

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ  
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО  
ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ГОУ ВПО ВГУ)

Факультет прикладной математики, информатики и механики  
Кафедра математического обеспечения ЭВМ

Разработка информационной системы учебного заведения

**Магистерская диссертация**

по направлению 010500 Прикладная математика и информатика  
Магистерская программа Системное программирование

Допущено к защите в ГАК 25.05.2010

Зав. Кафедрой \_\_\_\_\_ к.ф.-м.н., доц. Махортов С. Д.  
(подпись)

Студент \_\_\_\_\_ Асвад Фирас М.  
(подпись)

Руководитель \_\_\_\_\_ д.т.н., проф. Астахова И. Ф.  
(подпись)

Воронеж 2010

## **АННОТАЦИЯ**

Работа посвящена разработке информационной системы учебного заведения. Результатом данной работы является база данных Oracle, хранящая данные студентов, преподавателей, учебного процесса университета. При создании web-интерфейса применялся Dreamweaver Mx 2004.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	4
1. ПОСТАНОВКА ЗАДАЧИ.....	6
2. АНАЛИЗ ЗАДАЧИ .....	7
3. СРЕДСТВА РЕАЛИЗАЦИИ .....	9
4. ТРЕБОВАНИЯ К АППАРАТНОМУ И ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ .....	11
5. ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЯ.....	12
6. РЕАЛИЗАЦИЯ .....	18
6.1. Физическая модель БД.....	18
6.2. Взаимодействие функциональных блоков.....	21
6.3. Структура проекта .....	23
6.4. Подключение к БД .....	25
6.5. Выполнение SQL запросов.....	26
6.6. Отправка данных формы скрипту .....	27
6.7. Реализация списков .....	28
6.7.1. Фильтрация .....	28
6.7.2. Сортировка по полям .....	29
6.8. Работа с сессиями .....	31
6.9. Создание диаграмм.....	32
7. ТЕСТИРОВАНИЕ .....	33
ЗАКЛЮЧЕНИЕ .....	35
ЛИТЕРАТУРА.....	36
ПРИЛОЖЕНИЯ .....	37

## ВВЕДЕНИЕ

Веками человечество накапливало знания, навыки работы, сведения об окружающем мире, другими словами – собирало информацию. Вначале информация передавалась из поколения в поколение в виде преданий и устных рассказов. Возникновение и развитие книжного дела позволило передавать и хранить информацию в более надежном письменном виде. Открытия в области электричества привели к появлению телеграфа, телефона, радио, телевидения – средств, позволяющих оперативно передавать и накапливать информацию. Развитие прогресса обусловило резкий рост информации, в связи с чем, вопрос о ее сохранении и переработке становился год от года острее. С появлением вычислительной техники значительно упростились способы хранения, а главное, обработки информации. Сведения стали храниться в совокупности связанных файлов называемой базой данных, а для управления этими файлами создаются специальные программы – системы управления базами данных. С появлением архитектуры клиент-сервер доступ к базам данных стал осуществляться с помощью клиентских приложений. Многие из них были весьма требовательными к программным и аппаратным ресурсам клиента. Снизить нагрузку на клиентский компьютер помогла трёхзвенная архитектура и web-серверы. Теперь стало возможным работать с базой данных, имея на клиенте лишь обычный web-браузер. При этом упростился процесс доступа к серверу для пользователя, так как он не требует получения программы и ее настройки. Эти задачи перекладываются на сторону web-сервера и его модулей.

В настоящее время использование интернет-технологий стало неотъемлемой частью автоматизации во многих компаниях. Бурное развитие этих технологий заставляет по-новому взглянуть на разработку web-приложений, почувствовать их новые достоинства.

Целью данной работы является создание базы данных и интерактивного web-интерфейса к ней для автоматизации учебного заведения. Основной упор делается на разработку мощного web-приложения с применением некоторых последних достижений в области программирования для интернет.

## 1. ПОСТАНОВКА ЗАДАЧИ

С целью автоматизации работы учебного заведения требуется разработать базу данных и web-интерфейс к ней.

База данных должна хранить информацию о следующих объектах:

- факультеты;
- кафедры;
- преподаватели;
- студенты;
- экзамены;
- предметы.

Web-интерфейс должен обеспечивать выполнение следующих действий:

- добавление и редактирование основных данных;
- отображение статистических данных по учебному заведению;
- авторизация и многоуровневое разделение пользователей.

Помимо этих условий в качестве СУБД требуется использовать систему Mysql.

## 2. АНАЛИЗ ЗАДАЧИ

В первой части анализа рассматривается задача создания базы данных. Ее проектирование стоит начать с основных сущностей: «**факультет**» и «**кафедры**». Сущность «**факультет**» будет иметь свойства «название», «код факультета» и «количество студентов и преподавателей». «**Кафедры**» будут характеризоваться свойствами «название», «код кафедры» и «количество студентов и преподавателей». Эти сущности будут играть чисто информативную роль. «**Преподаватели**» будут характеризоваться следующими свойствами: «код преподавателя», «фамилия, имя», «код его факультета», «код его кафедры» и «адрес». «**Студенты**» будут характеризоваться следующими свойствами: «код студента», «фамилия, имя», «код его факультета», «код его кафедры» и «адрес», «курс» и «год поступления». «**Предмет**» будет характеризоваться следующими свойствами: «код предмета», «название предмета», «код факультета», «код кафедры», «семестр», «количество часов» и «курс». «**Экзамены**» будут характеризоваться следующими свойствами: «код экзамена», «имя студента, который сдает экзамен», «имя преподавателя, который принимает экзамен», «дата сдачи», «код предмета», «код факультета», «код кафедры», «оценка по экзамену», «курс».

Что касается web-приложения, то для удобства разработки и поддержки оно будет иметь модульную структуру. Главный файл приложения «index» в зависимости от GET-запроса будет подключать нужный модуль (страницу), который будет храниться в отдельной папке и будет иметь свой файл «index». Помимо этого файла в папке будут также содержаться подключаемые файлы с константами и функциями модуля, файлы, содержащие HTML-формы и др. Модули «статистика» и «пользователи» будут разрабатываться по тем же принципам что и другие модули. Отличия будут только в том, что для модуля «статистика»

потребуется лишь отображение данных с помощью диаграмм и списков, база данных здесь изменяться не будет. Для регистрации же пользователей, работающих с приложением, необходимо будет использовать сессии.

Большая часть времени будет потрачена на изучение новейших интернет-технологий и применение этих технологий на страницах приложения. Здесь могут возникнуть проблемы, связанные с использованием разных браузеров, а так же с передачей данных между клиентом и сервером приложений. Исходя из этого, потребуется ограничить набор поддерживаемых браузеров, а также настроить новый надежный способ передачи данных.



### 3. СРЕДСТВА РЕАЛИЗАЦИИ

При разработке программного продукта использовались ниже перечисленные средства реализации.

Mysql была выбрана для управления базой данных.

Она выбрана как очень удобная и функциональная среда разработки.

Для проектирования базы данных использовалась программа Power Designer 15.0.

В качестве сервера приложений был выбран web-сервер Apache 1.3.31. Важными преимуществами Apache перед другими web-серверами являются свободное распространение данного продукта, стабильность, высокая производительность, открытый исходный код. При наличии соответствующих аппаратных средств и настройке Apache может выдерживать самые высокие загрузки: Yahoo, Altavista, GeoCities, Hotmail основаны на различных версиях этого сервера.

Все серверные скрипты web-приложения написаны на языке PHP 5.0.3. PHP – простое и мощное средство для динамического построения страниц HTML на сервере. Одним из главных достоинств PHP является тот факт, что он внедряется прямо в HTML-код, поэтому программисту не приходится писать программу с множеством команд для простого вывода HTML. Благодаря наличию сотен стандартных функций PHP в состоянии решить практически любую задачу. В нем имеется обширная поддержка создания графики и операций с ней, математических вычислений, средств электронной коммерции, технологий XML, ODBC, а также работы со многими известными СУБД, в том числе и Mysql. Широкий выбор возможностей избавляет от необходимости рутинной и непростой работы по подключению сторонних модулей. Всё это делает его хорошим кандидатом для разработки приложений на стороне сервера.

В качестве клиентских приложений выступают обычные web-браузеры. Из всего множества браузеров были выбраны три наиболее популярных и надежных: Internet Explorer 6.0, Opera 9.0.2, Google Chrome.

Вся функциональность клиентских приложений реализована с помощью языков HTML и JavaScript.

Для написания кода клиентской и серверной сторон web-приложения использовалась среда разработки Macromedia Dreamweaver MX2004.

#### **4. ТРЕБОВАНИЯ К АППАРАТНОМУ И ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ**

Аппаратное обеспечение сервера данных: размеры свободного дискового пространства, необходимого для хранения базы данных зависят от количества данных и от версии Mysql, но не могут составлять менее 80 Мб.

Аппаратное обеспечение web-сервера: не менее 4Мб свободного дискового пространства для хранения файлов приложения.

Аппаратное обеспечение клиента: полностью определяется выбранным web-браузером.

Программное обеспечение сервера данных: Mysql и выше.

Программное обеспечение web-сервера: Apache версии 1.3 и выше; PHP версии 5 и выше; наличие подключённых библиотек php\_oci8.dll, php\_iconv.dll, php\_gd2.dll.

Программное обеспечение клиента: браузеры Internet Explorer, Opera или Mozilla Firefox с поддержкой объекта XMLHttpRequest.

## 5. ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЯ

Прежде чем начать работу с приложением необходимо пройти авторизацию (рис.1).



The screenshot shows a login form titled "Edit Data Base" on a teal background. The form consists of two input fields: "Enter Your Name" and "Enter Your Password". Below the fields is a "Go" button.

Enter Your Name	<input type="text"/>
Enter Your Password	<input type="password"/>

Go

Рис. 1. Форма авторизации

После прохождения авторизации пользователю будут показаны те пункты меню (рис. 2), доступ к которым ему разрешен.



Рис. 2. Меню

Полное меню пользователя имеет приведенную ниже структуру.

- **ФАКУЛЬТЕТ**
  - Добавить/удалить
  - Просмотреть/Редактировать
- **КАФЕДРА**
  - Добавить/удалить
  - Просмотреть/Редактировать
- **ПРЕПОДАВАТЕЛИ**
  - Добавить/удалить
  - Просмотреть/Редактировать
- **СТУДЕНТЫ**
  - Добавить/удалить
  - Просмотреть/Редактировать
- **ПРЕДМЕТЫ**
  - Добавить/удалить
  - Просмотреть/Редактировать
- **ЭКЗАМЕНЫ**
  - Открыть
  - Закрывать
  - Просмотреть
- **ВЫХОД**

Теперь пользователь может приступить к выполнению своей задачи. Каждая страница приложения снабжена указаниями (рис. 3), которыми пользователь может руководствоваться.

Чтобы добавить студента необходимо
<ul style="list-style-type: none"> <li>• Указать фамилию</li> </ul>
<ul style="list-style-type: none"> <li>• Выбрать кафедру</li> </ul>
<ul style="list-style-type: none"> <li>• Указать его адрес</li> </ul>
<ul style="list-style-type: none"> <li>• Указать курс</li> </ul>
<ul style="list-style-type: none"> <li>• Указать год поступления</li> </ul>
<ul style="list-style-type: none"> <li>• Добавить фотографию</li> </ul>

Рис. 3. Указания по добавлению студента

В приложении используется много различных форм. Все эти формы можно разделить на следующие группы: для добавления (рис. 3), для изменения (рис. 4), для навигации по спискам, для фильтрации списков и для выборки данных.


Рис. 4. Форма изменения параметров студента.

Рис. 5. Форма добавления преподавателя

Однако перед тем как редактировать данные, надо сначала найти то, что нужно отредактировать. Для этой цели и служит поиск преподавателя или студента (рис. 6).

Рис. 6. Поиск преподавателя или студента

Результатом поиска будет список студентов или преподавателей, которых необходимо было найти. (рис. 7).

<b>Search Results Of Students</b>						
<b>Id</b>	<b>Image</b>	<b>Name</b>	<b>Adress</b>	<b>College</b>	<b>Deprament</b>	<b>Class</b>
62		Firas aswad M.	diyala	computer	Met	5

### Search Results Of Students

<b>Id</b>	<b>Name</b>	<b>Adress</b>	<b>College</b>	<b>Department</b>	<b>Class</b>
1	Ahmae	dyala	computer	Faz	5
4	Ali Mohammed	Diala	computer	Russia	5
7	Ali Aswad	Voronezh	computer	Russia	5
8	Ade	Diyala	computer	Met	4
9	Adel	Bag	computer	Faz	5
16	Ali Firas	Diyala	computer	Met	5
40	Agfhj	Moscow	computer	Met	1
42	Ali Firas ff	Voronezh	computer	Met	1

Рис. 7. Форма фильтрации прокатов

Общую численность студентов и преподавателей можно отобразить следующей диаграммой (рис. 8).

