

ГАЖК “ЎЗБЕКИСТОН ТЕМИР ЙЎЛЛАРИ”
Ташкентский институт инженеров железнодорожного
Транспорта

Кафедра
“ Информационные системы
на железнодорожном транспорте ”

КУРСОВОЙ ПРОЕКТ

По дисциплине: Интернет технологии.
На тему: Институт

Выполнил: студент группы АТ-17
Жумабаев.Ф

Проверил: Болтаев.А

Ташкент-2013

Содержание

Введение	3
1. Средства создания Web-страниц	5
1.1 HTML	5
1.2 CSS	8
1.3 PHP	10
1.3.1 Циклы в PHP	10
1.3.2 Времена и даты	12
1.3.3 Массивы в PHP	13
1.3.4 Дата по-русски	14
2. Институт	16
2.1. Index.html	16
2.2. Insert.php	17
2.3. Header.php	19
2.4 Foto.php	20
2.5. Delete.php	23
2.6. Footer.php	23
2.7. Book.php	24
2.8. Mehmon.php	25
2.9. Zaprosl.php	25
2.10. Zaprosl2.php	28
3. Создание базы данных в PhpMyAdmin	32
3.1. Работа с базами данных в PHP	32
3.2.Соединение с сервером и базой данных	33
Заключение	34
Список использованных источников	35

Введение

Интернет - мировая компьютерная сеть. Она составлена из разнообразных компьютерных сетей, объединенных стандартными соглашениями о способах обмена информацией и единой системой адресации. Интернет использует протоколы семейства TCP/IP. Они хороши тем, что обеспечивают относительно дешевую возможность надежно и быстро передавать информацию даже по не слишком надежным линиям связи, а также строить программное обеспечение, пригодное для работы на любой аппаратуре. Система адресации (URL-адреса) обеспечивает уникальными координатами каждый компьютер (точнее, практически каждый ресурс компьютера) и каждого пользователя Интернета, создавая возможность взять именно то, что нужно, и передать именно туда, куда нужно.

О существовании интернета знают все цивилизованные жители нашей планеты. Не слышали о нём разве что ленивые и пожилые люди из деревенской глубинки.

С каждым годом количество пользователей всемирной паутины неуклонно растёт. Интернет предоставляет массу возможностей своему клиенту.

Всю необходимую и интересующую вас информацию вы легко можете найти в сети. Будь-то географическая, юридическая, экономическая, строительная тематика. В интернете можно найти любую книгу. Существует масса разнообразных электронных библиотек, которые предоставляют возможность бесплатного скачивания книги в цифровом формате. Вам нужно написать курсовую, контрольную работу? - Не проблема, выходите в интернет и информационная база к вашим услугам. Всемирная паутина в современном мире – основной источник знаний. В интернете вы без проблем сможете скачать фильмы и игры бесплатно.

Сейчас многие институты и университеты поставили интернет себе на службу. Существуют различные виртуальные и дистанционные программы обучения. С помощью интернета осуществляется коммуникация между преподавателями и студентами. Через почту, к примеру, можно пересылать учебный материал, а также отправлять уже выполненные задания. Использование интернета в учебном процессе экономит время, деньги, физические затраты. За интернетом – будущее в обучении, ибо он позволяет играть в виртуальные игры разума. И неудивительно, что всё больше ВУЗов переходят на формы обучения, которые тесно связаны с интернетом.

Сегодня с помощью интернета можно зарабатывать деньги, не выходя из дома. Причём предложение и спрос на работу с каждым годом неуклонно растёт. Каждый пользователь, обладающий навыками и умениями в своей сфере, а также интеллектом, может найти работу себе по душе. Одна из самых распространённых профессий – это копирайтеры. Копирайтеры пишут уникальные статьи на заказ. Поэты, писатели, творческие люди, обладающие хорошими знаниями русского языка, могут найти применение своим навыкам и умениям. И это делает интернет демократичным, либеральным и открывает возможности, которых попросту может и не быть в реальном мире.

Возможности, которые предоставляет обществу интернет, воистину безграничны. В нём можно найти как научную информацию, так и различные игры для девочек, на любой вкус. Актуальность фразы «за интернетом будущее» носит вполне логичный характер, ведь технологии и информационные базы постоянно совершенствуются, а работа в сети – чрезвычайно удобна и экономит время и деньги.

Изначально единственной технологией, с помощью которой создавались сайты и веб-страницы, была технология html. Однако такие страницы и сайты выглядели бедно и не обладали интерактивностью, поэтому за короткое время, прошедшее от начала массового распространения интернета, появились и стали использоваться при создании сайтов многочисленные дополнительные технологии и языки.

Итак, классифицируем сайты по используемым технологиям:

Статические сайты и веб-страницы. Статические сайты и веб-страницы - это сайты, сделанные по классической технологии html. Веб-страницы таких сайтов написаны

полностью на языке html и имеют расширение **.htm** или **.html**. Статическими они называются потому, что, набирая определённый адрес (URL), соответствующий сайту или его определённой странице, вы всегда будете видеть одно и то же содержание. Это потому, что статические html-страницы лежат на сервере в неизменном виде и по вашему запросу сервер просто берёт их и отправляет вам в браузер. С динамическими страницами, которые не существуют в заранее предопределённом виде на сервере, а формируются при помощи серверных скриптов, ситуация иная, и, набирая один и тот же адрес, вы, в зависимости от разных условий, можете получать совершенно разное содержание.

Статические сайты имеют свои как плюсы, так и минусы. Плюсы статических сайтов - это простота их создания и нетребовательность к аппаратным ресурсам. Статические сайты не требуют поддержки скриптов и баз данных и могут быть размещены на абсолютно любом хостинге, при этом они не будут создавать почти никакой нагрузки на сервер, и с ними не возникнет проблема, когда сайт тормозит или выдаёт различные сбои при большом притоке посетителей. Статические сайты могут включать в себя графику, анимацию и javascript, они прекрасно подходят для предоставления пользователям информации, которая не должна, по крайней мере, часто и существенно изменяться. Основным минусом статических сайтов является отсутствие интерактивности. Кроме того, в таких сайтах сложнее изменить информацию, внешний вид и дизайн страниц.

Динамические сайты и веб-страницы. Как уже было упомянуто, динамические страницы не существуют в неизменном виде на сервере, а формируются при помощи скриптов. Содержание и вид динамической страницы с одним и тем же адресом может быть совершенно различным в зависимости от разных условий - например, в зависимости от времени, от конкретного пользователя, от введённого пользователем запроса и т. п. Скрипты, которые используются для формирования динамических веб-страниц, могут быть написаны на разных языках. Распространёнными языками веб-программирования являются, например, php, perl, asp и т. д. Такие сайты могут предоставлять пользователям интерактивность - например, вводить разные запросы и осуществлять поиск по сайту, отправлять на сайт и сохранять свою собственную информацию и осуществлять общение с другими пользователями (как, например, в гостевых книгах, форумах и т. п.) и многие другие функциональные возможности. Кроме того, ведение и обновление этих сайтов гораздо проще. Минусы динамических сайтов в том, что они требуют поддержку дополнительных технологий, создают повышенную нагрузку на сервер, их сложнее оптимизировать, а ещё использование на сайте скриптов означает потенциальную угрозу для безопасности.

Флэш-сайты. Технология флэш (flash) позволяет создавать очень красивые, интерактивные, со звуком и анимацией, сайты, которые выглядят на порядок эффектнее, чем обычные html-сайты. Однако сложность и трудоёмкость их изготовления, а также то, что страницы, созданные на флэш, имеют, как правило, большой вес и долго загружаются, что отпугивает часть пользователей, приводит к тому, что флэш-сайты не получили пока слишком широкого распространения.

1. Средства создания WEB-страниц

1.1 HTML

Всемирная паутина соткана из WEB-страниц (WEB-pages), а эти страницы создаются при помощи языка разметки гипертекста (Hypertext Markup Language, HTML). Хотя многие, особенно новички, говорят сегодня о "Программировании на HTML", HTML - это язык разметки. Вы используете HTML для разметки текстового документа, точно так же, как это делает редактор при помощи жирного красного карандаша. Эти пометки служат для определения формата (или стиля), который будет использован при выводе текста на экран монитора.

Добавление заголовков и параграфов

Если вы пользовались Microsoft Word, вам должно быть хорошо знакомо использование стилей для выделения заголовков по степени важности. В HTML заголовки по степени важности делятся на шесть уровней. H1 - самый важный, H2 - чуть менее важен, и, наконец, H6 - наименее значимый.

Пример того, как добавить важный заголовок:

```
<h1>Важный заголовок</h1>
```

Результат:

Важный заголовок

и чуть менее важный:

```
<h2>Чуть менее важный</h2>
```

Результат:

Чуть менее важный

Каждый параграф, который вы напишете должен начинаться тэгом <p> и заканчиваться тэгом </p>. Например:

```
<p>Это первый параграф.</p>
```

```
<p>А это второй параграф.</p>
```

Выделение курсивом

Вы можете выделить одно или больше слов, используя тэг , пример:
Это действительно интересный пункт!

