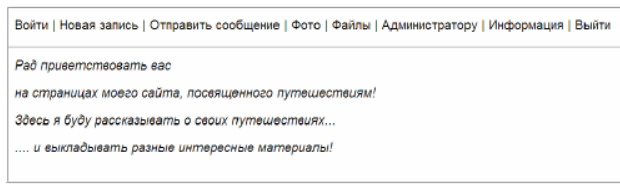
**Часть 1**

**Задание 1. Создание статической основы веб-страниц**

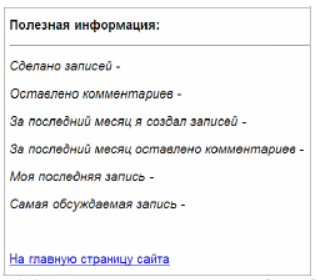
Необходимо создать две первые статические страницы проекта, которые станут основой для дальнейшей разработки.

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.

Пример страницы blog.html:



Пример страницы inform.html:

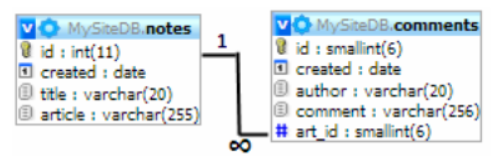


Свяжите гиперссылками созданные страницы

**Задание 2. Создание базы данных MySQL**

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:



Ход выполнения задания:

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:
   1. Сформируйте запрос на создание базы MySiteDB с использованием SQL;
   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).

**Задание 3. 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.

1. Реализуйте запрос.
2. 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.

**Задание 4 Создание таблицы notes**

Таблица notes содержит заметки автора блога. Данная таблица будет создана средствами PHP. Информацию о полях таблицы смотрите в таблице 1.

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\*:



\*использовать sql – запрос CREATE TABLE…

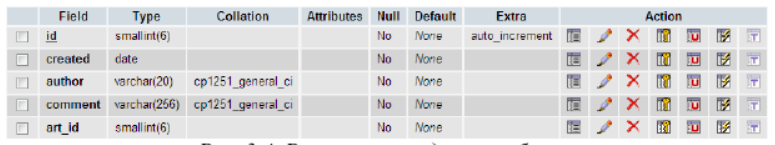
1. Реализуйте запрос на создание таблицы.
2. С помощью утилиты PhpMyAdmin убедитесь, что создана новая таблица. Для этого запустите утилиту, перейдите к базе данных MySiteDB и просмотрите ее структуру. В ней должна появиться соответствующая таблица.

**Задание 5 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.

Информация о полях таблицы:

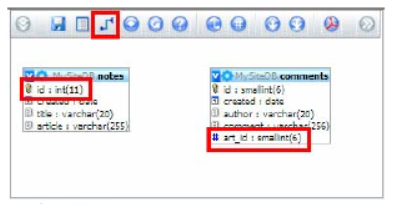
Результат:



**Задание 6 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»



1. 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.*

**Задание 7.**

Сделайте в каждой таблице не менее двух записей.

**Задание 8. Файл подключения базы данных**

1. Создайте файл MySiteDB.php и внесите в него следующий код:

<?php

$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());

?>

**Задание 9. Вывод данных из базы на страницу**

В этом упражнении на главную страницу сайта необходимо вывести все заметки из таблицы БД notes.

1. Переименуйте blog.html в blog.php (старую html-страницу после переименования можно удалить).
2. Удалите записи на странице. Должны остаться только приветствие и навигационное меню.
3. Создайте соединение с сервером. Оно у нас уже реализовано в файле MySitDB.php – файл надо просто включить с помощью функции require\_once(), в качестве параметра передав ей путь к файлу («Connections/MySiteBD.php»):

<?php require\_once ("connections/MySiteDB.php"); ?>

1. Далее необходимо вывести записи (строки) на страницу сайта из таблицы notes. Сначала надо реализовать запрос на выборку. Для этого:

4.1. выберите БД;

4.2. создайте SQL-запрос на выборку данных из таблицы (SELECT fields FROM tableName). Здесь SELECT – оператор выбора полей, FROM – оператор выбора таблицы-источника полей.

*⇒ Если вам необходимо выбрать все поля таблицы (как в данном случае), то запрос можно построить так: SELECT \* FROM tablename, где символ «\*» обозначает все поля таблицы.*

* 1. Реализуйте запрос на выборку.

1. Далее необходимо вывести записи на страницу сайта. Для этого воспользуйтесь материалом <https://metanit.com/php/mysql/3.4.php>