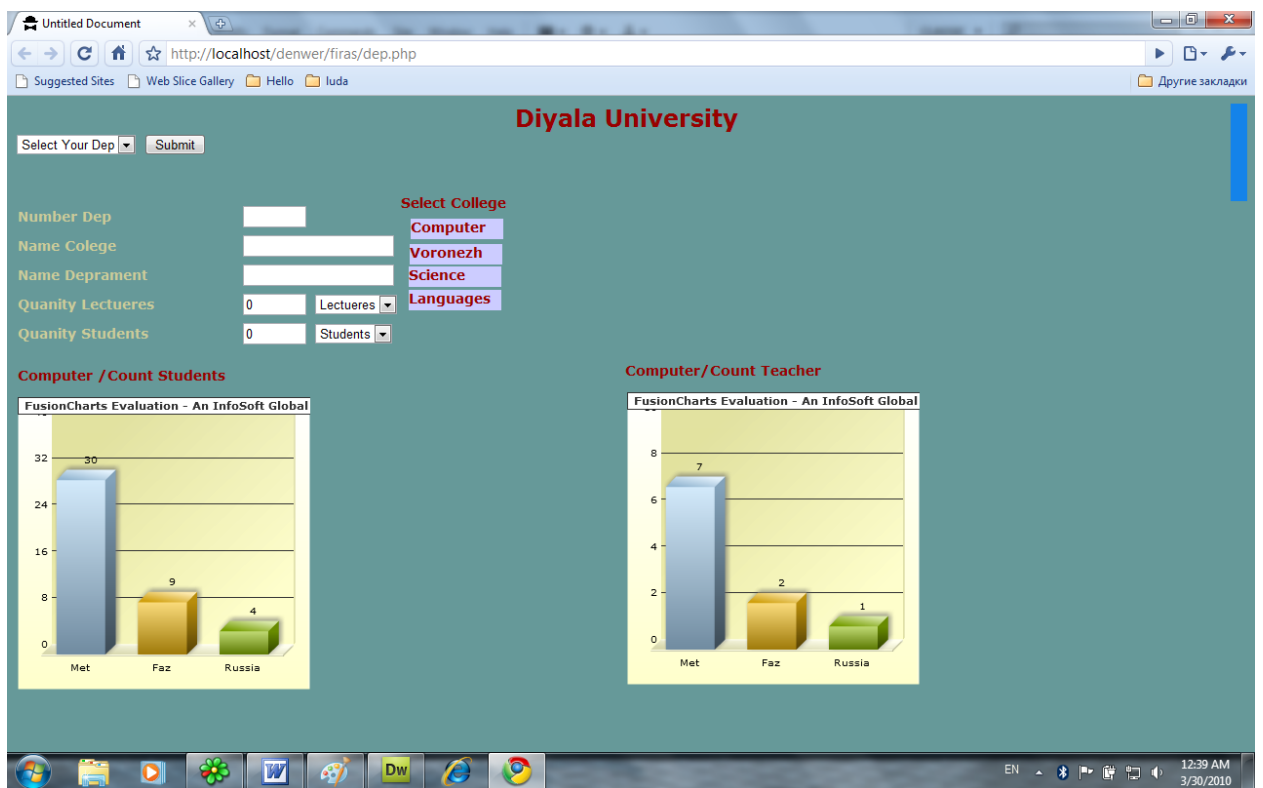


Рис. 8. Численность студентов и преподавателей



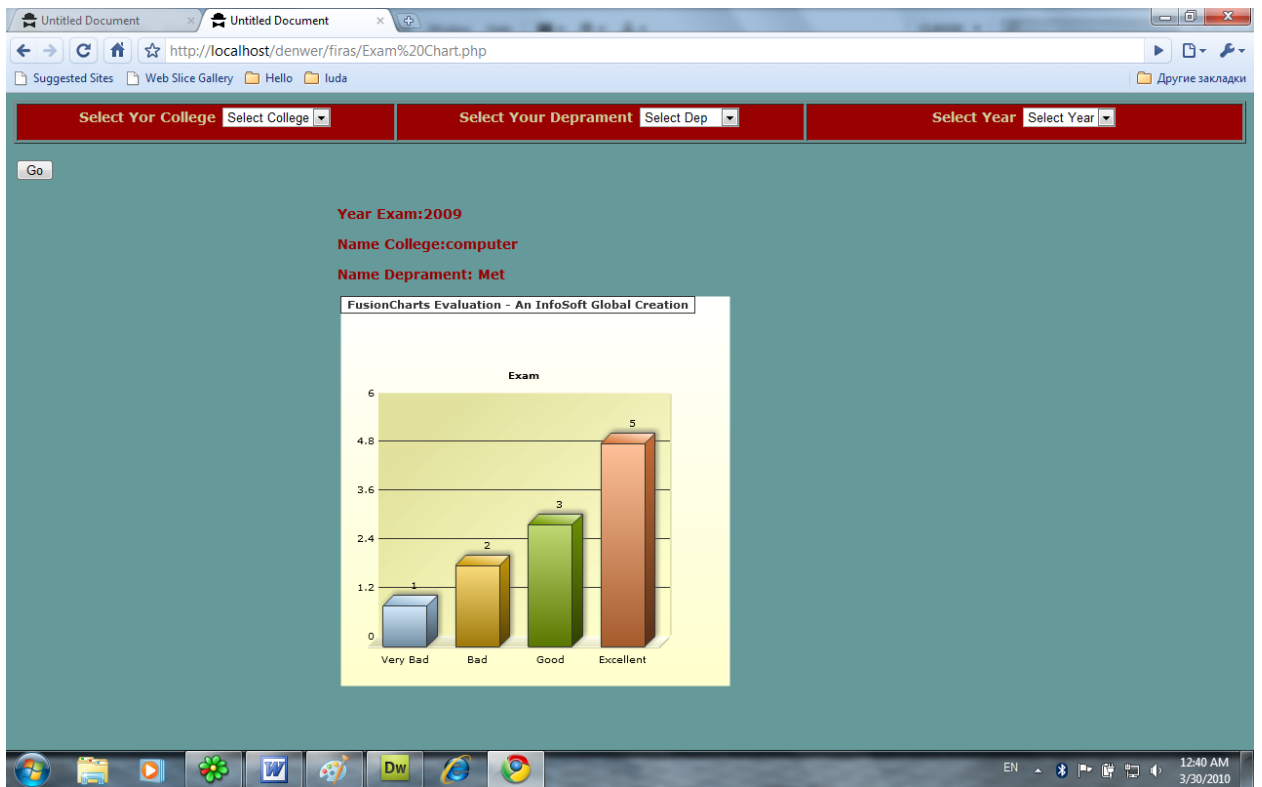


Рис. 9. Результаты сдачи экзаменов

## 6. РЕАЛИЗАЦИЯ

### 6.1. Физическая модель БД

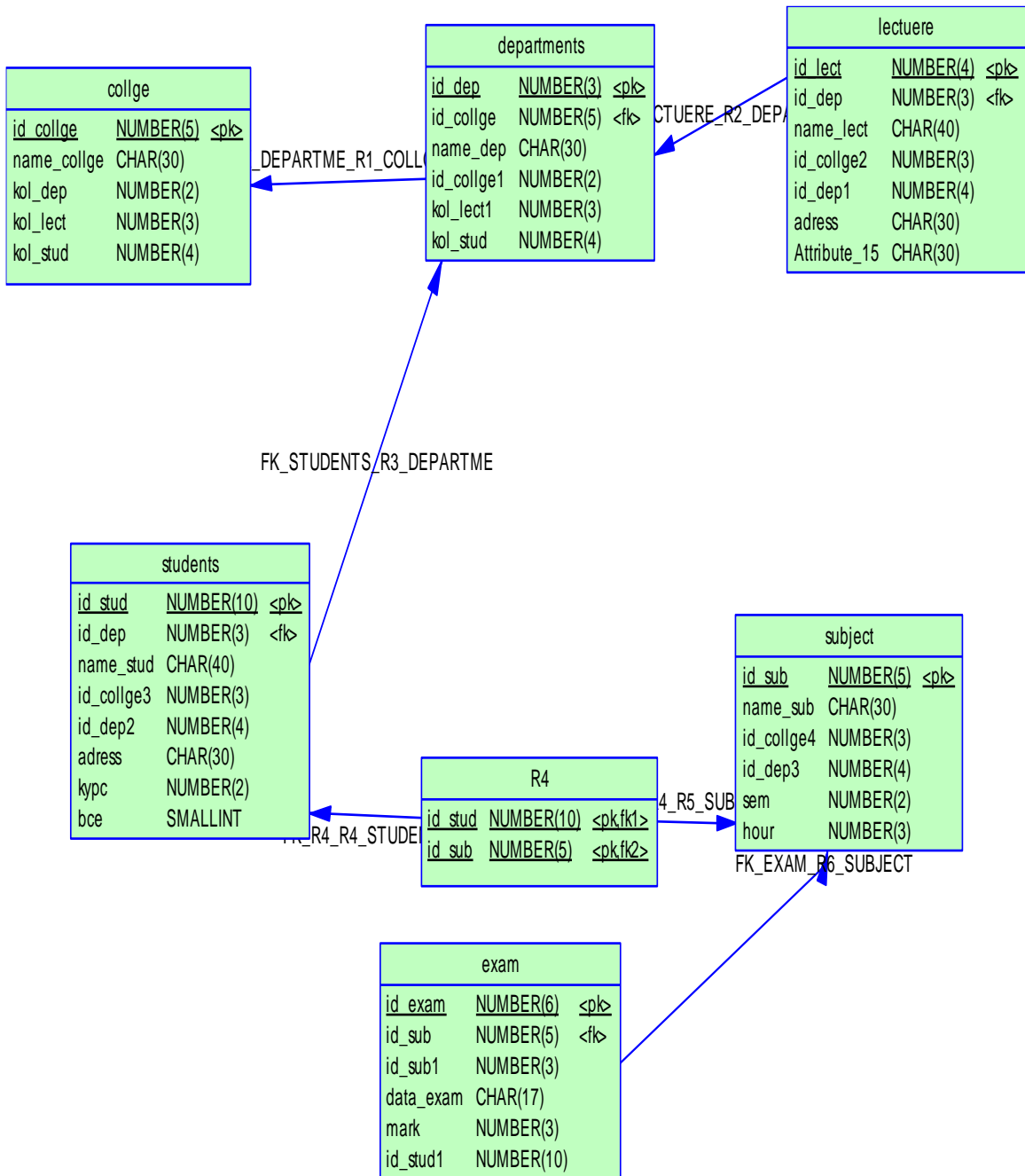


Рис. 10. Физическая модель БД

В этом разделе описана физическая модель базы данных (рис. 11).

Таблица **COLLEGE** содержит данные о факультете:

- ID\_COLLEGE – идентификатор;
- NAME\_COLLEGE – название факультета;
- KOL\_DEP – количество факультетов;
- KOL\_LEC – количество преподавателей;
- KOL\_STUD – количество студентов.

Таблица **DEPARTMENT** содержит данные о кафедрах:

- ID\_DEP – идентификатор;
- NAME\_DEP – название кафедры;
- ID\_COLLEGE1 – идентификатор факультета;
- KOL\_LEC – количество преподавателей;
- KOL\_STUD – количество студентов.

Таблица **LECTURER** содержит данные о преподавателях:

- ID\_LECT – идентификатор;
- NAME\_LECT – ФИО преподавателя;
- ID\_COLLEGE2 – идентификатор факультета;
- ID\_DEP1 – идентификатор кафедры;
- ADDRESS – адрес;

Таблица **STUDENT** содержит данные о студентах:

- ID\_STUD – идентификатор;
- NAME\_STUD – ФИО студента;
- ID\_COLLEGE3 – идентификатор факультета;
- ID\_DEP2 – идентификатор кафедры;
- ADDRESS – адрес;

- COURSE – курс, на котором обучается студент;
- YEAR – год поступления;
- IMAGE – фото студента.

Таблица **SUBJECT** содержит данные о предметах:

- ID\_SUB – идентификатор;
- NAME\_SUB – название предмета;
- ID\_COLLEGE4 – идентификатор факультета;
- ID\_DEP3 – идентификатор кафедры;
- SEM – семестр;
- HOUR – время проведения экзамена.

