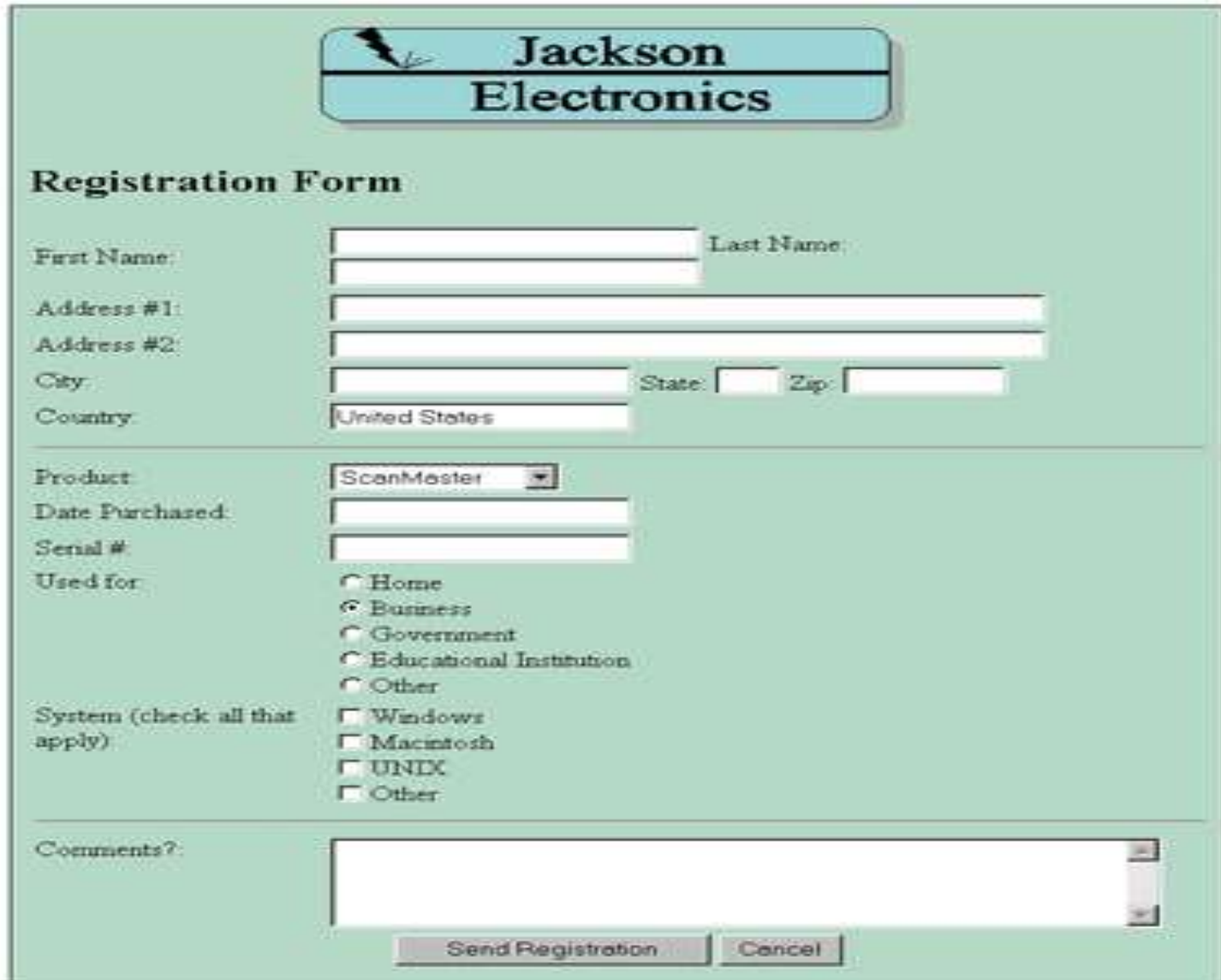


HTML FORMS

Client-side Web Application Development Forms



The image shows a web registration form for Jackson Electronics. The form is titled "Registration Form" and contains several input fields and a list of options. At the top, there is a logo for Jackson Electronics featuring a lightning bolt icon. The form is divided into several sections: personal information, address, product details, and system compatibility. The "Used for" section has radio buttons, and the "System" section has checkboxes. At the bottom, there is a "Comments?" field and two buttons: "Send Registration" and "Cancel".

