Ввод и передача данных

Работа с HTML формами

Ввод данных через HTML-формы

Чтобы принять данные (т.е. осуществить т.н. «пользовательский ввод») и обработать их, необходимо решить две задачи:

1. Обеспечить возможность ввода данных на стороне клиента с помощью HTML-форм;
2. Передать данные на сервер для обработки и получить результат.

Все элементы управления, используемые для ввода данных, которые должны быть отправлены серверной программе, обязательно должны находиться в форме.

Форма HTML

Форма - один из важных элементов любого HTML-документа, предназначенная для обмена данными между пользователем и сервером. Позволяет организовать пользовательский интерфейс web-приложения для отправки данных (пользователем) на сервер.

Возможность обработки форм является одним из сильнейших средств языка PHP.

Основные атрибуты <form>

```
<form action = "fileName.php"
        method = "POST"
                enctype = "multipart/form-data">
    <!-- Элементы  внутри формы -->
</form>
```

Атрибуты являются обязательными; если их значения не указаны – браузер подставляет значения по умолчанию.

- Action – определяет адрес документа, обрабатывающего данные из формы (по умолчанию – текущий документ).
- Method – определяет используемый метод отправки данных (GET или POST; по умолчанию - GET).
- Enctype – определяет способ кодирования данных (по умолчанию - application/x-www-form-urlencoded).

Обработка элементов форм: текстовые поля

Виды полей:

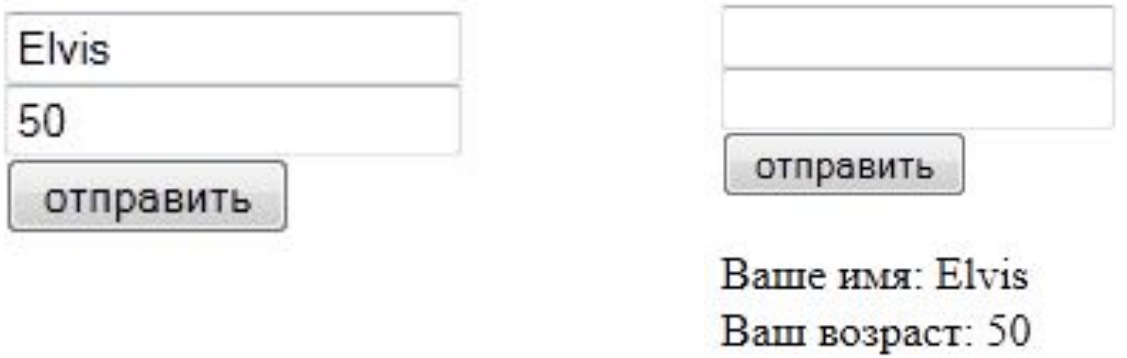
- Однострочное текстовое поле;
- Многострочное текстовое поле;
- Скрытое поле;
- Типы полей данных HTML 5.

Алгоритм обработки:

1. Обращение к полю в HTML форме происходит по значению атрибута name обрабатываемого элемента;
2. В зависимости от метода обработки формы (значение атрибута method тэга form) данные попадают в глобальным массив GET или POST; индексом/ключом массива выступает значение атрибута name элемента формы; значением элемента массива – введенное пользователем значение элемента.

Передача данных на сервер

Передача данных на сервер осуществляется методами GET и/или POST в запросе браузера:



Передача данных на сервер: GET

```
<html>
<body>
<form action=" " method="get">
<input type = "text" name = "name"><br>
<input type = "text" name = "age"><br>
<input type = "submit" value = "отправить">
</form>
</body>
</html>
```

```
<?php
echo "Ваше имя: ". $_GET['name']."<br>";
echo "Ваш возраст: ".$_GET['age'];
?>
```

```
GET /form.php?name=Elvis&age=50 HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:6.0) Gecko/20100101 Firefox/6.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ru-ru,ru;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Accept-Charset: windows-1251,utf-8;q=0.7,*;q=0.7
Connection: keep-alive
Referer: http://localhost/form.php
Cache-Control: max-age=0
```

Передача данных на сервер: POST

```
<html>
<body>
<form action=" " method="post">
<input type = "text" name = "name"><br>
<input type = "text" name = "age"><br>
<input type = "submit" value = "отправить">
</form>
</body>
</html>
```

```
<?php
echo "Ваше имя: ". $_POST['name']."<br>";
echo "Ваш возраст: ".$_POST['age'];
?>
```

```
POST /form.php?name=Elvis&age=50 HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:6.0) Gecko/20100101 Firefox/6.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ru-ru,ru;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Accept-Charset: windows-1251,utf-8;q=0.7,*;q=0.7
Connection: keep-alive
Referer: http://localhost/form.php?name=Elvis&age=50
Content-Type: application/x-www-form-urlencoded
Content-Length: 17
  name=Elvis&age=50
```

Глобальные массивы

Содержат информацию о состоянии сервера и среды выполнения скрипта. Доступны в любом месте скрипта без дополнительный объявлений. В том числе к глобальным массивам относятся:

$_GET – содержит список переменных, переданных скрипту методом GET, т.е. через параметры URL-запроса.