Таблица **EXAM** содержит данные об экзаменах:

- ID\_EXAM – идентификатор;
- ID\_SUB1 – идентификатор предмета;
- DATA\_EXAM – дата проведения экзамена;
- MARK – оценка;
- ID\_STUD1 – идентификатор студента;

## 6.2. Взаимодействие функциональных блоков

Схема взаимодействия функциональных блоков приложения довольно проста. Данное web-приложение состоит из нескольких модулей, работа с каждым из которых происходит автономно без привлечения других.

Для модулей «факультет», «кафедры», «преподаватели», «студенты», «экзамены», «предметы». Схема взаимодействия функциональных блоков имеет вид, представленный на рис. 11.

Для модуля «студенты» схема имеет вид, изображенный на рис. 11.



Рис. 11. Схема взаимодействия функциональных блоков

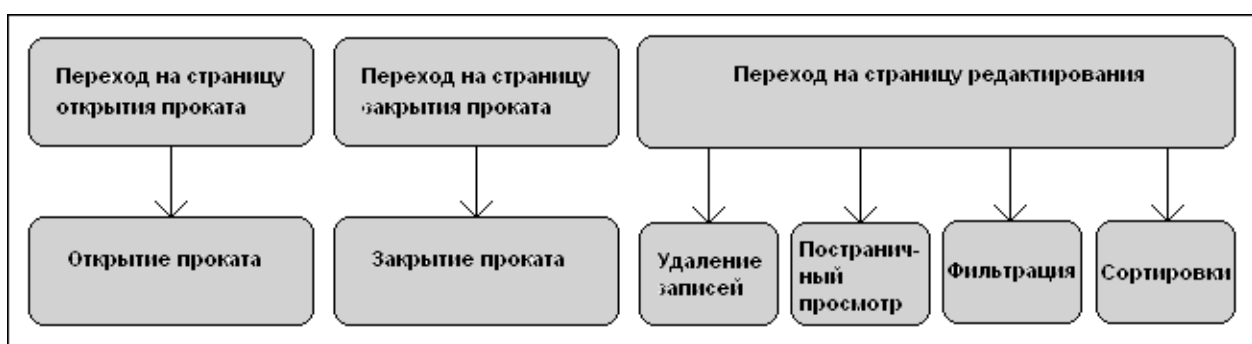


Рис. 12. Схема взаимодействия функциональных блоков модуля «студенты»

Для модулей же «преподаватели» и «экзамены» схемы взаимодействия функциональных блоков имеют вид, изображенный на рис. 13 и рис. 14 соответственно.



Рис. 13. Схема взаимодействия функциональных блоков модуля «факультет»



Рис. 14. Схема взаимодействия функциональных блоков модуля «кафедры»

### 6.3. Структура проекта

На рис. 15 приведена файловая структура проекта. Здесь более подробно рассказывается о том, как и из чего строятся интернет страницы, как вообще функционирует web-приложение.

File Name	Type	Size	Language	Date	Time
Site - Firas (Z:\denwer\www\den...	Folder			4/2/2010	6:19 PM
Connections	Folder			3/30/2010	12:31 ...
connect.php	1KB PHP Script			3/30/2010	12:31 ...
image	Folder			3/10/2010	6:50 AM
includes	Folder			3/8/2010	12:50 AM
FusionCharts	Folder			3/8/2010	12:50 AM
charts	Folder			3/8/2010	12:50 AM
Drafts	Folder			2/26/2010	2:19 PM
dynamic	Folder			3/8/2010	12:50 AM
data	Folder			3/8/2010	12:50 AM
js	Folder			3/8/2010	12:50 AM
DWFChart.inc.php	36KB PHP Script			3/5/2010	12:57 PM
FusionCharts.php	8KB PHP Script			3/5/2010	12:57 PM
config.cfg	1KB Microsoft...			3/3/2010	3:09 AM
Chart Student.php	4KB PHP Script			3/3/2010	2:22 AM
college1.php	14KB PHP Script			2/6/2010	6:44 PM
css.css	1KB Cascadin...			7/9/2009	3:37 PM
dae.php	5KB PHP Script			3/8/2010	11:48 AM
dataedit.php	11KB PHP Script			7/13/2009	6:20 AM
dep.php	18KB PHP Script			3/30/2010	12:38 ...
editdatabase.php	3KB PHP Script			1/27/2010	7:23 PM
Exam Chart.php	9KB PHP Script			3/10/2010	7:55 AM
exam.php	17KB PHP Script			3/5/2010	1:47 PM
examedit.php	17KB PHP Script			1/30/2010	2:00 AM
Firas.jpg	15KB JPEG image			3/9/2010	3:46 AM
index.html	2KB HTML Do...			2/11/2010	6:53 PM
lectueresall.php	12KB PHP Script			9/15/2009	5:22 PM
lectueresedit.php	13KB PHP Script			3/9/2010	2:31 AM
left.html	2KB HTML Do...			2/26/2010	8:19 PM
main.php	1KB PHP Script			2/26/2010	8:02 PM
right.php	2KB PHP Script			11/22/2009	2:53 PM
searchlectmain.php	4KB PHP Script			9/18/2009	1:19 PM
searchlectmain1.php	4KB PHP Script			3/9/2010	1:26 AM
searchstudent.php	3KB PHP Script			9/6/2009	2:17 AM
searchstudentmain.php	5KB PHP Script			3/9/2010	3:47 AM
searchstudentmain1.php	4KB PHP Script			9/18/2009	1:21 PM
studentall.php	14KB PHP Script			1/30/2010	2:34 AM
studentedit.php	16KB PHP Script			3/9/2010	2:31 AM
subjectall.php	16KB PHP Script			2/5/2010	4:04 PM
testfiras.php	14KB PHP Script			1/3/2010	2:20 PM
top.html	2KB HTML Do...			2/26/2010	8:01 PM
updateexam.php	10KB PHP Script			1/30/2010	2:17 AM
updatestuden1.php	7KB PHP Script			1/30/2010	9:34 PM
updatelect1.php	10KB PHP Script			9/5/2009	4:39 AM
user.php	5KB PHP Script			9/18/2009	1:10 PM

Рис. 15. Структура проекта

Папка «Connections» означает связь с базой данных, в папке «image» - рисунки для страниц. Папка «includes» содержит данные о диаграммах. Файл «Chart Student.php» показывает нам диаграмму о студентах. Файл «college1.php» содержит информацию о всех факультетах университета. Файл «dep.php» содержит информацию о всех кафедрах университета. Файл «Exam Chart.php» показывает нам диаграммы о студентах. Файл «exam edit.php» позволяет вносить данные по изменению, добавлению или удалению экзаменов. Файл «index.php» это главная страница программы. Когда пользователь набирает адрес сайта, выполняется файл из корневой директории «index.php», который из папки «/includes» подключает файлы «left.php» и «right.php», содержащие определения общих констант. Далее подключаются шапка и меню сайта из файлов «top.php» и «main.php». Файл «lectuerelsall.php» показывает информацию о преподавателях. Файл «lecrueledit.php» позволяет нам вносить изменения в данные о преподавателях, а также добавлять и удалять. Файлы «searchlectmain.php» и «serchstudentmain.» служат для поиска преподавателей и студентов соответственно. Файд «studentall.php» показывает информацию о всех студентах, обучающихся в университете. Файл «studedit.php» позволяет вносить изменения в данные о студентах, а также добавлять и удалять. Файл «suijectall.php» показывает информацию о всех предметах. Файл «user.php» включают в себя логин и пароль, при вводе которых пользователь сможет изменить, удалить или добавить интересующую его информацию. Каждый факультет имеет свой логин и пароль.



## 6.4. Подключение к БД

Подключение к базе данных с именем *DATABASE* пользователя с логином *LOGIN* и паролем *PASSWORD* осуществляется с помощью следующей функции:

```
function mysql_connect() {
    if (!( $\$connection = mysql\_connect(LOGIN,$ 
    PASSWORD, DATABASE))) {
         $\$err = mysql\_error();$ 
        echo "Error! Login and password are not
        correct
        mysql.". $\$err['message'];$ 
        return false;
    } else return  $\$connection;$ 
}
```

Функция *mysql\_connect* устанавливает соединение с сервером Denwer . Функция *mysql\_error* возвращает массив с информацией о последней ошибке. Элемент *message* этого массива возвращает строку с текстом ошибки.

## 6.5. Выполнение SQL запросов

Выполнение SQL запроса осуществляется с помощью следующих строк:

```
$stmt = mysql_parse($connection, $sql);
$exec = (mysql_execute($stmt) or die(mysql_ERR));
```

Здесь переменная *\$sql* содержит строку-запрос, *\$connection* – идентификатор подключения к базе данных. Функция *oci\_parse* подготавливает запрос к выполнению, а функция *oci\_execute* выполняет запрос.

Если требуется вывод результата SELECT запроса, возможно использование функции *oci\_fetch\_assoc*, которая возвращает следующую строку из результата запроса в виде ассоциативного массива:

```
while ($row = mysql_fetch_assoc($stmt)) {
    $str1 = $row["FIELD_1"];
    $str2 = $row["FIELD_2"];
}
```

В проекте также используется функция *mysql\_fetch\_all*, которая выбирает все строки из результата запроса в массив и возвращает количество выбранных строк:

```
$num_rows = mysql_fetch_all($stmt, $result);
foreach ($result['FIELD_1'] as $f1) {.....}
```

## 6.6. Отправка данных формы скрипту

Для того, чтобы отправить данные формы скрипту *script\_name.php*, необходимо для тега *FORM* указать параметры *action="script\_name.php"*, *method = "POST"* или *method = "GET"*, а для каждого объекта ввода указать уникальное имя. Форма должна также иметь кнопку с параметром *type="submit"*. Тогда при нажатии на кнопку выполнится указанный скрипт:

```
<form action="script_name.php" method = "POST">
<input type="text" name="name1" value="val1">
<input type="checkbox" name="name2" value="val2">
<select name="name3">
    <option value="val1">text1</option>
    <option value="val2">text2</option>
    .....
</select>
<input type="submit" value="Выполнить">
</form>
```

Чтобы скрипт «видел» данные полей ввода, к ним необходимо обращение вида  $\$_POST["name1"]$ , если используется метод POST, или  $\$_GET["name1"]$ , если используется GET.

## 6.7. Реализация списков

### 6.7.1. Фильтрация

Фильтры представляют собой элементы ввода и выпадающие списки. Когда пользователь заполняет фильтры и нажимает кнопку “submit”, данные посылаются PHP-скрипту методом GET. Скрипт работает примерно таким образом:

```

if (isset($_GET['namestudent']) &&
    ($_GET['namestudent'] != 0)) {
    $wmask_student = " AND ID_STUDENT =
''".$_GET['namestudent']."' ";
} else {
    $wmask_student = "";
}
if (isset($_GET['record']) &&
    ($_GET['record'] != 0)) {
    $wmask_record = " AND ID_REC = ''".$_GET['record']."' ";
} else {
    $wmask_record = "";
}
...
$big_mask = " WHERE (1=1) ".$wmask_student.$wmask_record;
$sql = 'SELECT ... FROM ... '.$big_mask;

```

Таким образом, скрипт проверяет, определены ли значения фильтров, и если определены, то SQL запрос для выборки данных пополняется еще несколькими условиями.

### 6.7.2. Сортировка по полям

Еще один способ сделать работу со списками более удобной - наделить их возможностью сортировки по полям. Щелкнув по названию поля, пользователь упорядочит все записи списка по возрастанию значений этого поля. После повторного щелчка данные будут упорядочены по убыванию и так далее. Секрет работы этого инструмента также состоит в приписывании некоторого условия к исходному SQL-запросу. Скрипт работает похожим образом:

```

$id_link = 'asc';
$name_link = 'asc';
switch ($_GET['sortby']) {
    case 'name':
        $sortby = 'FNAME';
        if($_GET['order']=='desc') {
            $order='DESC'; $name_link='asc';
        } else {
            $order='ASC'; $name_link='desc';
        }
        break;
    default:
        $sortby = 'ID_FILM';
        if($_GET['order']=='desc') {
            $order='DESC'; $id_link='asc';
        } else {
            $order='ASC'; $id_link='desc';
        }
        break;
}
...
$sql = 'SELECT ... FROM ... ORDER BY `.$sortby.` `.$order;

```

Если производится щелчок по названию поля “имя студента” в первый раз, то элемента `$_GET['order']` еще не существует, и строка `$order='ASC'` задает сортировку по возрастанию. В это же время новая ссылка формируется с дополнением “`&order=desc`”. При щелчке по этой ссылке записи упорядочиваются уже по убыванию значений поля “имя студента”. Как видно из кода, по умолчанию записи сортируются по идентификатору.