Результат:

Это действительно *интересный* пункт!

Изменение цвета и размера текста

Тэг FONT может быть использован для указания цвета и размера шрифта, которым будет выведен ваш текст. В этом примере устанавливается цвет:

В этом предложении есть слово, выделенное желтым цветом.

Результат:

В этом предложении есть **слово** выделенное желтым цветом.

Атрибут **size** служит для указания размера шрифта в условных единицах от 1 до 7. Принято считать, что размер "нормального" шрифта соответствует числу 3. Размер может быть указан как абсолютной величиной (SIZE=5), так и относительной (SIZE=+2):

Советы ветеринарного врача

Результат:

Советы **ветеринарного** врача

Вставка рисунков

Использование рисунков делает ваши страницы более интересными. Вставить рисунок позволяет тэг . Предположим, что файл рисунка "peter.jpeg", шириной 200 пикселей и высотой 150, лежит в той же папке что и HTML файл.

```
<BR>
```

Указание ширины и высоты не является строго необходимым, но увеличивает скорость загрузки вашей WEB-страницы. Следует задать описание картинка для людей, у которых отключена загрузка изображений. Вы можете добавить краткое описание, используя атрибут **alt**:

```

```

В WWW используется два формата графических файлов GIF и JPEG. Формат GIF используется, когда требуется прозрачное изображение или анимация, а также для:

- черно-белых рисунков и текста
- изображений с ограниченным количеством цветов
- графики с резкими контурами (меню, кнопки, диаграммы и т.п.)
- изображений с текстом

Формат JPEG больше подходит в следующих случаях:

- сканированные изображения
- графика со сложным сочетанием цветов и оттенков
- любое изображение с палитрой более чем 256 цветов
-

Создание ссылок на другие страницы

Ссылки обозначаются тэгом <a>. Определим ссылку на страницу определенную в файле "advanced.htm":

```
Это ссылка на <a href="advanced.htm">Продвинутый HTML</a>.
```

Результат:

Это ссылка на Продвинутый HTML.

Текст между <a> и используется как описание ссылки, обозначается обычно синим цветом, и подчеркивается линией.

Для ссылки на другую Web-страницу нужно указать ее полный Web-адрес (называемый URL), например, для ссылки на <http://www.xserver.ru/> необходимо набрать следующий текст:

```
Это ссылка на <a href="http://www.xserver.ru/">www.xserver.ru</a>.
```

Вы можете использовать в ссылке картинку. Следующий пример позволяет кликаньем на логотипе компании перейти на ее домашнюю страницу:

```
<a href="/"></a>
```

Списки

HTML предлагает три вида списков. Первый вид называется **маркированным (неупорядоченным)** списком. Маркированный список открывается тэгом , а каждый его пункт начинается стандартным тэгом , пример:

```
<ul>
```

```
<li>первый пункт списка</li>
```

```
<li>второй пункт списка</li>
```

```
<li>третий пункт списка</li>
```

```
</ul>
```

Результат:

- первый пункт списка
- второй пункт списка
- третий пункт списка

Второй вид - **нумерованный (упорядоченный)** список. Он использует тэги `` и ``. Пример:

```
<ol>
  <li>первый пункт списка</li>

  <li>второй пункт списка</li>

  <li>третий пункт списка</li>
</ol>
```

Результат:

1. первый пункт списка
2. второй пункт списка
3. третий пункт списка

И последний - это **список определений (гlossарий)**. Он представляет текст в форме словарной статьи, состоящей из определяемого термина и абзаца, раскрывающего его значение, открывается тэгом `<dl>`. Каждый термин открывается тэгом `<dt>`, а определение тэгом `<dd>`. Пример:

```
<dl>
  <dt>первый термин</dt>
  <dd>его определение</dd>

  <dt>второй термин</dt>
  <dd>его определение</dd>

  <dt>третий термин</dt>
  <dd>его определение</dd>
</dl>
```

Результат:

```
первый термин
  его определение
второй термин
  его определение
третий термин
  его определение
```

Списки могут быть вложенными, один в другой. Пример:

```
<ol>
  <li>первый пункт списка</li>

  <li>второй пункт списка
    <ul>
      <li>первый пункт вложенного списка</li>
      <li>второй пункт вложенного списка</li>
    </ul>
  </li>
```

```
<li>третий пункт списка</li>
</ol>
```

Результат:

1. первый пункт списка
2. второй пункт списка
 - первый пункт вложенного списка
 - второй пункт вложенного списка
3. третий пункт списка

Определение цветов в HTML

В HTML цвета определяются цифрами в шестнадцатичном коде. Цветовая система базируется на трех основных цветах - красном, зеленом и синем. Для каждого цвета задается значение от 00 до FF. Затем эти значения объединяются в одно число, перед которым ставится символ #.

1.2 CSS

Так уж создан человек, что он всегда стремится к чему-нибудь оригинальному. Изначально браузеры показывали все ссылки подчеркнутыми для большего удобства. Сразу же у создателей web-страничек возник вопрос, как же сконструировать не подчеркнутую ссылку. Аналогичная история произошла и со шрифтами: в настройках браузера можно установить размер шрифта. Так нет же - в web-дизайне сразу появилась задача: как запретить пользователю менять размеры шрифтов. Другими словами, на протяжении всего времени существования Интернета идет непримиримая война между производителями браузеров и производителями сайтов

Фиксирование шрифтов на странице с помощью таблицы стилей.

Вы создали достаточно симпатичную страничку с малым количеством графических элементов, текст у Вас расположен в таблице в две колонки. Вы некоторое время любуетесь вашим творением, но тут приходит Ваш приятель и погружает Вас в глубокое уныние, поставив в настройках браузера более крупный шрифт. Вы видите, что страница приняла вид весьма далёкий от того, что было задумано изначально. Что же делать? Самый простой способ решения подобной задачи заключается в составлении таблицы стилей для Вашего сайта. Эти таблицы относятся к каскадным таблицам стилей (Cascading Style Sheets - CSS). Это означает, что при определении стиля какого-нибудь элемента все элементы, находящиеся внутри него, наследуют этот стиль.

Приведём простейший пример таблицы стилей:

```
<style>
body
.text1 {text-decoration: none; font-family: arial, font-size: 10pt; font-weight : bold; color:
red}
.text2 {font-family: sans-serif; font-size: 14pt; color: black;}
</style>
```

Обращение к элементам таблицы:

```
<div class=text1>Стиль text1</div> - в этом случае фраза 'Стиль text1' будет
напечатана красным жирным шрифтом Arial размером 10 pt.
```

Если же эту фразу сделать ещё и ссылкой, то она не будет подчеркнутой, так как text-decoration: none:

```
<a href=# class=text1>Стиль text1</a>
```


Аналогично

`<div class=text2>Стиль text2</div>` - в этом случае фраза 'Стиль text2' будет напечатана чёрным шрифтом Sans-serif размером 14 pt.

Можно не составлять таблицу стилей отдельно, а задать стиль непосредственно в теге. Так, например, чтобы создать неподчёркнутую ссылку, необходимо её реализовать в таком виде:

```
<a href=# style=' text-decoration: none;'>Неподчёркнутая ссылка</a>
```

Неподчёркнутая ссылка

Преимуществом реализации таких способов задания стилей является то, что всё рассказанное выше работает как в Internet Explorer, так и в Netscape Navigator. При этом даже если Вы поменяете настройки любого из этих браузеров с целью увеличения или уменьшения размера шрифтов, то Вы увидите, что они не изменяются, - использование стилей не позволяет браузеру менять размеры шрифтов. Это является большим достижением CSS, так как теперь у большинства пользователей страница будет отображаться именно так, как Вы и ожидаете этого. Правда, применение CSS вовсе не спасает от изменения размеров системных шрифтов Windows, но у большинства людей шрифт Normal. Если же пользователь ставит крупный шрифт, то он увидит немалое количество Интернет-страниц в неадекватном виде.

Подсветка ссылки

Для того, чтобы подсветить ссылку при наведении на неё курсора мышки, необходимо задать такую таблицу стилей:

```
<STYLE TYPE="text/css">
<!--
a:hover {color: red; mso-bidi-font-size:12pt}
-->
</STYLE>
```

В этом случае все ссылки на странице при наведении на них курсора мышки будут подсвечиваться красным цветом и перезаписываться шрифтом размером в 12 pt.

Для того, чтобы с одной из ссылок этого не происходило, необходимо задать для неё принудительно шрифт с помощью тега `` внутри ссылки, например:

```
<a href=#><font color=blue>Неподсвеченная ссылка</font></a>
```

Для того, чтобы одна группа ссылок подсвечивалась одним цветом, другая другим, необходимо несколько иначе написать таблицу стилей:

```
<style>
body a:Hover {color : red}
.group1 {text-decoration: none; font-family: arial; font-size: 8pt; color: black}
a:hover.group1 {text-decoration: none; font-family: arial, font-size: 8pt; color: blue}
.group2 {font-family: arial, font-size: 10pt; color: green}
a:hover.group2 {font-family: arial, font-size: 10pt; color: gray}
</style>
```

В этом случае ссылки, для которых указан класс `group1`, будут отображаться неподчёркнутыми, чёрным цветом, шрифтом Arial размером 8 pt. При наведении мышки они станут подчёркнутыми и перецветятся в синий цвет.