$_POST – содержит список переменных, переданных массивом методом POST.

$_REQUEST - содержит данные переменных $_GET, $_POST и $_COOKIE.

Обработка элементов форм: переключатели (radio)

```
<html>
   <form action="radio.php" method = "GET" enctype = "multipart/form-data">
      <p><input type = "radio" name = "MyRadio" value = "First">First Variant</p>
      <p><input type = "radio" name = "MyRadio" value = "Second">Second Variant</p>
      <p><input type = "radio" name = "MyRadio" value = "Third">Third Variant</p>
      <p><input type = "submit" name = "submit">
   </form>
</html>
<?php
    $var = $_GET['MyRadio'];
    switch($var)
      {
         case "First":
         echo "You choose $var";
         break;

         case "Second":
         echo "You choose $var";
         break;

         case "Third":
         echo "You choose $var";
         break;
      }
?>
```

Обработка элементов форм: флажки (checkbox)

```
<html>
   <form action="checkbox.php" method = "GET" enctype = "multipart/form-data">
        <p><input type = "checkbox" name = "MyCheckBox[]" value = "First">First Variant</p>
        <p><input type = "checkbox" name = "MyCheckBox[]" value = "Second">Second Variant</p>
        <p><input type = "checkbox" name = "MyCheckBox[]" value = "Third">Third Variant</p>
        <p><input type = "submit" name = "submit">
   </form>
</html>
<?php
    $arr = $_GET['MyCheckBox'];
        if(empty($arr))
            {
                echo "Nothing";
            }
        else
        {
            $count = count($arr);
            for($i=0; $i<$count; $i++)
                {

                    echo $arr[$i]." ";
                }
        }
?>
```

Обработка элементов форм: поле со списком

```
<html>
    <form action="select.php" method = "GET" enctype = "multipart/form-data">
        <select name = "languages" size = "1">
            <optgroup label = "Романская группа">
                <option value = "Sp">Испанский</option>
                <option value = "Pr">Португальский</option>
                <option value = "Fr">Французский</option>
            </optgroup>
            <optgroup label = "Германская группа">
                <option value = "En">Английский</option>
                <option value = "Gr">Немецкий</option>
            </optgroup>
        </select>
        <p><input type = "submit" name = "submit">
    </form>
</html>
<?php
$var = $_GET['languages'];

//Далее - работа со значением атрибута value, например
if($var == "En")
    {
        echo "English";
    }
    else
    {
        echo "Other";
    }
?>
```

Элементы формы: кнопки

<input type = “button” …>

<input type = “submit” …>

<input type = “reset” …>

Submit – передача данных из формы обработчику;

Button – простая кнопка; для нее пишется обработчик;

Reset – кнопка сброса значений формы к первоначальным значениям.

Атрибуты кнопок:

Name – имя кнопки (для обработчика);

Value - значение кнопки (надпись).

Электронная почта

Функция mail()

Отправка почты

В PHP существует одна функция отправки почты – **mail()**.

Описание:

bool **mail** (string $to , string $subject , string $message
[, string $additional_headers [, string $additional_parameters]])

Список аргументов:

to - получатель, или получатели письма;

subject - тема отправляемого письма;

message - отправляемое сообщение;

additional_headers - используется для добавления дополнительных заголовков (From, Cc, and Bcc);

additional_parameters – используется для передачи дополнительных флагов в виде аргументов командной строки для программы сконфигурированной для отправки писем, указанной директивой *sendmail_path*. Например, можно установить отправителя письма при использовании sendmail с помощью опции *-f*.

Отправка почты

Пример отправки простого письма:

```
<?php
mail(‘reciver@recivehost.com’, ‘subject’, ‘body’);
?>
```

Пример отправки письма с дополнительными заголовками:

```
<?php
   $to      = ‘reciver@recivehost.com';
   $subject = 'subject';
   $message = ‘my letter';
   $headers = 'From: myAddress@myhost.com' . "\r\n" .
              'Reply-To: myAddress@myhost.com' . "\r\n“;

   mail($to, $subject, $message, $headers);
   ?>
```