## 6.8. Работа с сессиями

Сессией называется период времени, который начинается с момента прихода пользователя на сайт и завершается, когда пользователь покидает его. В течение сессии часто возникает необходимость в сохранении различных переменных, которые бы «сопровождали» пользователя при перемещениях на сайте, чтобы не приходилось вручную кодировать и присоединять к URL многочисленные скрытые поля и переменные. Одним из самых распространенных применений сессий является хранение логина и пароля зарегистрированного пользователя. Для работы с сессиями в PHP предусмотрен ряд функций. Так, например, перед созданием новой сессии с помощью функции *session\_start*, можно задать ее имя с использованием функции *session\_name*. После этого в глобальном массиве *\$\_SESSION* можно зарегистрировать любую переменную. Делается это следующим образом:

```
$_SESSION['param_name'] = $param_value;
```

Теперь эта переменная будет доступна всегда, пока не уничтожена сессия, и пока ее не удалили вручную, например, так:

```
unset($_SESSION['param_name']);
```

Все переменные текущей сессии можно удалить с помощью функции *session\_unset* или, уничтожив саму сессию при помощи функции *session\_destroy*.

## 6.9. Создание диаграмм

Прорисовка диаграмм в приложении производится с помощью библиотеки `php_gd2.dll`, которая входит в состав дистрибутива PHP 5.0.3. Код функции, рисующей диаграмму приведен в приложении 3. Итак, чтобы начать рисовать, необходимо сначала создать переменную изображения с помощью функции `imagecreate`. Затем необходимо нарисовать фон диаграммы. Функция `imageline` используется для рисования линий, а функции `imagefilledpolygon` и `imagefilledrectangle` – для рисования закрашенного многоугольника и прямоугольника соответственно, при этом цвет задается с помощью функции `imagecolorallocate`. Подписи по осям выводятся функцией `imagefttext`. Затем прорисовываются столбики диаграммы. Теперь чтобы изображение сохранилось в формате GIF необходимо послать следующий заголовок вида «Content-Type: image/gif» с помощью функции `header`. После этого изображение можно сохранить, используя функцию `imagegif`, а переменную изображения можно удалить с помощью функции `imagedestroy`.



## 7. ТЕСТИРОВАНИЕ

План тестирования web-приложения имеет структуру, показанную ниже.

1. Тестирование аппарата авторизации:
  - 1.1 проверка доступа зарегистрированных пользователей к элементам приложения;
  - 1.2 проверка доступа незарегистрированных пользователей к элементам приложения.
2. Тестирование модуля «факультет»:
  - 2.1 проверка обработки некорректных данных для формы добавления;
  - 2.2 проверка обработки некорректных данных для формы редактирования;
  - 2.3 проверка добавления данных при корректных данных;
  - 2.4 проверка редактирования данных при корректных данных;
  - 2.5 проверка фильтров списка;
  - 2.6 проверка сортировок списка по некоторым полям;
  - 2.7 проверка удаления записей из списка;
  - 2.8 проверка правильности динамического построения выпадающих списков.
3. Тестирование модуля «кафедры» (аналогично тестированию модуля «факультет»).
4. Тестирование модуля «преподаватели»:
  - 4.1 проверка обработки некорректных данных для формы открытия;
  - 4.2 проверка обработки некорректных данных для формы закрытия;
  - 4.3 проверка открытия при корректных данных;
  - 4.4 проверка закрытия при корректных данных;

- 4.5 проверка фильтров списка;
  - 4.6 проверка сортировок списка по некоторым полям;
  - 4.7 проверка удаления записей из списка;
  - 4.8 проверка правильности расчетов залога и ежедневных выплат;
  - 4.9 проверка правильности динамически подгружаемых данных.
5. Тестирование модуля «студенты» (аналогично тестированию модуля «преподаватели»).
6. Тестирование модуля «предметы»:
- 6.1 проверка обработки некорректных данных для всех форм;
  - 6.2 проверка правильности подсчетов на корректных данных;
7. Тестирование модуля «экзамены» (аналогично тестированию модуля «предметы»).

## **ЗАКЛЮЧЕНИЕ**

Итак, спроектирован и реализован проект по разработке информационной системы учебного заведения. База данных позволяет хранить данные о факультетах, кафедрах, преподавателях, студентах, предметах и экзаменах. Кроме того в базе данных хранятся некоторые параметры web-приложения, а также данные о пользователях и их уровнях доступа к страницам приложения – это пакет процедур и функций, написанный на языке PL/SQL. Для доступа к базе данных разработано web-приложение, позволяющее считывать и модифицировать основные данные. Имеется возможность разделения пользователей по нескольким уровням доступа к страницам web-приложения, а также возможность просмотра и возможность изменения некоторых параметров приложения. Программный продукт имеет удобный интерактивный интерфейс, описания возможных действий пользователя, простое меню, удобные списки данных с возможностью постраничного просмотра, фильтрации и сортировок по полям.

## ЛИТЕРАТУРА

1. Астахова И.Ф. SQL в примерах и задачах/ И.Ф.Астахова, А.П. Толстобров, В.М.Мельников. – Минск: Новое знание, 2002. – 174 с.
2. Аткинсон Л., Сураски З. PHP 5. Библиотека профессионала: - М.: Вильямс, 2005. – 944 с.
3. Бегус Ю., Габзовский В., Кравчук В. Введение в PL./SQL: - М.: ДиаСофт, 1998. – 400 с.
4. Вагнер Р., Вайк А. JavaScript. Энциклопедия пользователя: - М.: ДиаСофт, 2001. – 464 с.
5. Дейт К.Дж. Введение в системы баз данных. – 7-е изд. – М.: Вильямс, 2001. – 1072 с.: илл.
6. Коггзолл Д. PHP 5. Полное руководство: - М.: Вильямс, 2006. – 752 с.
7. Конолли Т., Бегг К., Страчан А. Базы данных: проектирование, реализация и сопровождение. Теория и практика. – 2-е изд. – М.: Вильямс, 2000. – 1120 с.: илл.
8. Олищук А. В., Чаплыгин А. Н. Разработка WEB-приложений на PHP 5. Профессиональная работа: - М.: Вильямс, 2006. – 352 с.
9. Флэнаган Д. JavaScript. Подробное руководство, 4-е издание: - М.: Символ-Плюс, 2004. – 960 с.
10. Форта Б. Освой самостоятельно регулярные выражения. 10 минут на урок: - М.: Вильямс, 2005. – 184 с.
11. Шлосснейгл Д. Профессиональное программирование на PHP: - М.: Вильямс, 2006. – 624 с.

## Приложения

Создание таблиц и полей базы данных

```
CREATE TABLE `students` (
`id_stud` INT( 30 ) NOT NULL AUTO_INCREMENT ,
`name_stud` VARCHAR( 50 ) NOT NULL ,
`adres` VARCHAR( 50 ) NOT NULL ,
`id_colleg` VARCHAR( 5 ) NOT NULL ,
`id_dep` VARCHAR( 5 ) NOT NULL ,
`class` INT( 2 ) NOT NULL ,
`Data Year` DATE NOT NULL ,
`image_stud` BLOB NOT NULL ,
PRIMARY KEY ( `id_stud` )
);
```

### **connect.php**

```
<?php
# FileName="Connection_php_mysql.htm"
# Type="MYSQL"
# HTTP="true"
$hostname_connect = "localhost";
$databse_connect = "diyala";
$username_connect = "root";
$password_connect = "";
$connect = mysql_pconnect($hostname_connect, $username_connect, $password_connect) or
trigger_error(mysql_error(),E_USER_ERROR);
?>
```

### **Cssmain.css**

```
body {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 14px;
    font-weight: bold;
    color: #990000;
    background-positi text-align: center;
    background-color: #669999;
    tion: center center;
    background-position: left;
    text-align: left;
}
table {
```

```

font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 14px;
font-weight: bold;
width: 100%;
background-position: center top;
line-height: normal;
text-align: center;
color: #CCCC99;
}

```

## Index.html

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
</head>

<frameset rows="57,*" cols="*" framespacing="1" frameborder="yes" border="1">
  <frame src="top.html" name="topFrame" scrolling="no" noresize="noresize" id="topFrame"
title="topFrame" />
  <frameset rows="*" cols="172,*" framespacing="1" frameborder="yes" border="1">
    <frame src="left.html" name="leftFrame" scrolling="No" noresize="noresize"
id="leftFrame" title="leftFrame" />
    <frameset rows="*" cols="*,160" framespacing="1" border="1">
      <frame src="main.php" name="mainFrame" scrolling="yes" noresize="noresize"
id="mainFrame" title="mainFrame" />
      <frame src="right.php" name="rightFrame" scrolling="No" noresize="noresize"
id="rightFrame" title="rightFrame" />
    </frameset>
  </frameset>
</frameset>
</frameset>
</frameset>
<noframes><body>
</body>

```

```
</noframes></html>
```

## **Studentall.php**

```
<?php require_once('Connections/connect.php'); ?>
```

```
<?php
```

```
$currentPage = $_SERVER["PHP_SELF"];

```

```
$maxRows_all = 10;

```

```
$pageNum_all = 0;

```

```
if (isset($_GET['pageNum_all'])) {

```

```
    $pageNum_all = $_GET['pageNum_all'];

```

```
}

```

```
$startRow_all = $pageNum_all * $maxRows_all;

```

```
mysql_select_db($database_connect, $connect);

```

```
$query_all = "SELECT * FROM college, deprements, student WHERE
student.id_colleg=college.id and deprements.id_depr=student.id_dep";

```

```
$query_limit_all = sprintf("%s LIMIT %d, %d", $query_all, $startRow_all, $maxRows_all);

```

```
$all = mysql_query($query_limit_all, $connect) or die(mysql_error());

```

```
$row_all = mysql_fetch_assoc($all);

```

```
if (isset($_GET['totalRows_all'])) {

```

```
    $totalRows_all = $_GET['totalRows_all'];

```

```
} else {

```

```
    $all_all = mysql_query($query_all);

```

```
    $totalRows_all = mysql_num_rows($all_all);

```

```
}

```

```
$totalPages_all = ceil($totalRows_all/$maxRows_all)-1;

```

```
mysql_select_db($database_connect, $connect);

```

```
$query_college = "SELECT id, name FROM college";

```

```
$college = mysql_query($query_college, $connect) or die(mysql_error());

```

```
$row_college = mysql_fetch_assoc($college);

```

```
$totalRows_college = mysql_num_rows($college);

```

```
mysql_select_db($database_connect, $connect);

```

```
$query_dep = "SELECT id_depr, name_dep FROM deprements ";

```

```
$dep = mysql_query($query_dep, $connect) or die(mysql_error());

```

```
$row_dep = mysql_fetch_assoc($dep);

```

```
$totalRows_dep = mysql_num_rows($dep);

```

```
$maxRows_Recordset1 = 5;

```

```

$pageNum_Recordset1 = 0;
if (isset($_GET['pageNum_Recordset1'])) {
    $pageNum_Recordset1 = $_GET['pageNum_Recordset1'];
}
$startRow_Recordset1 = $pageNum_Recordset1 * $maxRows_Recordset1;

$colname2_Recordset1 = "-1";
if (isset($_POST['de'])) {
    $colname2_Recordset1 = (get_magic_quotes_gpc()) ? $_POST['de'] :
addslashes($_POST['de']);
}
$colname_Recordset1 = "-1";
if (isset($_POST['class'])) {
    $colname_Recordset1 = (get_magic_quotes_gpc()) ? $_POST['class'] :
addslashes($_POST['class']);
}
$colname1_Recordset1 = "-1";
if (isset($_POST['col'])) {
    $colname1_Recordset1 = (get_magic_quotes_gpc()) ? $_POST['col'] :
addslashes($_POST['col']);
}
mysql_select_db($database_connect, $connect);
$query_Recordset1 = sprintf("SELECT student.id_stud, student.name_stud, student.addres,
student.kypc, college.name, depraments.name_dep, student.`Data Year` FROM student, college,
depraments WHERE kypc = %s and student.id_colleg= %s and student.id_dep=%s AND
student.id_colleg = college.id and student.id_dep=depraments.id_depr",
$colname_Recordset1,$colname1_Recordset1,$colname2_Recordset1);
$query_limit_Recordset1 = sprintf("%s LIMIT %d, %d", $query_Recordset1,
$startRow_Recordset1, $maxRows_Recordset1);
$Recordset1 = mysql_query($query_limit_Recordset1, $connect) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

if (isset($_GET['totalRows_Recordset1'])) {
    $totalRows_Recordset1 = $_GET['totalRows_Recordset1'];
} else {

```