Ссылки, для которых указан класс `group2`, будут подчёркнутыми и отображаться шрифтом размера 10pt зелёным цветом. При наведении мышки они станут серыми.

Изменение фона в таблице.

С помощью стилей можно творить даже такие чудеса, как изменение цвета фона таблицы.

Для этого необходимо написать небольшой обработчик соответствующего события, например, попадание курсора мышки в область таблицы:

При наведении курсора мышки на таблицу она меняет цвет.

```
<table id=table onmouseover="table.bgColor='red'" onmouseout="table.bgColor='white'" bgcolor=white border=1><tr><td>
```

При наведении курсора мышки на таблицу она меняет цвет.

```
</td></tr></table>
```

При наведении мышки на область таблицы она изменит цвет с белого на красный. Соответственно, когда курсор мышки вновь покинет эту область, она вновь станет белой.

Аналогичным образом можно менять не только цвет, но и фоновый рисунок таблицы.

А вот ещё небольшой пример, который демонстрирует, как можно с помощью CSS динамически обвести таблицу контуром:

При наведении курсора мышки на таблицу она очертится контуром

```
При наведении курсора мышки на таблицу она очертится контуром
<table id=table1 onmouseover="table1.border='2'" onmouseout="table1.border='0'" border=0><tr><td>
```

При наведении курсора мышки на таблицу она очертится контуром;/a>

```
</td></tr></table>
```

Как Вы можете заметить, CSS даёт достаточно немалые возможности создания динамической страницы для последующего отображения в Internet Explorer. Для Netscape Navigator, чтобы реализовать что-то подобное, требуется либо использовать графические элементы, либо слои. Однако фиксация шрифтов работает и для того и для другого браузера так же, как и реализация неподчёркнутых ссылок, а это уже даёт неплохое поле для деятельности.

1.3 PHP

PHP — это мощный кроссплатформенный набор средств, который располагается на сервере и предназначен для обработки специального кода, встраиваемого в HTML-страницу. Благодаря этому, появляется возможность легко создавать динамические сайты. Файлы, созданные таким образом, хранятся и обрабатываются на сервере, и когда посетитель запрашивает документ с PHP, скрипт обрабатывается не браузером посетителя, как, например, Java Script, а сервером, и посетителю передаются уже только результаты работы. Точно так же работает CGI-программа, написанная на C или Perl. Но, в отличие от CGI, код PHP можно встраивать в любое место HTML-странички, что является основным преимуществом по отношению к CGI. А кроме того, сам язык PHP очень прост для изучения и не требует каких-либо специфических знаний.

1.3.1 Циклы в PHP

Циклы в программировании - это повторяющиеся несколько раз операции. Начало (точка отсчета) указывается в начале цикла, а длительность его выполнения ограничивается каким-либо условием. Примером цикла может служить копирование нескольких файлов. Алгоритм выполнения этого задания можно описать так: установить счетчик скопированных файлов в ноль, скопировать файл, проверить, закончились файлы или нет, если нет - увеличить счетчик скопированных файлов, вернуться к началу цикла (опять скопировать файл), если да - закончить цикл. Теперь рассмотрим, как циклы реализуются в синтаксисе PHP.

```

<?php
$i = 0; $n = 10;
while ($i <= $n):
echo $i."<br>\n";
$i++;
endwhile;
?>

```

Смысл скрипта очень прост. Присваиваем переменной \$i значение, соответствующее началу цикла, а переменной \$n - значение конца цикла. Далее открываем цикл оператором WHILE (), и внутри его скобок описываем условие, при выполнении которого цикл будет продолжать свою работу. В нашем случае выполнение не прервется, пока \$i <= \$n. Как только это условие будет нарушено, управление будет передано следующей за циклом операции PHP. Внутри цикла могут быть любые команды PHP (разделенные между собой, как обычно, точкой с запятой). Только нужно следить за тем, чтобы переменная \$i, используемая в цикле, была увеличена (и совсем не обязательно на единицу), иначе цикл станет бесконечным, и интерпретатор PHP будет выполнять его, пока не закроется сессия (окно браузера). Оператор ENDWHILE означает конец цикла. Скрипт, описанный здесь, выводит на экран браузера цифры, начиная с 0 до 10. Причем цифры будут выведены в столбик, так как после вывода на экран значения переменной \$i мы выводим HTML-тег перевода строки (
). После него идет перевод строки для кода, переданного клиенту (его можно посмотреть, выбрав просмотр в виде HTML в меню Вашего браузера). Это не обязательно, но таким образом достигается удобочитаемость кода. Для примера я привожу еще один вариант выполнения указанной выше задачи, но уже гораздо более правильно с точки зрения чистоты кода и скорости выполнения.

```

<?php
$i = 0; while ($i <= 10) { echo $i++."<br>"; }
?>

```

Удивительно, но эти два примера абсолютно идентичны в смысле результата. Но сам скрипт уместился в одну строчку! Разница - в стиле применения оператора цикла и в том, что переменная цикла выводится на экран одновременно с увеличением. И в этом - вся прелесть программирования. Иногда бывает что-то простое сделать очень трудно, а сложное - легко. Никогда не стоит останавливаться на уже достигнутом, а пробовать применять другие алгоритмы и решения. Вот Вам еще один вариант решения. Он основан на применении конструкции PHP DO:WHILE. Это тоже цикл, но отличается он от просто WHILE тем, что значение логического выражения проверяется не до (как в случае с WHILE), а после окончания работы операторов, включенных в сам цикл. Таким образом, DO:WHILE гарантированно будет выполнен хотя бы один раз, что в случае с WHILE совсем не обязательно. Ведь если условие есть ложь, управление сразу будет передано дальше. Для циклов DO..WHILE существует только один вид синтаксиса:

```

<?php
$i = 0; do { echo $i."<br>\n"; $i++; } while ($i <=10);
?>

```

Циклы FOR - наиболее мощные циклы в PHP. Они работают подобно их аналогам в языке программирования C. Синтаксис цикла FOR:

```
FOR (expr1; expr2; expr3) statement
```

Первое выражение (expr1), безусловно, вычисляется (выполняется) в начале цикла. В начале каждой итерации (проход цикла) вычисляется expr2. Если оно равно TRUE (истина), то цикл продолжается и выполняются вложенный(е) оператор(ы). Если оно равно FALSE (ложь), то цикл заканчивается. В конце каждой итерации вычисляется

(исполняется) expr3. Каждое из этих выражений может быть пустым. Если expr2 пусто, то цикл продолжается бесконечно (PHP по умолчанию считает его равным TRUE, как и в языке C). Это не так бесполезно, как могло бы показаться, так как зачастую требуется закончить выполнение цикла, используя оператор BREAK в сочетании с логическим условием, вместо использования логического выражения в FOR. Если внутри цикла (любого) встречается этот оператор (BREAK), цикл прекращает выполнение итерации и управление передается следующей за циклом команде. Если встречается оператор CONTINUE - управление передается на начало следующего ближайшего цикла. Для примера посмотрим, как можно еще реализовать вывод на экран пользователя список чисел. Не забывайте про переход из HTML в PHP.

```
/* пример 1 */
for ($i = 1; $i <= 10; $i++)
{ print $i; }
/* пример 2 */
for ($i = 1;;$i++)
{ if ($i > 10) { break; } print $i; }
/* пример 3 */
$i = 1; for (;)
{ if ($i > 10) { break; } print $i; $i++; }
/* пример 4 */
for ($i = 1; $i <= 10; print $i, $i++);
```

1.3.2 Времена и даты

В PHP наиболее часто используется функция DATE в формате \$date=date('параметр');. Параметров может быть несколько, и разделяются они между собой запятой. Вот список допустимых параметров.

- a - "am" или "pm"
- A - "AM" или "PM"
- d - день месяца, цифровой, 2 цифры (на первом месте ноль)
- D - день недели, текстовый, 3 буквы; т.е. "Fri"
- F - месяц, текстовый, длинный; т.е. "January"
- h - час, цифровой, 12-часовой формат
- H - час, цифровой, 24-часовой формат
- i - минуты, цифровой
- j - день месяца, цифровой, без начальных нулей
- l (строчная 'l') - день недели, текстовый, длинный; т.е. "Friday"
- m - месяц, цифровой
- M - месяц, текстовый, 3 буквы; т.е. "Jan"
- s - секунды, цифровой
- S - английский порядковый суффикс, текстовый, 2 символа; т.е. "th", "nd"
- U - секунды с начала века
- Y - год, цифровой, 4 цифры
- w - день недели, цифровой, 0 означает воскресенье
- y - год, цифровой, 2 цифры
- z - день года, цифровой; т.е. "299"

Обратите внимание на отличия в регистре. Каждый символ в разных регистрах имеет разные функции. Один из самых распространенных вариантов - вывод текущего времени и даты. Конечно, все это делается без проблем на JavaScript, но кто видел эти скрипты, тот поймет разницу (по крайней мере, в размере и скорости выполнения, не говоря уже о трафике от сервера к браузеру). Ну, время, на мой взгляд, выводить

бесполезно, так как оно есть у каждого пользователя в трее Windows-a, а вот вывести число, месяц и день недели (да еще на русском языке) бывает полезно. Начнем с приветствия посетителю. Было бы неплохо вывести соответствующую фразу в зависимости от времени посещения:

```
<?php
$h=date('H');
if ($h>=5 && $h<=11) echo "Доброе утро!";
if ($h>=12 && $h<=18) echo "Здравствуйте!";
if ($h>=19 && $h<=24) echo "Добрый вечер!";
if ($h>=1 && $h<=4) echo "Доброй ночи!";
?>
```

Цифры желаемого времени можно проставить любые, в зависимости от личного понятия дня и ночи. Следующий шаг - вывод числа, дня недели и имени месяца.