```

$all_Recordset1 = mysql_query($query_Recordset1);
$totalRows_Recordset1 = mysql_num_rows($all_Recordset1);
}
$totalPages_Recordset1 = ceil($totalRows_Recordset1/$maxRows_Recordset1)-1;
$queryString_all = "";
if (!empty($_SERVER['QUERY_STRING'])) {
    $params = explode("&", $_SERVER['QUERY_STRING']);
    $newParams = array();
    foreach ($params as $param) {
        if (strpos($param, "pageNum_all") == false &&
            strpos($param, "totalRows_all") == false) {
            array_push($newParams, $param);
        }
    }
    if (count($newParams) != 0) {
        $queryString_all = "&" . htmlentities(implode("&", $newParams));
    }
}
$queryString_all = sprintf("&totalRows_all=%d%s", $totalRows_all, $queryString_all);
$queryString_Recordset1 = "";
if (!empty($_SERVER['QUERY_STRING'])) {
    $params = explode("&", $_SERVER['QUERY_STRING']);
    $newParams = array();
    foreach ($params as $param) {
        if (strpos($param, "pageNum_Recordset1") == false &&
            strpos($param, "totalRows_Recordset1") == false) {
            array_push($newParams, $param);
        }
    }
    if (count($newParams) != 0) {
        $queryString_Recordset1 = "&" . htmlentities(implode("&", $newParams));
    }
}
$queryString_Recordset1 = sprintf("&totalRows_Recordset1=%d%s", $totalRows_Recordset1,
$queryString_Recordset1);

```

```

?><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
<link href="cssmain.css" rel="stylesheet" type="text/css" />
<style type="text/css">
<!--
.style3 { color: #920000}
.style4 { font-size: 24px}
.style5 { color: #990000}
-->
</style>
<script type="text/JavaScript">
<!--
function MM_findObj(n, d) { //v4.01
  var p,i,x; if(!d) d=document; if((p=n.indexOf("?"))>0&&parent.frames.length) {
    d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
  if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
  for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findObj(n,d.layers[i].document);
  if(!x && d.getElementById) x=d.getElementById(n); return x;
}
function MM_validateForm() { //v4.0
  var i,p,q,nm,test,num,min,max,errors="",args=MM_validateForm.arguments;
  for (i=0; i<(args.length-2); i+=3) { test=args[i+2]; val=MM_findObj(args[i]);
  if (val) { nm=val.name; if ((val=val.value)!="") {
    if (test.indexOf('isEmail')!=-1) { p=val.indexOf('@');
      if (p<1 || p==(val.length-1)) errors+='- '+nm+' must contain an e-mail address.\n';
    } else if (test!='R') { num = parseFloat(val);
      if (isNaN(val)) errors+='- '+nm+' must contain a number.\n';
      if (test.indexOf('inRange') != -1) { p=test.indexOf(':');
        min=test.substring(8,p); max=test.substring(p+1);
        if (num<min || max<num) errors+='- '+nm+' must contain a number between '+min+' and
'+max+'.\n';

```

```

    } } } else if (test.charAt(0) == 'R') errors += '- '+nm+' is required.\n'; }
} if (errors) alert("The following error(s) occurred:\n'+errors);
document.MM_returnValue = (errors == "");
}
//-->
</script>
</head>
<body>
<form id="form1" name="form1" method="post" action="studentall.php">
  <label>
  <div align="center"><span class="style4"> Diyala University</span> <br />
    <br />
  </div>
</label>
<label></label>
<label>
  <select name="col" id="col">
    <option value="value">Select College</option>
    <?php
do {
?>
  <option value="<?php echo $row_college['id']?>"><?php echo
$row_college['name']?></option>
  <?php
} while ($row_college = mysql_fetch_assoc($college));
$rows = mysql_num_rows($college);
if($rows > 0) {
  mysql_data_seek($college, 0);
  $row_college = mysql_fetch_assoc($college);
}
?>
</select>
</label>
<label>
  <select name="de" id="de">

```

```

<option value="value">Select Deparment</option>
<?php
do {
?>
    <option value="<?php echo $row_dep['id_depr']?>"><?php echo
$row_dep['name_dep']?></option>
    <?php
} while ($row_dep = mysql_fetch_assoc($dep));
$rows = mysql_num_rows($dep);
if($rows > 0) {
    mysql_data_seek($dep, 0);
    $row_dep = mysql_fetch_assoc($dep);
}
?>
</select>
Enter Class
</label>
<label>
<input name="class" type="text" id="class"
onblur="MM_validateForm('class','RinRange1:5');return document.MM_returnValue"
size="7" />
</label>
<label>
<input name="Submit" type="submit" value="Submit" />
</label>
</form>
<?php if ($totalRows_Recordset1 > 0) { // Show if recordset not empty ?>
    <table border="2">
        <tr>
            <td width="5%"><span class="style5">ID</span></td>
            <td width="28%"><span class="style5">Name</span></td>
            <td width="18%"><span class="style5">Adress</span></td>
            <td width="11%"><span class="style5">College</span></td>
            <td width="11%"><span class="style5">Deparment</span></td>
            <td width="2%"><span class="style5">Class</span></td>

```

```

<td width="3%"><span class="style5">Year</span></td>
</tr>
<?php do { ?>
  <tr>
    <td><?php echo $row_Recordset1['id_stud']; ?></td>
    <td><?php echo $row_Recordset1['name_stud']; ?></td>
    <td><?php echo $row_Recordset1['adres']; ?></td>
    <td><?php echo $row_Recordset1['name']; ?></td>
    <td><?php echo $row_Recordset1['name_dep']; ?></td>
    <td><?php echo $row_Recordset1['kypc']; ?></td>
    <td><?php echo $row_Recordset1['Data Year']; ?></td>
  </tr>
  <?php } while ($row_Recordset1 = mysql_fetch_assoc($Recordset1)); ?>
</table>
<?php } // Show if recordset not empty ?><table border="0" width="50%"
align="center">
  <?php if ($totalRows_Recordset1 > 0) { // Show if recordset not empty ?>
    <tr>
      <td width="23%" align="center"><?php if ($pageNum_Recordset1 > 0) { // Show if
not first page ?>
        <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0,
$queryString_Recordset1); ?>"></a>
        <?php } // Show if not first page ?>
      </td>
      <td width="31%" align="center"><?php if ($pageNum_Recordset1 > 0) { // Show if
not first page ?>
        <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, max(0,
$pageNum_Recordset1 - 1), $queryString_Recordset1); ?>"></a>
        <?php } // Show if not first page ?>
      </td>
      <td width="23%" align="center"><?php if ($pageNum_Recordset1 <
$totalPages_Recordset1) { // Show if not last page ?>

```

```

        <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
min($totalPages_Recordset1, $pageNum_Recordset1 + 1), $queryString_Recordset1);
?"></a>
        <?php } // Show if not last page ?>
    </td>
    <td width="23%" align="center"><?php if ($pageNum_Recordset1 <
$totalPages_Recordset1) { // Show if not last page ?>
        <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$totalPages_Recordset1, $queryString_Recordset1); ?>"></a>
        <?php } // Show if not last page ?>
    </td>
</tr>
<?php } // Show if recordset not empty ?>
</table>
<?php if ($totalRows_Recordset1 == 0) { // Show if recordset empty ?>
    <table width="100%" border="2" align="center">
        <tr align="center" valign="top">
            <td width="5%"><span class="style3">ID</span></td>
            <td width="28%"><span class="style3">Name</span></td>
            <td width="18%"><span class="style3">Adress</span></td>
            <td width="11%"><span class="style3">College</span></td>
            <td width="11%"><span class="style3">Deperament</span></td>
            <td width="2%"><span class="style3">Class</span></td>
            <td width="3%"><span class="style5">Year</span></td>
        </tr>
        <?php do { ?>
            <tr align="center" valign="top">
                <td><?php echo $row_all['id_stud']; ?></td>
                <td><?php echo $row_all['name_stud']; ?></td>
                <td><?php echo $row_all['adres']; ?></td>
                <td><?php echo $row_all['name']; ?></td>
                <td><?php echo $row_all['name_dep']; ?></td>
                <td><?php echo $row_all['kypc']; ?></td>
                <td><?php echo $row_all['Data Year']; ?></td>
            </tr>

```

```

    <?php } while ($row_all = mysql_fetch_assoc($all)); ?>
</table>
<?php } // Show if recordset empty ?><table border="0" width="50%" align="center">
<?php if ($totalRows_Recordset1 == 0) { // Show if recordset empty ?>
    <tr>
        <td width="23%" align="center"><?php if ($pageNum_all > 0) { // Show if not first
page ?>
            <a href="<?php printf("%s?pageNum_all=%d%s", $currentPage, 0,
$queryString_all); ?>"></a>
            <?php } // Show if not first page ?>
        </td>
        <td width="31%" align="center"><?php if ($pageNum_all > 0) { // Show if not first
page ?>
            <a href="<?php printf("%s?pageNum_all=%d%s", $currentPage, max(0,
$pageNum_all - 1), $queryString_all); ?>"></a>
            <?php } // Show if not first page ?>
        </td>
        <td width="23%" align="center"><?php if ($pageNum_all < $totalPages_all) { //
Show if not last page ?>
            <a href="<?php printf("%s?pageNum_all=%d%s", $currentPage,
min($totalPages_all, $pageNum_all + 1), $queryString_all); ?>"></a>
            <?php } // Show if not last page ?>
        </td>
        <td width="23%" align="center"><?php if ($pageNum_all < $totalPages_all) { //
Show if not last page ?>
            <a href="<?php printf("%s?pageNum_all=%d%s", $currentPage, $totalPages_all,
$queryString_all); ?>"></a>
            <?php } // Show if not last page ?>
        </td>
    </tr>
    <?php } // Show if recordset empty ?>
</table>
<p>&nbsp;</p>
</body>

```

```

</html>
<?php
mysql_free_result($all);
mysql_free_result($college);
mysql_free_result($dep);
mysql_free_result($Recordset1);
?>

```

## Enter To System.php

```

<?php require_once('Connections/connect.php'); ?>
<?php
// *** Validate request to login to this site.
if (!isset($_SESSION)) {
    session_start();
}

$loginFormAction = $_SERVER['PHP_SELF'];
if (isset($_GET['accesscheck'])) {
    $_SESSION['PrevUrl'] = $_GET['accesscheck'];
}

if (isset($_POST['pass'])) {
    $loginUsername=$_POST['pass'];
    $password=$_POST['user'];
    $MM_fldUserAuthorization = "";
    $MM_redirectLoginSuccess = "editdatabase.php";
    $MM_redirectLoginFailed = "user.php";
    $MM_redirecttoReferrer = false;
    mysql_select_db($database_connect, $connect);

    $LoginRS__query=sprintf("SELECT pass, user FROM input WHERE pass='%s' AND
user='%s'",
        get_magic_quotes_gpc() ? $loginUsername : addslashes($loginUsername),
        get_magic_quotes_gpc() ? $password : addslashes($password));

```



```

$LoginRS = mysql_query($LoginRS__query, $connect) or die(mysql_error());
$loginFoundUser = mysql_num_rows($LoginRS);
if ($loginFoundUser) {
    $loginStrGroup = "";

    //declare two session variables and assign them
    $_SESSION['MM_Username'] = $loginUsername;
    $_SESSION['MM_UserGroup'] = $loginStrGroup;

    if (isset($_SESSION['PrevUrl']) && false) {
        $MM_redirectLoginSuccess = $_SESSION['PrevUrl'];
    }
    header("Location: " . $MM_redirectLoginSuccess );
}
else {
    header("Location: ". $MM_redirectLoginFailed );
}
}
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
<style type="text/css">
<!--
body {
    background-color: #669999;
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-weight: bold;
    font-size: 36px;
}
-->

```

```
</style>
```

```
<style type="text/css">
```

```
<!--
```

```
.style1 {font-size: 18px }
```

```
body,td,th {
```

```
    font-family: Georgia, Times New Roman, Times, serif;
```

```
    font-size: 14px;
```

```
    color: #990000;
```

```
    font-weight: bold;
```

```
}
```

```
-->
```

```
</style>
```

```
<script type="text/JavaScript">
```

```
<!--
```

```
function MM_findObj(n, d) { //v4.01
```

```
    var p,i,x; if(!d) d=document; if((p=n.indexOf("?"))>0&&parent.frames.length) {
```

```
        d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
```

```
    if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
```

```
    for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findObj(n,d.layers[i].document);
```

```
    if(!x && d.getElementById) x=d.getElementById(n); return x;
```

```
}
```

```
function MM_validateForm() { //v4.0
```

```
    var i,p,q,nm,test,num,min,max,errors="",args=MM_validateForm.arguments;
```

```
    for (i=0; i<(args.length-2); i+=3) { test=args[i+2]; val=MM_findObj(args[i]);
```

```
        if (val) { nm=val.name; if ((val=val.value)!="") {
```

```
            if (test.indexOf('isEmail')!=-1) { p=val.indexOf('@');
```

```
                if (p<1 || p==(val.length-1)) errors+='- '+nm+' must contain an e-mail address.\n';
```

```
            } else if (test!='R') { num = parseFloat(val);
```

```
                if (isNaN(val)) errors+='- '+nm+' must contain a number.\n';
```

```
                if (test.indexOf('inRange') != -1) { p=test.indexOf(':');
```

```
                    min=test.substring(8,p); max=test.substring(p+1);
```

```
                    if (num<min || max<num) errors+='- '+nm+' must contain a number between '+min+' and '+max+'.\n';
```

```

    } } } else if (test.charAt(0) == 'R') errors += '- '+nm+' is required.\n'; }
} if (errors) alert("The following error(s) occurred:\n'+errors);
document.MM_returnValue = (errors == "");
}
//-->
</script>
</head>

<body>
<form ACTION="<?php echo $loginFormAction; ?>" method="POST" name="form3"
id="form3" onsubmit="MM_validateForm('user','R','pass','RisNum');return
document.MM_returnValue">
  <p>
  <label>
    <div align="center">

  <p align="center">&nbsp;</p>
  <p align="center">&nbsp;</p>
  <p align="center">&nbsp;</p>
  <p align="center"><span class="style1">Edit Data Base </span></p>
  <p align="center">
  </p>
  <table border="2" align=center >
  <tr align=left valign="top">
  <td width="181"> <div align="center">Enter Your Name</div></td>
  <td width="211" ><input name="user" type="text" id="user" size="30" /></td>
  </tr>

  <tr align=left valign="top">
  <td width="181" bgcolor="#669999"> <div align="center">Enter Your Password</div></td>

  <td> <input name="pass" type="password" id="pass" size="10" /></td>
  </tr>
</table>

```

```

<p align="center">
  <label>
    <input type="submit" name="Submit2" value="Go" />
  </label>
</p>
</form>
</body>
</html>

```

## Edit Student.php

```

<?php require_once('Connections/connect.php'); ?>
<?php
//initialize the session
if (!isset($_SESSION)) {
    session_start();
}

// ** Logout the current user. **
$logoutAction = $_SERVER['PHP_SELF']."?doLogout=true";
if ((isset($_SERVER['QUERY_STRING'])) && ($_SERVER['QUERY_STRING'] != "")){
    $logoutAction .="&". htmlentities($_SERVER['QUERY_STRING']);
}

if ((isset($_GET['doLogout'])) &&($_GET['doLogout']=="true")){
    //to fully log out a visitor we need to clear the session variables
    $_SESSION['MM_Username'] = NULL;
    $_SESSION['MM_UserGroup'] = NULL;
    $_SESSION['PrevUrl'] = NULL;
    unset($_SESSION['MM_Username']);
    unset($_SESSION['MM_UserGroup']);
    unset($_SESSION['PrevUrl']);

    $logoutGoTo = "user.php";
    if ($logoutGoTo) {
        header("Location: $logoutGoTo");
        exit;
    }
}

```

```

    }
}
?>
<?php
if (!function_exists("GetSQLValueString")) {
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
$theNotDefinedValue = "")
{
    if (PHP_VERSION < 6) {
        $theValue = get_magic_quotes_gpc() ? stripslashes($theValue) : $theValue;
    }

    $theValue = function_exists("mysql_real_escape_string") ?
mysql_real_escape_string($theValue) : mysql_escape_string($theValue);

    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "long":
        case "int":
            $theValue = ($theValue != "") ? intval($theValue) : "NULL";
            break;
        case "double":
            $theValue = ($theValue != "") ? doubleval($theValue) : "NULL";
            break;
        case "date":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "defined":
            $theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
            break;
    }
    return $theValue;
}
}

```

```

}
?>
<?php require_once('Connections/connect.php'); ?>
<?php
$currentPage = $_SERVER["PHP_SELF"];

function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
$theNotDefinedValue = "")
{
    $theValue = (!get_magic_quotes_gpc()) ? addslashes($theValue) : $theValue;

    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "long":
        case "int":
            $theValue = ($theValue != "") ? intval($theValue) : "NULL";
            break;
        case "double":
            $theValue = ($theValue != "") ? "'" . doubleval($theValue) . "'" : "NULL";
            break;
        case "date":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "defined":
            $theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
            break;
    }
    return $theValue;
}

$editFormAction = $_SERVER['PHP_SELF'];
if (isset($_SERVER['QUERY_STRING'])) {
    $editFormAction .= "?" . htmlentities($_SERVER['QUERY_STRING']);
}

```

```
}

```

```
if ((isset($_GET['recordID'])) && ($_GET['recordID'] != "")) {
    $deleteSQL = sprintf("DELETE FROM student WHERE id_stud=%s",
        GetSQLValueString($_GET['recordID'], "int"));

```

```
mysql_select_db($database_connect, $connect);
$result1 = mysql_query($deleteSQL, $connect) or die(mysql_error());
}

```

```
if ((isset($_POST["MM_insert"])) && ($_POST["MM_insert"] == "form1")) {
    $insertSQL = sprintf("INSERT INTO student (name_stud, id_colleg, id_dep, adres, `Data
    Year`, kypc, image_stud) VALUES (%s, %s, %s, %s, %s, %s, %s)",

```

```
        GetSQLValueString($_POST['name_stud'], "text"),
        GetSQLValueString($_POST['id_colleg'], "int"),
        GetSQLValueString($_POST['id_dep'], "int"),
        GetSQLValueString($_POST['adres'], "text"),
        GetSQLValueString($_POST['Year'], "int"),
        GetSQLValueString($_POST['kypc'], "int"),
        GetSQLValueString($_POST['imagefile'], "text"));

```

```
mysql_select_db($database_connect, $connect);
$result1 = mysql_query($insertSQL, $connect) or die(mysql_error());
}

```

```
mysql_select_db($database_connect, $connect);
$query_All = "SELECT * FROM student";
$result_All = mysql_query($query_All, $connect) or die(mysql_error());
$row_All = mysql_fetch_assoc($result_All);
$totalRows_All = mysql_num_rows($result_All);

```

```
mysql_select_db($database_connect, $connect);
$query_college = "SELECT id, name FROM college";
$result_college = mysql_query($query_college, $connect) or die(mysql_error());
$row_college = mysql_fetch_assoc($result_college);

```

```

$totalRows_college = mysql_num_rows($college);

$colname_dep = "-1";
if (isset($_SESSION['MM_Username'])) {
    $colname_dep = (get_magic_quotes_gpc()) ? $_SESSION['MM_Username'] :
addslashes($_SESSION['MM_Username']);
}
mysql_select_db($database_connect, $connect);
$query_dep = sprintf("SELECT id_depr, name_dep FROM deprements WHERE
deprements.id_college=%s", $colname_dep);
$dep = mysql_query($query_dep, $connect) or die(mysql_error());
$row_dep = mysql_fetch_assoc($dep);
$totalRows_dep = mysql_num_rows($dep);

$maxRows_select = 8;
$pageNum_select = 0;
if (isset($_GET['pageNum_select'])) {
    $pageNum_select = $_GET['pageNum_select'];
}
$startRow_select = $pageNum_select * $maxRows_select;

$colname_select = "-1";
if (isset($_SESSION['MM_Username'])) {
    $colname_select = (get_magic_quotes_gpc()) ? $_SESSION['MM_Username'] :
addslashes($_SESSION['MM_Username']);
}
mysql_select_db($database_connect, $connect);
$query_select = sprintf(" SELECT * FROM college, deprements, student WHERE college. id =
%s AND student.id_colleg=college.id and deprements.id_depr=student.id_depr",
$colname_select);
$query_limit_select = sprintf("%s LIMIT %d, %d", $query_select, $startRow_select,
$maxRows_select);
$select = mysql_query($query_limit_select, $connect) or die(mysql_error());
$row_select = mysql_fetch_assoc($select);

```



```

if (isset($_GET['totalRows_select'])) {
    $totalRows_select = $_GET['totalRows_select'];
} else {
    $all_select = mysql_query($query_select);
    $totalRows_select = mysql_num_rows($all_select);
}
$totalPages_select = ceil($totalRows_select/$maxRows_select)-1;

$queryString_select = "";
if (!empty($_SERVER['QUERY_STRING'])) {
    $params = explode("&", $_SERVER['QUERY_STRING']);
    $newParams = array();
    foreach ($params as $param) {
        if (strpos($param, "pageNum_select") == false &&
            strpos($param, "totalRows_select") == false) {
            array_push($newParams, $param);
        }
    }
    if (count($newParams) != 0) {
        $queryString_select = "&" . htmlentities(implode("&", $newParams));
    }
}
$queryString_select = sprintf("&totalRows_select=%d%s", $totalRows_select,
$queryString_select);
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
<style type="text/css">
<!--
#11 {
    position:absolute;

```

```
    left:2px;
    top:2px;
    width:709px;
    height:336px;
    z-index:1;
}
#add1 {
    position:absolute;
    left:9px;
    top:4px;
    width:350px;
    height:296px;
    z-index:2;
    visibility: visible;
}
#di {
    position:absolute;
    left:594px;
    top:5px;
    width:321px;
    height:100px;
    z-index:3;
}
#search {
    position:absolute;
    left:596px;
    top:133px;
    width:321px;
    height:78px;
    z-index:4;
}
#select {
    position:absolute;
    left:6px;
    top:322px;
```

```

width:913px;
height:236px;
z-index:5;
}
-->
</style>
<link href="layerborder.css" rel="stylesheet" type="text/css" />
<link href="1.css" rel="stylesheet" type="text/css" />
<style type="text/css">
<!--
.style2 { color: #CCCC99}
#Layer1 {
    position:absolute;
    left:400px;
    top:33px;
    width:159px;
    height:59px;
    z-index:6;
}
-->
</style>
<script type="text/JavaScript">
<!--
function MM_findObj(n, d) { //v4.01
    var p,i,x; if(!d) d=document; if((p=n.indexOf("?"))>0&&parent.frames.length) {
        d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
    if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
    for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findObj(n,d.layers[i].document);
    if(!x && d.getElementById) x=d.getElementById(n); return x;
}

function MM_validateForm() { //v4.0
    var i,p,q,nm,test,num,min,max,errors="",args=MM_validateForm.arguments;
    for (i=0; i<(args.length-2); i+=3) { test=args[i+2]; val=MM_findObj(args[i]);
        if (val) { nm=val.name; if ((val=val.value)!="") {

```

```

if (test.indexOf('isEmail')!=-1) { p=val.indexOf('@');
    if (p<1 || p==(val.length-1)) errors+='- '+nm+' must contain an e-mail address.\n';
} else if (test!='R') { num = parseFloat(val);
    if (isNaN(val)) errors+='- '+nm+' must contain a number.\n';
    if (test.indexOf('inRange') != -1) { p=test.indexOf(':');
        min=test.substring(8,p); max=test.substring(p+1);
        if (num<min || max<num) errors+='- '+nm+' must contain a number between '+min+' and
'+max+'.\n';
    } } } else if (test.charAt(0) == 'R') errors += '- '+nm+' is required.\n'; }
} if (errors) alert("The following error(s) occurred:\n'+errors);
document.MM_returnValue = (errors == "");
}
//-->
</script>
</head>

<body>
<div class="layerBorder" id="add1">
    <p>Add New Student </p>

    <form action="<?php echo $editFormAction; ?>" method="post" name="form1"
onsubmit="MM_validateForm('name_stud','R','adres','R','kypc','RinRange1:5');return
document.MM_returnValue">
    <table align="center">
        <tr valign="baseline">
            <td nowrap align="right">Name Student:</td>
            <td><input type="text" name="name_stud" value="" size="32"></td>
        </tr>
        <tr valign="baseline">
            <td nowrap align="right">Deprament:</td>
            <td><select name="id_dep">
                <option value="value" <?php if (!(strcmp("value", "Select Dep")))) {echo
"selected=\"selected\"";} ?>>Select Deprament</option>
            </td>
        </tr>
    </table>
    <?php
do {

```