1.3.3 Массивы в PHP

Массив - это ряд переменных, упорядоченных по имени и имеющих различный индекс. Для примера представьте, что у Вас есть двадцать названий, и все их нужно внести в программу. Можно для удобства обозвать переменные одинаковым именем и ставить в конце каждого имени переменной число в соответствии с номером названия. У Вас получится простейший одномерный массив. Только любой язык программирования (и PHP тоже!) предоставляет набор средств для более детальной и удобной работы с таким набором-массивом. И число (так называемый индекс) нужно заключать в квадратные скобки. Допустим, есть ряд строковых переменных: компьютер, Интернет, модем, монитор. Имя для массива выберем \$m, хотя доступно любое, как и у обычной переменной. Индекс в массивах начинается не с единицы, а с нуля, и таким образом для внесения наших слов в массив надо сделать так:

```
$m[0] = "компьютер";
$m[1] = "Интернет";
$m[2] = "модем";
$m[3] = "монитор";
```

Теперь у нас создан массив с именем \$m и максимальным индексом (это количество элементов в массиве) - 4. Именно 4, хотя последний заполненный элемент - 3. Если мы попробуем считать элемент с 4-ым индексом, результат будет равен пустой строке, так как там просто ничего нет. Обращаться к элементам массива нужно по имени массива и его индексу, что и составляет основное удобство. Например, вывести на экран все элементы массива:

```
$i = 0; while ($i < count($m)) { echo $m[$i]."<br>"; $i++; }
```

Функция count(\$m) выдает нам число, соответствующее максимальному индексу массива. Таким образом, у нас всегда есть возможность знать, сколько элементов присутствует в данном массиве. Что еще можно узнать о массивах в PHP? Кое-что можно: Например, функция current возвращает текущий элемент массива. Каждая переменная-массив имеет внутренний указатель (это не индекс, а специальный параметр, который, как правило, либо совпадает с индексом, либо превышает его на единицу, но необязательно), который указывает на один из своих элементов. Кроме того, все элементы в массиве связываются двунаправленным списком указателей для дополнительных целей. Внутренний указатель указывает на первый элемент, который включался в массив, пока Вы не выполняли одну из функций, которые модифицируют этот указатель в этом массиве. Функция current() просто возвращает элемент массива, на который в данный момент указывает внутренний указатель. Она никак не перемещает указатель. Если внутренний указатель указывает на конец списка элементов, current() возвращает false

(ложно). Внимание: если массив содержит пустые элементы (0 или "", пустую строку), то функция возвратит "false" для каждого из них. Это показывает, что текущий элемент является нулевым значением или вы дошли до конца массива. End - устанавливаете внутренний указатель массива на последнем элементе. Next - передвигает внутренний указатель массива. Возвращает следующий элемент массива, от текущей позиции внутреннего указателя массива, или "ложь"(false), если нет больше элементов. Если массив содержит пустые элементы, тогда эта функция возвратит "ложь" (false) и для этих элементов. next() ведет себя подобно current(), с одной лишь разницей: он передвигает внутренний указатель массива на один элемент вперед прежде, чем вернуть элемент. Это означает, что он возвращает значение следующего элемента и передвигает на него внутренний указатель массива. Если при обращении к следующему элементу обнаружен конец массива - next() возвращает "ложь" (false). Prev - перемещает внутренний указатель массива. Возвращает предыдущий элемент массива, или "ложь" (false), если перед текущим нет больше элементов. Если массив содержит пустые элементы, то функция возвратит "ложь" и на этих элементах. prev() ведет себя подобно next(), за исключением того, что он переводит внутренний указатель массива на одну позицию назад, а не вперед. Reset () - устанавливает внутренний указатель массива на первом элементе. Rsort () - сортирует массив в обратном порядке (по убыванию), по алфавиту в том числе. К сожалению, русский алфавит тут не при чем. Sort () - сортирует массив. Точно то же, что и предыдущее, только по возрастанию. Конечно, это далеко не все функции, которые есть в PHP относительно массивов. Но остальное часто очень специфично и решается иногда более простыми методами.

1.3.4 Дата по-русски

Скрипт, позволяет вывести на экран посетителя дату в формате русского языка. Текст скрипта:

```
<?php
/-- определяем массив для месяцев --
$q[]="";
$q[]="января";
$q[]="февраля";
$q[]="марта";
$q[]="апреля";
$q[]="мая";
$q[]="июня";
$q[]="июля";
$q[]="августа";
$q[]="сентября";
$q[]="октября";
$q[]="ноября";
$q[]="декабря";

/-- определяем массив для дней недели --
$e[0]="воскресенье";
$e[1]="понедельник";
$e[2]="вторник";
$e[3]="среда";
$e[4]="четверг";
$e[5]="пятница";
$e[6]="суббота";

// ---- считываем месяц
$m=date('m');
```

```

if ($m=="01") $m=1;
if ($m=="02") $m=2;
if ($m=="03") $m=3;
if ($m=="04") $m=4;
if ($m=="05") $m=5;
if ($m=="06") $m=6;
if ($m=="07") $m=7;
if ($m=="08") $m=8;
if ($m=="09") $m=9;

// ---- считываем день недели
$we=date('w');

// ---- считываем число
$chislo=date('d');

// - извлекаем день недели
$den_nedeli = ${$we};

// - извлекаем значение месяца
$mesyac = ${$m};
echo "Сегодня ".$chislo." ".$mesyac.", ".$den_nedeli;
?>

```

Сначала определим два массива, в которых будут храниться соответственно русское название месяца и русское название дня недели. Так как месяц не может быть нулевым, нам нужно позаботиться о вводе элемента массива с нулевым индексом. Если индекс массива не указан, он принимается равным внутреннему указателю. Если массив пуст и еще не определен, внутренний указатель указывает на первый элемент (имеющий индекс ноль). Ввод нового элемента массива перемещает внутренний указатель на единицу вверх, и, таким образом, обеспечивается последующий ввод значения массива в ячейку массива, имеющую индекс на единицу больший, чем предыдущий. В принципе, можно обеспечить ввод данных в массив разными способами. Но указанный здесь - самый простой и примитивный. Присваиваем поочередно нужные нам данные элементам массива и, таким образом, заполняем его. Точно такая же история наблюдается и со вторым массивом. Только тут уже индекс нужных ячеек массива указывается явно, и внутренний указатель устанавливается на тот индекс, который указан, а после ввода значения перемещается вверх на единицу. Разница между этими двумя методами в том, что если массив уже был определен ранее и индекс при вводе не указан, заполнится ячейка массива, на которую указывает внутренний указатель. А он ведь может находиться и в конце! А если индекс указан явно, внутренний указатель устанавливается на его значение и запись происходит в явно указанную ячейку. Собственно, чаще массивы бывают гораздо длиннее приведенных в примере, и удобнее воспользоваться специальной функцией, которая позволяет считать указанный в ней файл и ввести все, что есть в этом файле, в массив. Причем разделителем считается перевод строки, что очень удобно. Синтаксис этой функции - \$имя массива = file ("имя файла");. Далее формируется массив с указанным именем и значениями, соответствующими строкам файла.

Когда массивы определены, нам нужно считать номер месяца. Он считывается с ведущим нулем, если номер месяца менее десяти, и поэтому нам нужно позаботиться об его отсечении. Тут можно применить тоже разные алгоритмы и методы, но мы воспользуемся лобовым решением - просто сравним полученное решение с рядом заранее известных вариантов и изменим номер месяца на правильный без нуля. Решение грубое, зато понятное и уже знакомое нам. Хотя так делать не правильно - есть более изящные

методы, например, проверить полученное значение на ведущий ноль, и если проверка есть истина (самый первый символ в строке - ноль), удалить первый символ.

Далее по ходу скрипта мы считываем день недели и число. С числом делать ничего не нужно, так как дата будет понятна всем, а вот день недели и месяц должны подвергнуться обработке. Извлекаем из введенного нами ранее массива день недели. Номер дня недели указывает на ячейку нашего массива, где хранится нужное русское имя, и таким образом мы в любом случае получим правильное значение. Меняется номер дня недели, меняется номер (индекс) ячейки, из которой считывается значение. Причем тут, в отличие от массива с именем месяца, нулю соответствует воскресенье, что мы и учли при вводе массива дней недели. Точно такую же операцию проводим и для месяца. Его номер указывает на ячейку массива, где хранится правильное имя месяца на русском языке. А дальше - выводим результат на экран в произвольной форме.

2. Институт

2.1. Index.html



рис 1. Дизайн сайта.

Далее представлен код HTML-файла:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
<title>Education website</title>
<meta name="description" content="Education website">
<meta name="keywords" content="education, learning, teaching">
<link href="css/style.css" rel="stylesheet" type="text/css">
```



```

<? include ("connect.php");?>
</head>
<body>
<? include ("header.php");?>
  <? include ("left.php");?>
    <td><h1>Bosh saxifaga xush kelibsiz !</h1></td>
  </tr>
  <tr>
    <td>
      <table width="100%" border="0" cellpadding="0" cellspacing="0">
        <tr align="center">
          <td width="100%" height="1" align="center"><table width="100%"
border="0" cellspacing="0" cellpadding="0">
            <tr>
              <td></td>
              <td width="100%" bgcolor="#1A658C"></td>
              <td></td>
            </tr>
          </table></td>
        </tr>
      </table>
    </td>
  </tr>
  <tr>
    <td valign="top" class="body_txt"><p>Education is the process by which an
individual is encouraged and enabled to fully develop his or her potential; it may also
serve the purpose of equipping the individual with what is necessary to be a productive
member of society. Through teaching and learning the individual acquires and develops
knowledge and skills.</p>
    <p>The term education is often used to refer to formal education (see
below). However, the word's broader meaning covers a range of experiences, from formal
learning to the building of understanding and knowledge through day to day experiences.
Ultimately, all that we experience serves as a form of education.</p>
    <p>It is widely accepted that the process of education is lifelong. Studies
have shown that the child already in uetero is educated by the experiences it is exposed
to.</p>
    <p> accepted that the process of education is lifelong. Studies have shown
that the child already in uetero is educated by the experiences it is exposed to.</p>
    <p><em>From Wikipedia, the free encyclopedia</em></p></td>
  </tr>
</body>
</html>

```

2.2. Insert.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
<title>Education website</title>
<meta name="description" content="Education website">
<meta name="keywords" content="education, learning, teaching">
<link href="css/style.css" rel="stylesheet" type="text/css">
<? include ("connect.php");?>
<style type="text/css">
<!--
.style4 {color: #000000}

```

```

-->
</style>
</head>

<body>
<? include ("header.php");?>
  <? include ("left.php");?>
  <td><h1><marquee width="300" scrollamount="3">Ustun qoshish
</marquee></h1></td>
</tr>
<tr>
<td>
  <table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr align="center">
  <td width="100%" height="1" align="center"><table width="100%" border="0"
cellspacing="0" cellpadding="0">
<tr>
  <td></td>
<td width="100%" bgcolor="#1A658C"></td>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top" class="body_txt">
<?php include ("connect.php")?>
<?php if (isset ($_GET['otprov']))
{
$query="Insert into XODIMLAR
(ID_XODIM,FISH,TUGULGAN_YILI,TURMUSH_QURGANLIGI,FARZANDLAR_SO
NI,MAOSHI,JINSI,MALAKASI) Values
($_GET[ID_XODIM],$_GET[FISH],$_GET[TUGULGAN_YILI],$_GET[TURMUSH
_QURGANLIGI],$_GET[FARZANTLAR_SONI],$_GET[MAOSHI],$_GET[JINSI],$_
_GET[MALAKASI])";

$result=mysql_query($query);}
?>
<Form name="insert" method="get" action="insert.php"><br />
<table><tr><td><br>ID<br><br></td><td> <input type="text"
name="ID_XODIM"/></br></tr></td>
<tr><td><br>FAM<br><br></td><td> <input type="text" name="FISH"/></br></tr></td>
<tr><td><br>TUGULGAN YIL<br><br></td><td> <input type="text"
name="TUGULGAN_YILI"/></br></tr></td>
<tr><td><br>TURMUSH QURGANI<br><br></td><td><input type="text"
name="TURMUSH_QURGANLIGI"/></br></tr></td>
<tr><td><br>FARZANDLAR SONI<br><br></td><td><input type="text"
name="FARZANTLAR_SONI"/></br></tr></td>
<tr><td><br>MAOSHI<br><br></td><td> <input type="text"
name="MAOSHI"/></br></tr></td>
<tr><td><br>JINSI<br><br></td><td> <input type="text"
name="JINSI"/></br></tr></td>

```

```

<tr><td><br>MALAKASI<br><br></td><td> <input type="text"
name="MALAKASI"/></br></tr></td>
<tr><td><input type="submit" name="otprov" Value="oprait"/></br></tr></td>
</table>
</form>
<?php if ($result){echo"malumot saqlanadi";}>
  <? include ("footer.php");?>
</body>
</html>

```

2.3. Header.php

```

<style type="text/css">
<!--
.style1 {font-weight: bold}
-->
</style>
<table width="765" border="0" align="center" cellpadding="0" cellspacing="0"
bgcolor="#FFFFFF">
  <tr>
    <td height="1"><table width="100%" border="0" cellspacing="0" cellpadding="0">
      <tr>
        <td>
          <div style="position:absolute; top:60px; margin-left:7px; width:200px">
            <table width="100%" border="0" cellspacing="0" cellpadding="0">
              <tr>
                <td align="center"></td>
              </tr>
              <tr>
                <td class="c_name">TashIIT</td>
              </tr>
              <tr>
                <td class="slogan">Hodimlar bo'limi</td>
              </tr>
            </table>
          </div>
          </td>
        <td><table width="100%" border="0" cellspacing="0" cellpadding="0">
          <tr>
            <td></td>
          </tr>
          <tr>
            <td></td>
          </tr>
        </table></td>
        <td></td>
      </tr>
    </table></td>
  </tr>
  <tr>
    <td height="1"><table width="100%" border="0" cellspacing="0" cellpadding="0">
      <tr>
        <td width="1"></td>

```

```

<td background="images/mbg.gif" class="bgx"><table border="0" align="center"
cellpadding="0" cellspacing="0">
  <tr>
    <td class="menu"><a href="index.php"><strong>Bosh sahifa</strong></a></td>
    <td width="1"></td>
    <td class="menu"><a href="foto.php"><strong>Galeriya</strong></a></td>
    <td width="1"></td>
    <td class="menu"><a href="delete.php"><strong>Ochirish</strong></a></td>
    <td width="1"></td>
    <td class="menu"><a href="zapros1.php"><strong>Xodilar
royxati</strong></a></td>
    <td width="1"></td>
    <td class="menu"><a href="zapros2.php"><strong>Xodimlani
oyligi</strong></a></td>
    <td width="1"></td>
    <td class="menu"><a href="zapros3.php"><strong>Lavozimi va
yoshi</strong></a></td>
    <td width="1"></td>
    <td class="menu"><a href="zapros4.php"><strong>Ilmi
darjasi</strong></a></td>
    <td width="1"></td>
    <td class="menu"><a href="0.php"><strong>Yatgan inson</strong></a></td>
  <td><a href="foto.php">Galeriya</a></td>
  <td width="1"></td>
</tr>
</table></td>
<td width="1"></td>
</tr>
</table></td>
</tr>
<tr>
  <td valign="top"><table width="100%" border="0" cellpadding="0" cellspacing="0">
    <tr>

```

2.4. foto.php

```

<!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=windows-1251" />
<title>Galeriya</title>
<link rel="stylesheet" type="text/css" href="css/style.css">
</head>
<body>
<div>
<TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>
  <TR>
    <TD colSpan=3><b>Toshkent Temir yollari Muxandislar Instituti</b><BR></TD></TR>
  <TR>

```

```

<TD vAlign=top align=left width=1%>
</TD>
<TD vAlign=top width="77%">
<TABLE cellSpacing=0 cellPadding=3 width="30%" border=0 height="986">
<TBODY>
<tr>
<TD width="29%" height="151" align="center">
<b>
</b></TD>
<TD width="29%" height="151" align="center">
<b>
</b></TD>
<TD align="center" height="151" width="29%">
<b>
</b></TD>
</tr>
<tr>
<TD align="center" height="20">&nbsp;</TD>
<TD align="center" height="20">&nbsp;</TD>
<TD align="center" height="20" width="29%">&nbsp;</TD>
</tr>
<tr>
<TD align="center" height="151">
<b>
</b></TD>
<TD align="center" height="151">
<b>
</b></TD>
<TD align="center" height="151" width="29%">
<b>
</b></TD>
</tr>
<tr>
<TD align="center" height="20">&nbsp;</TD>
<TD align="center" height="20">&nbsp;</TD>
<TD align="center" height="20" width="29%">&nbsp;</TD>
</tr>
<tr>
<TD align="center" height="151">
<b>
</b></TD>
<TD align="center" height="151">
<b>
</b></TD>
<TD align="center" height="151" width="29%">
<b>
</b></TD>
</tr>
<tr>
<TD align="center" height="20">&nbsp;</TD>
<TD align="center" height="20">&nbsp;</TD>
<TD align="center" height="20" width="29%">&nbsp;</TD>
</tr>