```

?><option value="<?php echo $row_dep['id_depr']?>"<?php if (!(strcmp($row_dep['id_depr'],
"Select Dep")) { echo "selected=\"selected\""; } ?>><?php echo
$row_dep['name_dep']?></option>
    <?php
} while ($row_dep = mysql_fetch_assoc($dep));
$rows = mysql_num_rows($dep);
if($rows > 0) {
    mysql_data_seek($dep, 0);
    $row_dep = mysql_fetch_assoc($dep);
}
?>
        </select>    </td>
<tr>
<tr valign="baseline">
    <td nowrap align="right">Adres:</td>
    <td><input type="text" name="adres" value="" size="32"></td>
</tr>
<tr valign="baseline">
    <td align="right" nowrap>Kypc:</td>
    <td><input type="text" name="kypc" value="" size="5"></td>
</tr>
<tr valign="baseline">
    <td align="right" nowrap>Year</td>
    <td><label>
        <input name="Year" type="text" id="Year" />
    </label></td>
</tr>
<tr valign="baseline">
    <td align="right" nowrap>Image</td>
    <td><input type="file" name="imagefile" />
    <?php
$stemp = preg_split ('/[\\\/]+/',$_files ["imagefile"]["name"]);
$filename =$stemp[count($stemp)-1];
$upload_dir="firas/image10/";
$upload_file=$upload_dir.$filename;

```

```

if (move_uploaded_file( $_files ["imagefile"]["tmp_name"],$upload_file))
{
    echo "file Ok";}

?>
    &nbsp;</td>
</tr>
<tr valign="baseline">
    <td nowrap align="right">&nbsp;</td>
    <td><p>
        <input type="submit" value="Insert record">
    </p>
    <p>&nbsp;</p></td>
</tr>
</table>
<input type="hidden" name="id_colleg" value="<?php echo $_SESSION['MM_Username'];
?>">
    <input type="hidden" name="MM_insert" value="form1">
</form>
<p align="left"><a href="editdatabase.php"></a> </p>
</div>
<div class="layerBorder" id="di">
    <div align="right"><a href="<?php echo $logoutAction ?>">Log out</a></div>
    <p>Diyala University </p>
    <p>College:<?php echo $row_select['name']; ?></p>
</div>
<div class="layerBorder" id="search">
    <form action="searchstudentmain1.php" method="get" name="form2" target="_blank"
id="form2" onsubmit="MM_validateForm('name','R');return document.MM_returnValue">
    <label for="textfield">Search Student </label>
    <input name="name" type="text" id="name" onblur="MM_validateForm('name','R');return
document.MM_returnValue" />

```

```

<label for="Submit"></label>
<input type="submit" name="Submit" value="Go" id="Submit" />
</form>
</div>
<div id="Layer1"><a href="editdatabase.php">Back To Home</a></div>
<p>&nbsp;</p>
<div class="layerBorder" id="select">
<table border="2">
<tr>
<td width="136">id</td>
<td width="136">Name</td>
<td width="106">Addres</td>
<td width="97">College</td>
<td width="130">Deprament</td>
<td width="45">Kypc</td>
<td width="45">Year</td>
<td width="51">&nbsp;</td>
<td width="54">&nbsp;</td>
</tr>
<?php do { ?>

<tr>
<td><span class="style2"><?php echo $row_select['id_stud']; ?></span></td>
<td><span class="style2"><?php echo $row_select['name_stud']; ?></span></td>
<td><span class="style2"><?php echo $row_select['addres']; ?></span></td>
<td><span class="style2"><?php echo $row_select['name']; ?></span></td>
<td><span class="style2"><?php echo $row_select['name_dep']; ?></span></td>
<td><span class="style2"><?php echo $row_select['kypc']; ?></span></td>
<td><?php echo $row_select['Data Year']; ?></td>
<td><a href="studentedit.php?recordID=<?php echo $row_select['id_stud']; ?>"
class="style2">Delete</a></td>
<td><a href="updatastuden1.php?recordID=<?php echo $row_select['id_stud']; ?>"
target="_blank" class="style2">Updata</a></td>
</tr>
<?php } while ($row_select = mysql_fetch_assoc($select)); ?>

```

```

</table>

<table border="0" width="50%" align="center">
  <tr>
    <td width="23%" align="center"><?php if ($pageNum_select > 0) { // Show if not first page
?>
      <a href="<?php printf("%s?pageNum_select=%d%s", $currentPage, 0,
$queryString_select); ?>"></a>
      <?php } // Show if not first page ?>    </td>
    <td width="31%" align="center"><?php if ($pageNum_select > 0) { // Show if not first page
?>
      <a href="<?php printf("%s?pageNum_select=%d%s", $currentPage, max(0,
$pageNum_select - 1), $queryString_select); ?>"></a>
      <?php } // Show if not first page ?>    </td>
    <td width="23%" align="center"><?php if ($pageNum_select < $totalPages_select) { //
Show if not last page ?>
      <a href="<?php printf("%s?pageNum_select=%d%s", $currentPage,
min($totalPages_select, $pageNum_select + 1), $queryString_select); ?>"></a>
      <?php } // Show if not last page ?>    </td>
    <td width="23%" align="center"><?php if ($pageNum_select < $totalPages_select) { //
Show if not last page ?>
      <a href="<?php printf("%s?pageNum_select=%d%s", $currentPage, $totalPages_select,
$queryString_select); ?>"></a>
      <?php } // Show if not last page ?>    </td>
  </tr>
</table>
</div>
</body>
</html>
<?php
mysql_free_result($All);

mysql_free_result($college);

```



```
mysql_free_result($dep);
```

```
mysql_free_result($select);
```

```
?>
```

### **Search Student.php**

```
<?php require_once('Connections/connect.php'); ?>
```

```
<?php
```

```
if (!function_exists("GetSQLValueString")) {
```

```
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
```

```
$theNotDefinedValue = "")
```

```
{
```

```
if (PHP_VERSION < 6) {
```

```
    $theValue = get_magic_quotes_gpc() ? stripslashes($theValue) : $theValue;
```

```
}
```

```
$theValue = function_exists("mysql_real_escape_string") ?
```

```
mysql_real_escape_string($theValue) : mysql_escape_string($theValue);
```

```
switch ($theType) {
```

```
case "text":
```

```
    $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
```

```
    break;
```

```
case "long":
```

```
case "int":
```

```
    $theValue = ($theValue != "") ? intval($theValue) : "NULL";
```

```
    break;
```

```
case "double":
```

```
    $theValue = ($theValue != "") ? doubleval($theValue) : "NULL";
```

```
    break;
```

```
case "date":
```

```
    $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
```

```
    break;
```

```
case "defined":
```

```
    $theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
```

```
    break;
```

```

}
return $theValue;
}
}

$maxRows_Recordset1 = 10;
$pageNum_Recordset1 = 0;
if (isset($_GET['pageNum_Recordset1'])) {
    $pageNum_Recordset1 = $_GET['pageNum_Recordset1'];
}
$startRow_Recordset1 = $pageNum_Recordset1 * $maxRows_Recordset1;

$colname_Recordset1 = "-1";
if (isset($_GET['name'])) {
    $colname_Recordset1 = $_GET['name'];
}
mysql_select_db($database_connect, $connect);
$query_Recordset1 = sprintf("SELECT student.id_stud, student.name_stud, student.addres,
student.kypc, college.name, depraments.name_dep, student.image_stud FROM student, college,
depraments WHERE name_stud LIKE %s and student.id_colleg=college.id and
student.id_dep=depraments.id_depr", GetSQLValueString($colname_Recordset1 . "%", "text"));
$query_limit_Recordset1 = sprintf("%s LIMIT %d, %d", $query_Recordset1,
$startRow_Recordset1, $maxRows_Recordset1);
$Recordset1 = mysql_query($query_limit_Recordset1, $connect) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

if (isset($_GET['totalRows_Recordset1'])) {
    $totalRows_Recordset1 = $_GET['totalRows_Recordset1'];
} else {
    $all_Recordset1 = mysql_query($query_Recordset1);
    $totalRows_Recordset1 = mysql_num_rows($all_Recordset1);
}
$totalPages_Recordset1 = ceil($totalRows_Recordset1/$maxRows_Recordset1)-1;

session_start();

```

```

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
<link href="cssmain.css" rel="stylesheet" type="text/css" />

<style type="text/css">
<!--
.style1 { color: #990000}
.style4 { font-size: 18px }
.style6 { font-size: 36px }
-->
</style>
</head>

<body>
<p>&nbsp;</p>
<p align="center"><span class="style6">Search Results Of Students </span></p>

<?php if ($totalRows_Recordset1 == 0) { // Show if recordset empty ?>
  <p align="center" class="style4">Sorry,there are no records match the student name you have
entered !!!! </p>
  <?php } // Show if recordset empty ?>
<?php if ($totalRows_Recordset1 > 0) { // Show if recordset not empty ?>
  <table border="1">
  <tr>
    <td width="2%"><span class="style1">Id</span></td>
    <td width="3%"><span class="style1 style1">Image</span></td>
    <td width="28%"><span class="style1 style1">Name</span></td>
    <td width="18%"><span class="style1 style1">Adress</span></td>
    <td width="11%"><span class="style1">College</span></td>
    <td width="11%"><span class="style1 style1">Deparment</span></td>

```

```

<td width="5%"><span class="style1 style1">Class</span></td>
</tr>
<?php do { ?>
    <tr>
        <td><?php echo $row_Recordset1['id_stud']; ?></td>
        <td><?php echo "<img src = '". $row_Recordset1['image_stud']. "'/>"; ?></td>
        <td><?php echo $row_Recordset1['name_stud']; ?></td>
        <td><?php echo $row_Recordset1['adres']; ?></td>
        <td><?php echo $row_Recordset1['name']; ?></td>
        <td><?php echo $row_Recordset1['name_dep']; ?></td>
        <td><?php echo $row_Recordset1['kypc']; ?></td>
    </tr>
    <?php } while ($row_Recordset1 = mysql_fetch_assoc($Recordset1)); ?>
</table>
<?php } // Show if recordset not empty ?>
<p>&nbsp;</p>
</body>
</html>
<?php
mysql_free_result($Recordset1);
?>

```

### Chart Student.php

```

<?php require_once('Connections/connect.php'); ?>
<?php include('includes/FusionCharts/dynamic/DWFChart.inc.php'); ?>
<?php
// fusion chart # FusionChart1 data include
include('includes/FusionCharts/dynamic/data/fc_FusionChart1_data.php');

// fusion chart # FusionChart2 data include
include('includes/FusionCharts/dynamic/data/fc_FusionChart2_data.php');

if (!function_exists("GetSQLValueString")) {
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
    $theNotDefinedValue = "")
{

```

```

if (PHP_VERSION < 6) {
    $theValue = get_magic_quotes_gpc() ? stripslashes($theValue) : $theValue;
}

```

```

$theValue = function_exists("mysql_real_escape_string") ?
mysql_real_escape_string($theValue) : mysql_escape_string($theValue);

```

```

switch ($theType) {
    case "text":
        $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
        break;
    case "long":
    case "int":
        $theValue = ($theValue != "") ? intval($theValue) : "NULL";
        break;
    case "double":
        $theValue = ($theValue != "") ? doubleval($theValue) : "NULL";
        break;
    case "date":
        $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
        break;
    case "defined":
        $theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
        break;
}

```

```

return $theValue;

```

```

}

```

```

}

```

```

?>

```

```

<?php require_once('Connections/connect.php'); ?>

```

```

<?php

```

```

$maxRows_colege = 10;

```

```

$pageNum_colege = 0;

```

```

if (isset($_GET['pageNum_colege'])) {

```

```

$pageNum_colege = $_GET['pageNum_colege'];
}
$startRow_colege = $pageNum_colege * $maxRows_colege;

mysql_select_db($database_connect, $connect);
$query_colege = "SELECT id, name FROM college";
$query_limit_colege = sprintf("%s LIMIT %d, %d", $query_colege, $startRow_colege,
$maxRows_colege);
$colege = mysql_query($query_limit_colege, $connect) or die(mysql_error());
$row_colege = mysql_fetch_assoc($colege);

if (isset($_GET['totalRows_colege'])) {
    $totalRows_colege = $_GET['totalRows_colege'];
} else {
    $all_colege = mysql_query($query_colege);
    $totalRows_colege = mysql_num_rows($all_colege);
}
$totalPages_colege = ceil($totalRows_colege/$maxRows_colege)-1;

$colname_Recordset1 = "-1";
if (isset($_POST['select'])) {
    $colname_Recordset1 = (get_magic_quotes_gpc()) ? $_POST['select'] :
addslashes($_POST['select']);
}
mysql_select_db($database_connect, $connect);
$query_Recordset1 = sprintf("SELECT * FROM college WHERE id = %s",
$colname_Recordset1);
$Recordset1 = mysql_query($query_Recordset1, $connect) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);
$totalRows_Recordset1 = mysql_num_rows($Recordset1);

$colname_dep10 = "-1";
if (isset($_POST['select'])) {
    $colname_dep10 = (get_magic_quotes_gpc()) ? $_POST['select'] :
addslashes($_POST['select']);
}

```