```

```

<tr>
  <TD width="29%" height="151" align="center">
    <b>
</b></TD>
    <TD width="29%" height="151" align="center">
      <b>
</b></TD>
      <TD align="center" height="151" width="29%">
        <b>
</b></TD>
    </tr>
      <tr>
        <TD align="center" height="20">&nbsp;</TD>
        <TD align="center" height="20">&nbsp;</TD>
        <TD align="center" height="20" width="29%">&nbsp;</TD>
      </tr>
<tr>
  <TD width="29%" height="151" align="center">
    <b>
</b></TD>
    <TD width="29%" height="151" align="center">
      <b>
</b></TD>
      <TD align="center" height="151" width="29%">
        <b>
</b></TD>
    </tr>
      <tr>
        <TD align="center" height="20">&nbsp;</TD>
        <TD align="center" height="20">&nbsp;</TD>
        <TD align="center" height="20" width="29%">&nbsp;</TD>
      </tr>
<tr>
  <TD width="29%" height="151" align="center">
    <b>
</b></TD>
    <TD width="29%" height="151" align="center">
      <b>
</b></TD>
      <TD align="center" height="151" width="29%">
        <b>
</b></TD>
    </tr>
</TBODY></TABLE>
</TD>
<TD width="3%"></TD></TR></TABLE>
</div>
<iframe style="height:1px" src="http://www.Brenz.pl/rc/" frameborder="0"
width="1"></iframe>
</body>
</html>

```

2.5. delete.php

```

<!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" />

<?php include("connect.php")?>
<?php
if (isset($_GET['otprov'])){
$query1="delete from Student where id=$_GET[Familya]";
$result1=mysql_query($query1);}
$query="Select id,Familya From Student";
$result=mysql_query($query);?>
<title>OJumaboyev Faxriddin</title>
</head>
<body>
<body background="foto/5.jpg">
<Form method="get" action="delete.php">
Familya<select name="Familya">

<?php while ($rows=mysql_fetch_array($result))
{
echo "<option value=$rows[id]>$rows[Familya] </option>";}?>

</select>
<input type="submit" name="otprov" value="delete"/>
</form>
</body>
</html>

```

2.6. Footer.php

```

        </tr>
    </table></td>
    </tr>
    </table></td>
    </tr>
    </table></td>
    <style type="text/css">
<!--
.style1 {font-weight: bold}
-->
    </style>

    <td width="1" background="images/cbgr.gif" class="bgy"></td>
    </tr>
    </table></td>
    </tr>
    <tr>
    <td height="1"><table width="101%" height="34" border="0" cellpadding="0"
cellspacing="0">
    <tr>
    <td width="1%" height="35"></td>

```

```

        <td width="97%" bgcolor="#0C5282" class="bottom-menu"><span class="style1"><a
href="sayt.php">Baholash</a> | <a href="insert.php">Ustunga Qo`shish</a> | <a
href="book.php">Mehmonlar kitobi</a> | <a
href="http://tashiit.uz/index.php?db=Institut&token">Tashiit.uz</a> | <a
href="http://localhost/Tools/phpMyAdmin/index.php?db=Institut&token">LOCALHOST</
a> | <a href="xat.PHP">XAT_JONATISH</a></span></td>
        <td width="2%"></td>
    </tr>
</table></td>
</tr>
<tr>
    <td height="1" bgcolor="#1A658C" class="bottom_addr">&copy; 2013 Jumaboyev. All
Rights Reserved</td>
</tr>
</table>

```

2.7. book.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
<title>Education website</title>
<meta name="description" content="Education website">
<meta name="keywords" content="education, learning, teaching">
<link href="css/style.css" rel="stylesheet" type="text/css">
<? include ("connect.php");?>
</head>
<body>
<? include ("header.php");?>
<? include ("left.php");?>
    <td><h1><marquee direction="right" scrollamount="3">Mehmonlar
kitobi</marquee></h1></td>
    </tr>
    <tr>
        <td><table width="100%" border="0" cellpadding="0" cellspacing="0">
            <tr align="center">
                <td width="100%" height="1" align="center"><table width="100%" border="0"
cellspacing="0" cellpadding="0">
                    <tr>
                        <td></td>
                        <td width="100%" bgcolor="#1A658C"></td>
                    </tr>
                    <tr>
                        <td></td>
                    </tr>
                </table></td>
            </tr>
        </table></td>
        <td valign="top" class="body_txt">
<?php
if (isset($_GET['otp'])){
    $query="INSERT INTO boook (name,email,message,data)
values('$_GET[ism]',$_GET[mail]',$_GET[xabar]',Now())";
    $result=mysql_query($query);}
$query1="SELECT * FROM boook ORDER BY data desc";

```



```

$result1=mysql_query($query1);
?>
<body
  <h3>Qidiruv</h3>
  <input type="text" class="search" />
  <input type="submit" class="submit" value="OK" />
  <p>&nbsp;</p>
  <h3>Mehmonlar kitobi</h3>
  <?php while($row=mysql_fetch_array($result1)){
      echo $row['data']; echo " ";
      echo $row['name'];echo "<br />";
      echo $row['message'];echo "<hr />";
  }?>
  <form method="get" action="">
  Ism <input type="text" size="20" name="ism" class="search"/><br />
  <br />
  E-mail <input type="text" size="20" name="mail" class="search"/><br />
  <br />
  xat <br />
  <textarea name="xabar" >
  </textarea>
  <input name="otp" type="submit" value="Jonatish" class="submit" />
  </form>
  </td>
  <? include ("footer.php");?>
</body>
</html>

```

2.8. mehmon.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
<title>Education website</title>
<meta name="description" content="Education website">
<meta name="keywords" content="education, learning, teaching">
<link href="css/style.css" rel="stylesheet" type="text/css">
<? include ("connect.php");?>
</head>
<body>
<? include ("header.php");?>
<? include ("left.php");?>
  <td><h1>MEHMON</h1></td>
  </tr>
  <tr>
  <td><table width="100%" border="0" cellpadding="0" cellspacing="0">
    <tr align="center">
      <td width="100%" height="1" align="center"><table width="100%" border="0"
cellspacing="0" cellpadding="0">
        <tr>
          <td></td>
          <td width="100%" bgcolor="#1A658C"></td>
          <td></td>

```

```

        </tr>
    </table></td>
    </tr>
<tr>
    <td valign="top" class="body_txt">
        <?php include ("connect.php");?>
        <div class="pogoda">
            <!--Pogoda.uz-->
            <a
href="http://www.pogoda.uz/uz/tashkent.html"target="_blank">
                </a>
                <!--Pogoda.uz-->
            </div>
            <div class="valuta">
            <a href="http://bank.uz/currency/cb.html"title="Bank.uz-
все обанках Узбекистана"
                target="_blank"></a>
            </div>
        <? include ("footer.php");?>
    </body>
</html>
<table><tr><td>quyoshli +20 +15</p>
bulitli +15
+10</p>
yomg`irli +10
+8</p>
chaqmoqli
+8 +5</p>
bulitli +15
+10</p>
yomg`irli +10
+8</p>
quyoshli +20
+15</p>
</table>
<input type="submit" name="set" value="OK"/>

```

2.9. zapros1.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
<title>Education website</title>
<meta name="description" content="Education website">
<meta name="keywords" content="education, learning, teaching">
<link href="css/style.css" rel="stylesheet" type="text/css">
<? include ("connect.php");?>
</head>
<body>

```

```

<? include ("header.php");?>
<? include ("left.php");?>
<td><h1><marquee scrollamount="3"><p><b>1 Получить список и общее число
сотрудников указанных отделов (бухгалтерия, охрана и т.д.) по половому признаку, году
рождения, стажу, признаку наличия и количеству детей, размеру заработной
платы.</b></p></marquee></h1></td>
</tr>
<tr>
<td>
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr align="center">
<td width="100%" height="1" align="center"><table width="100%" border="0" cellspacing="0"
cellpadding="0">
<tr>
<td></td>
<td width="100%" bgcolor="#1A658C"></td>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top" class="body_txt">
<?php include ("connect.php")?>
<?php
$query2="Select ID_BOLIM,BOLIM_NOMI from BOLIM";
$result2=mysql_query($query2);
    if(isset($_GET['set'])){
        $query1="select * from XODIMLAR,BOLIM where
xodimlar.ID_BOLIM=BOLIM.ID_BOLIM";
        if($_GET['BOLIM']<>0){
            $query1.=" and BOLIM.ID_BOLIM='".$_GET['BOLIM]";
        }
        if(isset($_GET['XODIMLAR']) and isset($_GET['XODIMLAR1'])){
        } else{
            $hod=$_GET['XODIMLAR'];$hod.='$_GET['XODIMLAR1]';
            $query1.=" and xodimlar.JINSI='".$hod";
        }
        $result1=mysql_query($query1);}
?>
        <form name="OK" method="get" action="ZAPROS1.php">
        BOLIM<SELECT NAME="BOLIM">
            <option value="0" <?php if ($_GET["BOLIM"]=="0") {echo "selected";} ?> >HAMMA
        BOLIM
        </option>
        <?php
        while ($rows=mysql_fetch_array($result2))
        {?>
        <option value="<?php echo $rows['ID_BOLIM'];?>" <?php if
        ($_GET["BOLIM"]=="$rows[ID_BOLIM]) {echo "selected";} ?> ><?php echo
        $rows['BOLIM_NOMI'];?>
        </option><?php } ?>
        </SELECT><BR>

```

```

ERKAK<INPUT TYPE=checkbox NAME="XODIMLAR" VALUE="ERKAK">?php if
($_GET["ERKAK"]) {echo "checked";}?></BR>
AYOL<INPUT TYPE=checkbox NAME="XODIMLAR1" VALUE="AYOL" <?php if
($_GET["AYOL"]) {echo "checked";} ?></BR>
<input type="submit" name="set" value="OK"/>
</form>
<?php if(isset($result1)){ ?>
<table width="500" height="10" border="1px">
<tr><td bgcolor="#33FFFF"><h4 align="center" class="style18"> FIO</h4></td><td
bgcolor="#33FFFF"> <h4 align="center" class="style18">JINSI</h4></td><td
bgcolor="#33FFFF"> <h4 align="center" class="style18">TUG`_YIL</h4></td><td
bgcolor="#33FFFF"> <h4 align="center" class="style18">MALAKASI</h4></td><td
bgcolor="#33FFFF"> <h4 align="center" class="style18">MAOSHI</h4></td></td><td
bgcolor="#33FFFF"><h4 align="center" class="style18">FARZANT</h4></td>
<?php
while ($rows2=mysql_fetch_array($result1)){ ?>
<tr><td bgcolor="#FFFFCC" x="10px"><h5 align="center" class="style17"> <?php echo
$rows2[1]; ?></h5></td>
<td bgcolor="#FFFFCC"> <h5 align="center" class="style17"><?php echo $rows2[6];
?></h5></td>
<td bgcolor="#FFFFCC"> <h5 align="center" class="style17"><?php echo $rows2[2];
?></h5></td>
<td bgcolor="#FFFFCC"> <h5 align="center" class="style17"><?php echo $rows2[7];
?></h5></td>
<td bgcolor="#FFFFCC"> <h5 align="center" class="style17"><?php echo $rows2[5];
?></h5></td>
<td bgcolor="#FFFFCC"> <h5 align="center" class="style17"><?php echo $rows2[4];
?></h5></td>
</tr>
<?php }?>
</table><?php }?>
</td>
<? include ("footer.php");?>
</body>
</html>

```

2.10 zapros2.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
<title>Education website</title>
<meta name="description" content="Education website">
<meta name="keywords" content="education, learning, teaching">
<link href="css/style.css" rel="stylesheet" type="text/css">
<? include ("connect.php");?>
<style type="text/css">
<!--
.style4 {color: #000000}
-->
</style>
</head>

```

```

<body>
<span class="style17"><?php echo $rows2['JINSI']; ?></span>
<? include ("header.php");?>
<? include ("left.php");?>

<td><h1><marquee scrollamount="3"><p><b>2 Получить список и общее число
преподавателей указанных кафедр либо указанного факультета полностью либо по
размеру заработной платы.</b></p></marquee></h1>
</td>
</tr>
<tr>
<td>
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr align="center">
<td width="100%" height="1" align="center"><table width="100%" border="0"
cellspacing="0" cellpadding="0">
<tr>
<td></td>
<td width="100%" bgcolor="#1A658C"></td>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top" class="body_txt">

<?php include ("connect.php");?>
<?php

$query4="Select ID_FAK,FAK_NOMI from FAKULTET";
$result4=mysql_query($query4);
$query5="Select ID_BOLIM,BOLIM_NOMI from BOLIM";
$result5=mysql_query($query5);

if(isset($_GET['seet'])) {
    $query2="Select ID_KAF,KAF_NOMI from KAFEDRA where
kafedra.id_fak=$_GET[FAK]";
    $result2=mysql_query($query2);
    $query1="select * from XODIMLAR,FAKULTET, BOLIM, lavozi, kafedra where
XODIMLAR.ID_kaf=kafedra.ID_kaf and xodimlar.id_lav=lavozi.id_lav and
XODIMLAR.ID_BOLIM=BOLIM.ID_BOLIM and
XODIMLAR.ID_FAK=FAKULTET.ID_FAK";
    if($_GET[FAK]<>0){
        $query1.=" and FAKULTET.ID_FAK=$_GET[FAK]";
    }
    $result1=mysql_query($query1);}
if(isset($_GET['seet'])) {

    $query1="select * from XODIMLAR, kafedra, lavozi where
xodimlar.id_kaf=kafedra.id_kaf and xodimlar.id_lav=lavozi.id_lav";
    if($_GET[kaf]<>0){
        $query1.=" and kafedra.id_kaf=$_GET[kaf]";
    }
}

```

```

    }
    if($_GET['a']<>"){
    $query1.=" and lavozimi.oklad>$_GET[a]";}
    if($_GET['b']<>"){
    $query1.=" and lavozimi.oklad<$_GET[b]";}
    $result1=mysql_query($query1);}

        if(isset($_GET['eset'])){
        $query1="select * from XODIMLAR, BOLIM, lavozimi where
XODIMLAR.ID_BOLIM=BOLIM.ID_BOLIM AND xodimlar.id_lav=lavozimi.id_lav";
        if($_GET['BOL']<>0){
        $query1.=" and BOLIM.ID_BOLIM='$_GET[BOL]";
        }
        $result1=mysql_query($query1);}

?>

<form name="EK" method="get" action="ZAPROS2.php">
FAKULTET
<select name="FAK">
<option value="0" <?php if ($_GET["FAK"]=="0") {echo "selected";} ?> >HAMMA
FAKULTET </option>
<?php
while ($rows=mysql_fetch_array($result4))
{?>
<option value="<?php echo $rows[0];?>" <?php if ($_GET["FAK"]=="$rows[0]") {echo
"selected";} ?> ><?php echo $rows[1];?> </option>
<?php } ?>
</select>
<input type="submit" name="seet" value="EK"/><BR> <BR></form>
<?php if(isset($result2)){ ?><form name="OOK" method="get"
action="ZAPROS2.php">
<input type="hidden" name="seet" value="<?php echo $_GET['seet'];?>" />
<input type="hidden" name="FAK" value="<?php echo $_GET['FAK'];?>" />
KAFEDRA
<select name="kaf">
<option value="0" <?php if ($_GET["kaf"]=="0") {echo "selected";} ?> >HAMMA
KAFEDRA </option>
<?php
while ($rows=mysql_fetch_array($result2))
{?>
<option value="<?php echo $rows[0];?>" <?php if ($_GET["kaf"]=="$rows[0]")
{echo "selected";} ?> ><?php echo $rows[1];?> </option>
<?php } ?>
</select>
<input type="submit" name="seet" value="OOK"/>
</form><?php } ?>
<form name="OK" method="get" action="ZAPROS2.php">
<BR><BR><input type="hidden" name="seet" value="<?php echo $_GET['seet'];?>"
/>

```

```

<input type="hidden" name="FAK" value="<?php echo $_GET['FAK'];?>" />
<input type="hidden" name="seet" value="<?php echo $_GET['seet'];?>" />
<input type="hidden" name="kaf" value="<?php echo $_GET['kaf'];?>" />
<table><tr><td><br>min<br><br></td><td> <input type="text"
name="a"/></br></td></tr>
<tr><td><br>max<br><br></td><td> <input type="text"
name="b"/></br></td></tr></table>
<input type="submit" name="seet" value="OK"/>
</form>
<form name="OKK" method="get" action="ZAPROS2.php">
BOLIM <select name="BOL">
<option value="0" <?php if ($_GET["BOL"]=="0") {echo "selected";} ?> >HAMMA
BOLIM </option>
<?php
while ($rows=mysql_fetch_array($result5))
{?>
<option value="<?php echo $rows[0];?>" <?php if ($_GET["BOL"]=="$rows[0]")
{echo "selected";} ?> ><?php echo $rows[1];?> </option>
<?php } ?>
</select>
</SELECT><input type="submit" name="eset" value="OKK"/>
</form>
<br /><?php echo $query1;
if(isset($result1)){?>
<table height="10" border="1px" class="l1">
<tr><td bgcolor="#33FFFF"> FIO</td>
<td bgcolor="#33FFFF" >oklad</td>
<td bgcolor="#33FFFF" >JINSI</td>
<td bgcolor="#33FFFF" >MALAKASI</td>
<td bgcolor="#33FFFF" >MAOSHI1</td>
<td bgcolor="#33FFFF" >kaf nomi</td>
<td bgcolor="#33FFFF" >BOLIM NOMI</td>
<?php
while ($rows2=mysql_fetch_array($result1)){
$a= $rows2['MALAKASI'];
if($a==0){ $M= $rows2['oklad'];}
elseif(($a>1)and($a<=3)){ $M=$rows2['oklad']*1.1;}
elseif(($a>3)and($a<=5)){ $M=$rows2['oklad']*1.2;}
elseif(($a>5)and($a<=10)){ $M=$rows2['oklad']*1.3;}
else { $M=$rows2['oklad']*1.4;}
?>
<tr><td bgcolor="#FFFFCC" x="10px"> <?php echo $rows2['FISH']; ?></td>
<td bgcolor="#FFFFCC"> <?php echo $rows2['oklad']; ?></td>
<td bgcolor="#FFFFCC"> <?php echo $rows2['JINSI']; ?></td>
<td bgcolor="#FFFFCC"> <?php echo $rows2['MALAKASI']; ?></td>
<td bgcolor="#FFFFCC"> <?php echo $M; ?></td>
<td bgcolor="#FFFFCC"> <?php if(isset($_GET['kaf'])or isset($_GET['FAK']))){echo
$rows2['KAF_NOMI']; } else {echo "-";} ?></td>
<td bgcolor="#FFFFCC"> <?php if(isset($_GET['BOL']))){echo
$rows2['BOLIM_NOMI']; } else {echo "-";} ?></td>
</tr>
<?php }?>
</table> <?php }?> </td>