```

}
mysql_select_db($database_connect, $connect);
$query_dep10 = sprintf("SELECT count( name_dep) FROM deprements WHERE id_college =
%s", $colname_dep10);
$dep10 = mysql_query($query_dep10, $connect) or die(mysql_error());
$row_dep10 = mysql_fetch_assoc($dep10);
$totalRows_dep10 = mysql_num_rows($dep10);

$colname_dep11 = "-1";
if (isset($_POST['select'])) {
    $colname_dep11 = (get_magic_quotes_gpc()) ? $_POST['select'] :
addslashes($_POST['select']);
}
mysql_select_db($database_connect, $connect);
$query_dep11 = sprintf("SELECT id_depr, name_dep FROM deprements WHERE id_college =
%s", $colname_dep11);
$dep11 = mysql_query($query_dep11, $connect) or die(mysql_error());
$row_dep11 = mysql_fetch_assoc($dep11);
$totalRows_dep11 = mysql_num_rows($dep11);

$colname_lect10 = "-1";
if (isset($_POST['select'])) {
    $colname_lect10 = (get_magic_quotes_gpc()) ? $_POST['select'] :
addslashes($_POST['select']);
}
mysql_select_db($database_connect, $connect);
$query_lect10 = sprintf("SELECT count( name_lect) FROM lectuere WHERE id_college =
%s", $colname_lect10);
$lect10 = mysql_query($query_lect10, $connect) or die(mysql_error());
$row_lect10 = mysql_fetch_assoc($lect10);
$totalRows_lect10 = mysql_num_rows($lect10);

$colname_lect11 = "-1";
if (isset($_POST['select'])) {

```

```

$colname_lect11 = (get_magic_quotes_gpc()) ? $_POST['select'] :
addslashes($_POST['select']);
}
mysql_select_db($database_connect, $connect);
$query_lect11 = sprintf("SELECT id_lect, name_lect FROM lectuere WHERE id_college =
%s", $colname_lect11);
$result11 = mysql_query($query_lect11, $connect) or die(mysql_error());
$row_lect11 = mysql_fetch_assoc($result11);
$totalRows_lect11 = mysql_num_rows($result11);

$colname_stud10 = "-1";
if (isset($_POST['select'])) {
    $colname_stud10 = (get_magic_quotes_gpc()) ? $_POST['select'] :
addslashes($_POST['select']);
}
mysql_select_db($database_connect, $connect);
$query_stud10 = sprintf("SELECT count( name_stud) FROM student WHERE id_colleg = %s",
$colname_stud10);
$result10 = mysql_query($query_stud10, $connect) or die(mysql_error());
$row_stud10 = mysql_fetch_assoc($result10);
$totalRows_stud10 = mysql_num_rows($result10);

$colname_stud11 = "-1";
if (isset($_POST['select'])) {
    $colname_stud11 = (get_magic_quotes_gpc()) ? $_POST['select'] :
addslashes($_POST['select']);
}
mysql_select_db($database_connect, $connect);
$query_stud11 = sprintf("SELECT id_stud, name_stud FROM student WHERE id_colleg =
%s", $colname_stud11);
$result11 = mysql_query($query_stud11, $connect) or die(mysql_error());
$row_stud11 = mysql_fetch_assoc($result11);
$totalRows_stud11 = mysql_num_rows($result11);

mysql_select_db($database_connect, $connect);

```



```
$query_Charts = "SELECT count(name_stud), id_colleg, college.name FROM student, college
WHERE college.id=student.id_colleg GROUP BY id_colleg";
```

```
$Charts = mysql_query($query_Charts, $connect) or die(mysql_error());
```

```
$row_Charts = mysql_fetch_assoc($Charts);
```

```
$totalRows_Charts = mysql_num_rows($Charts);
```

```
mysql_select_db($database_connect, $connect);
```

```
$query_Gharttecher = "SELECT count(name_lect), id_college ,college.name FROM
lectuere,college WHERE college.id= lectuere.id_college GROUP BY id_college";
```

```
$Gharttecher = mysql_query($query_Gharttecher, $connect) or die(mysql_error());
```

```
$row_Gharttecher = mysql_fetch_assoc($Gharttecher);
```

```
$totalRows_Gharttecher = mysql_num_rows($Gharttecher);
```

```
// fusion chart #FusionChart1 dynamic filter data
```

```
// fusion chart #FusionChart2 dynamic filter data
```

```
$dFCFusionChart1 = new DWFCChart("FusionChart1", "Column3D",
```

```
"includes/FusionCharts/charts/", 400, 400, "", "", "", "", "", "");
```

```
$dFCFusionChart1->setVersion("1.1.0");
```

```
$dFCFusionChart1->setConfigXML($FusionChart1_dataXML);
```

```
$dFCFusionChart1->setCategory($Charts, "name", "", "");
```

```
$dFCFusionChart1->addSeries($Charts, "count(name_stud)", "", "color='195118'
seriesName='count(name_stud)' renderAs='Column' parentYAxis='P'", "default", "");
```

```
$dFCFusionChart1->setOrdering('None', 'asc');
```

```
$dFCFusionChart1->prepareData();
```

```
$dFCFusionChart2 = new DWFCChart("FusionChart2", "Column3D",
```

```
"includes/FusionCharts/charts/", 400, 400, "", "", "", "", "", "");
```

```
$dFCFusionChart2->setVersion("1.1.0");
```

```
$dFCFusionChart2->setConfigXML($FusionChart2_dataXML);
```

```
$dFCFusionChart2->setCategory($Gharttecher, "name", "", "");
```

```
$dFCFusionChart2->addSeries($Gharttecher, "count(name_lect)", "", "color='211703'
seriesName='count(name_lect)' renderAs='Column' parentYAxis='P'", "default", "");
```

```
$dFCFusionChart2->setOrdering('None', 'asc');
```

```
$dFCFusionChart2->prepareData();
```

```
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
<style type="text/css">
<!--
.style8 { font-size: 24px }
-->
</style>
<link href="cssmain100.css" rel="stylesheet" type="text/css" />
<script type="text/javascript"
src="includes/FusionCharts/dynamic/js/FusionCharts.js"></script>
<style type="text/css">
<!--
#apDiv1 {
    position:absolute;
    left:425px;
    top:39px;
    width:223px;
    height:36px;
    z-index:1;
    visibility: hidden;
}
#apDiv4 {
    position:absolute;
    left:11px;
    top:244px;
    width:116px;
    height:18px;
    z-index:3;
    visibility: visible;
}
```

```
#apDiv2 {  
    position:absolute;  
    left:12px;  
    top:270px;  
    width:116px;  
    height:16px;  
    z-index:3;  
    visibility: visible;  
}  
#apDiv3 {  
    position:absolute;  
    left:197px;  
    top:274px;  
    width:116px;  
    height:31px;  
    z-index:3;  
    visibility: visible;  
}  
#apDiv5 {  
    position:absolute;  
    left:424px;  
    top:52px;  
    width:198px;  
    height:108px;  
    z-index:4;  
    visibility: hidden;  
}  
#apDiv6 { position:absolute;  
    left:662px;  
    top:36px;  
    width:223px;  
    height:36px;  
    z-index:1;  
    visibility: visible;  
}
```

```

-->
</style>
<script type="text/javascript">
<!--
function MM_showHideLayers() { //v9.0
  var i,p,v,obj,args=MM_showHideLayers.arguments;
  for (i=0; i<(args.length-2); i+=3)
    with (document) if (getElementById && ((obj=getElementById(args[i])!=null)) {
v=args[i+2];
  if (obj.style) { obj=obj.style; v=(v=='show')?'visible':(v=='hide')?'hidden':v; }
  obj.visibility=v; }
}
//-->
</script>
</head>

<body>
<div id="apDiv4" onclick="MM_showHideLayers('apDiv5','hide','apDiv1','show')">Count
Student</div>
<div id="apDiv5"><?php
// (FCChart Begin) #FusionChart2
renderFusionChart($dFCFusionChart2, 400, 400);
// #FusionChart2 (FCChart End)
?></div>
<form action="college1.php" method="post" enctype="" name="form1" id="form1">
<label>
<div align="center"><span class="style8">Diyala University </span><br />
</div>
</label>
<p>
<label>
<select name="select" id="select">
<option value="value" <?php if (!(strcmp("Firaas", $row_colege['name_colege']))) {echo
"selected=\"selected\"Firas";} ?>>Select Your College</option>
<?php

```

```

do {
?>
    <option value="<?php echo $row_colege['id']?>"<?php if (!(strcmp($row_colege['id'],
$row_colege['name_colege']))) {echo "selected=\"selected\"";} ?>><?php echo
$row_colege['name']?></option>
    <?php
} while ($row_colege = mysql_fetch_assoc($colege));
$rows = mysql_num_rows($colege);
if($rows > 0) {
    mysql_data_seek($colege, 0);
    $row_colege = mysql_fetch_assoc($colege);
}
?>
</select>
<input name="Submit" type="submit" value="Submit" />
</label>
</p>
</form>
<form id="form2" name="form2" method="post" action="">
<table width="64%">
<tr>
<td width="18%"><label>Number College </label></td>
<td width="46%"><input name="textfield" type="text" value="<?php echo
$row_Recordset1['id']; ?>" size="5" /></td></tr>
<tr>
<td width="18%"><label>Name Colege </label></td>
<td><input name="textfield2" type="text" value="<?php echo $row_Recordset1['name']; ?>"
</td></tr>
<tr>
<td width="18%"><label>Quantity Deprament</label></td>
<td><input name="textfield3" type="text" value="<?php echo $row_dep10['count(
name_dep)']; ?>" size="5" />
<label>
<select name="select3">

```

```

    <option value="value">Deparment</option>
    <?php
do {
?>
    <option value="<?php echo $row_dep11['id_depr']?>"><?php echo
$row_dep11['name_dep']?></option>
    <?php
} while ($row_dep11 = mysql_fetch_assoc($dep11));
$rows = mysql_num_rows($dep11);
if($rows > 0) {
    mysql_data_seek($dep11, 0);
    $row_dep11 = mysql_fetch_assoc($dep11);
}
?>
</select>
</label></td></tr>
<tr>
<td width="18%"><label>Quantity Lectueres</label></td>
<td><input name="textfield4" type="text" value="<?php echo $row_lect10['count(
name_lect)']; ?>" size="5" />
<label>
<select name="select">
    <option value="value">Lectueres</option>
    <?php
do {
?>
    <option value="<?php echo $row_lect11['id_lect']?>"><?php echo
$row_lect11['name_lect']?></option>
    <?php
} while ($row_lect11 = mysql_fetch_assoc($lect11));
$rows = mysql_num_rows($lect11);
if($rows > 0) {
    mysql_data_seek($lect11, 0);
    $row_lect11 = mysql_fetch_assoc($lect11);
}
}

```

```

?>
    </select>
    </label></td>
<td width="36%"></td>
</tr>

<tr>
<td width="18%"><label>Quantity Students</label></td>
    <td> <input name="textfield5" type="text" value="<?php echo $row_stud10['count(
name_stud)']; ?>" size="5" />
        <label>
        <select name="select2">
            <option value="value">Students</option>
            <?php
do {
?>
            <option value="<?php echo $row_stud11['id_stud']?>"><?php echo
$row_stud11['name_stud']?></option>
            <?php
} while ($row_stud11 = mysql_fetch_assoc($stud11));
$rows = mysql_num_rows($stud11);
if($rows > 0) {
    mysql_data_seek($stud11, 0);
    $row_stud11 = mysql_fetch_assoc($stud11);
}
?>
    </select>
    </label></td>
<td> </td>
</tr>
</table>
</form>
<p>
<div id="apDiv1">
<p>

```

```

<?php
// (FCChart Begin) #FusionChart1
renderFusionChart($dFCFusionChart1, 400, 400);
// #FusionChart1 (FCChart End)
?>

</p>
<p>
</p>
</div>
<p align="(No)"
onclick="MM_showHideLayers('apDiv4','show','apDiv1','hide')">&nbsp;</p>
<p onclick="MM_showHideLayers('apDiv4','show','apDiv1','hide')">&nbsp;</p>
<p onclick="MM_showHideLayers('apDiv4','show','apDiv1','hide')">&nbsp;</p>
<div id="apDiv2" onclick="MM_showHideLayers('apDiv5','show','apDiv1','hide')">Count
Techers</div>
<p>
</p>
</body>
</html>
<?php
mysql_free_result($colege);
mysql_free_result($Recordset1);
mysql_free_result($dep10);
mysql_free_result($dep11);
mysql_free_result($lect10);
mysql_free_result($lect11);
mysql_free_result($stud10);
mysql_free_result($stud11);
mysql_free_result($Charts);
mysql_free_result($Gharttecher);
?>

```