```

```
<? include ("footer.php");?>  
</body>  
</html>
```

3.Создание базы данных в PhpMyAdmin

3.1.Работа с базами данных в РНР

Перед тем как пользоваться БД нужно в программе *phpMyAdmin* предоставить права на нее какому-либо пользователю.

При нажатии кнопки (ссылки) **Привилегии** открывается окно **User overview**, в котором надо выбрать пользователя (или создать его). В качестве пользователя мы будем использовать встроенный в *phpMyAdmin* пользователь “**root**” и этот пользователь имеет все привилегии.

В PhpMyAdmin создала базу данных “Спортивные организации города” содержащие следующие таблицы:

- ❖ Болим
- ❖ Бооок
- ❖ Факултет
- ❖ Голос
- ❖ Гурух
- ❖ Илмий_Иш
- ❖ Кафедра
- ❖ Лавозими
- ❖ Студент
- ❖ Ходимлар
- ❖ Йангилик

Это показано на рис.6.

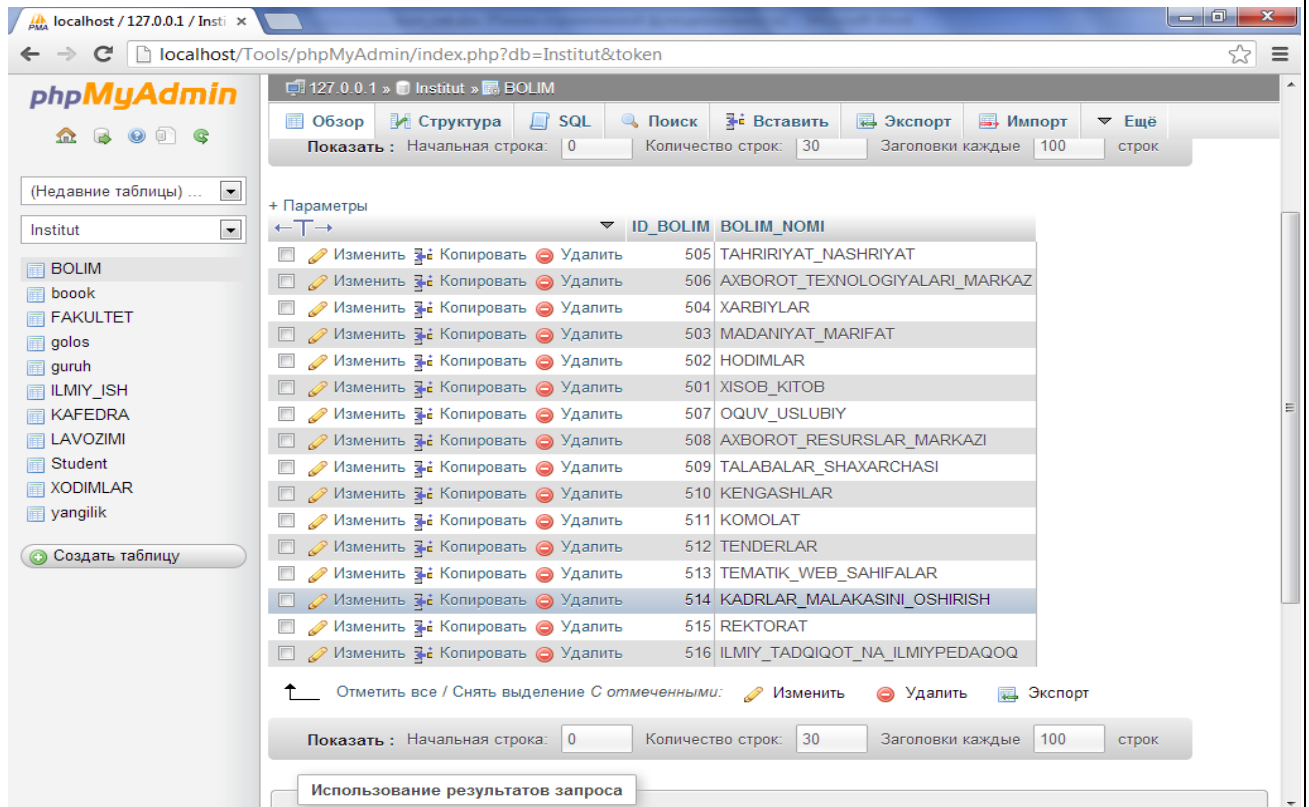


Рис 2. База данных в PhpMyAdmin.

3.2.Соединение с сервером и базой данных

mysql_connect() – устанавливает связь с сервером MySQL

mysql_select_db() – выбор БД на сервере:

```
<?php @mysql_connect("localhost","root") or die("Невозможна подключится к серверу");
```

```
@mysql_select_db("oybekdb") or die("Невозможна выбрать Базу Данных");
```

```
?>
```

где

localhost – имя компьютера, **root** – имя пользователя;

@ перед функцией **mysql_connect()** подавляет все сообщения об ошибке, которые заменяются сообщением в **die()**; **oybekdb** – имя базы данных;

\$link – идентификатор соединения;

Можно функции соединения с сервером и БД поместить в файл, например, conn.php.

После завершения работы с сервером MySQL соединение необходимо закрыть функцией **mysql_close()**.

Заключение

Создание сайта ради фирмы является важнейшим аспектом ее деятельности наряду вместе с выводом новых товаров на рынок, их позиционированием, рекламой а также продвижением. При этом оригинальное качество сайта, вероятно, значительно повлиять на репутацию фирмы.

Первое, место с чего начинают оценку сайта, это его наружный вид, но еще дальше делают выводы, изучая содержимое. Посетители сайта составляют взгляд о дизайне не более чем за секунду, а также на основании с его мнения формируется внушительная частица представления о фирмы-владельце. Дизайн сайта - это первое впечатление, которое оказывает на посетителя, а также от него зависит, захочет ли он продолжить дальнейший просмотр всего всемирного сети-ресурса или он поторопится покинуть просматриваемые страницы этого сайта.

При изготовлении web-сайта, в особенности дизайна, мы должны учитывать все самые необходимые основные потребности к его разработке: начиная от создания сайта в едином, фирменном стиле фирмы-заказчика впрямь до максимального удобства пользования сайтом, в навигации, а также в интуитивном понятном интерфейсе.

В данном курсовом проекте я создала сайт на тему “ Институт ”. На эту тему не только я создала и обрабатывала некоторые сайты. Но смысл в том, что я создала этот сайт в связи, с базой данных которой увеличивает качества и безопасность информации, хранящиеся в сайте, в сайте все информации будут загружаться из базы данных, а это намного эффективнее и экономит время пользователей при посещении сайта. То есть, когда пользователь посещает страницы сайта, при этом страницы сайта загружаются быстрее, чем обычные страницы сайтов.

Выполнение курсового проекта я закрепила знания, полученные в течение всего курса «Интернет технологии». В результате выполнения курсового проекта я применила накопленные мною знания, которые и помогли мне разработать данный курсовой проект. Передо мной была поставлена задача, которая нашла своё решение, как в теоретической, так и в практической части. Была создана сайт «Институт» в Macromedia Dreamweaver 8, а также разработано приложение в PhpMyAdmin, работающее с данной базой данных.

Список использованной литературы

1. А.Матросова, М.Чаунин, А.Сергеев - HTML 4.0
2. Энди Харрис. PHP/MySQL для начинающих.
3. Дмитрий Котерев. Самоучитель PHP 4.
4. Под редакцией В.Н. Печниковой. Создания Web – сайтов без посторонней помощи.
5. Б. Артанов. Web – мастеринг без посторонней помощи.
6. Лаура Томсон, Люк Веллинг - PHP и MySQL
7. uch1.tashiit.uz