Jackson Electronics

Registration Form

First Name: Last Name:

Address #1:

Address #2:

City: State: Zip:

Country:

Product:

Date Purchased:

Serial #:

Used for:

- Home
- Business
- Government
- Educational Institution
- Other

System (check all that apply):

- Windows
- Macintosh
- UNIX
- Other

Comments?:

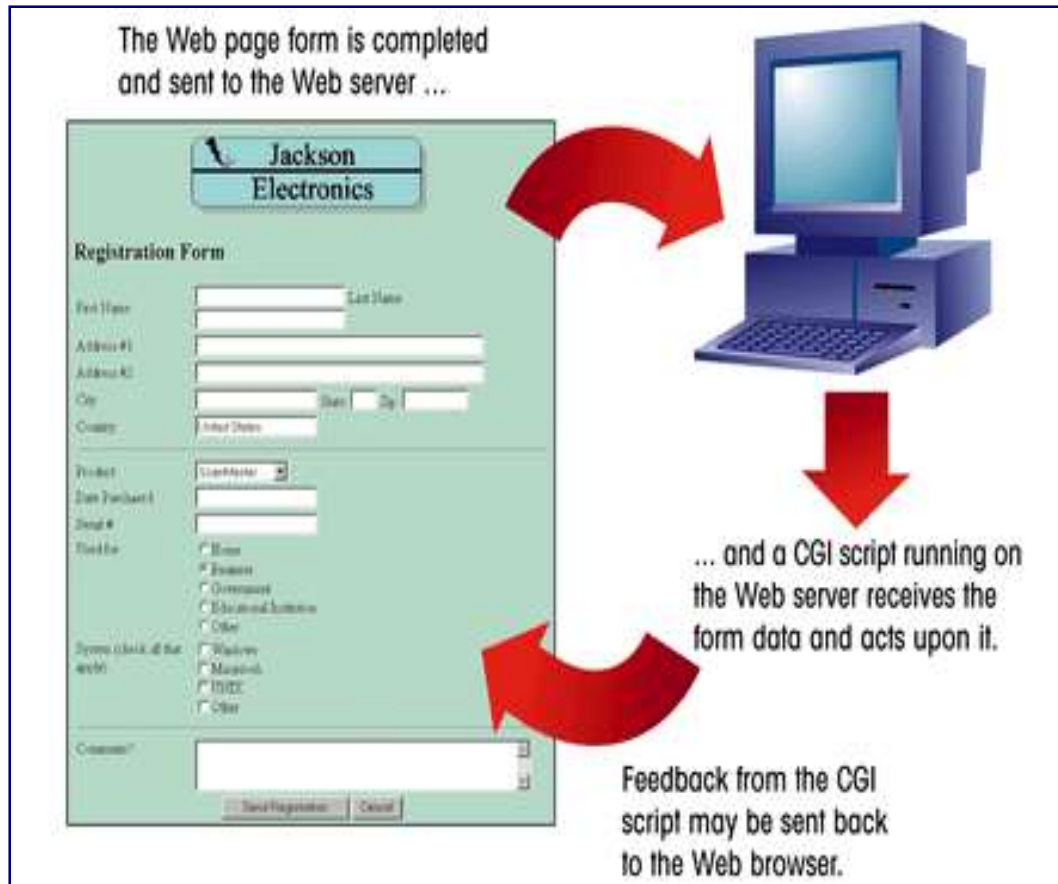
How do forms work?

- Forms are used on the Web to collect information from users
- The information is then sent to a program running on the Web server called CGI script (Common Gateway Interface)
- The CGI script receives the data from the Web page, then acts on that data to perform a certain task

What are forms?

- `<form>` is just another kind of HTML tag
- HTML forms are used to create (rather primitive) GUIs on Web pages
 - Usually the purpose is to ask the user for information
 - The information is then sent back to the server
- A form is an area that can contain form elements
 - The syntax is: `<form parameters> ...form elements... </form>`
 - Form elements include: buttons, checkboxes, text fields, radio buttons, drop-down menus, etc
 - Other kinds of HTML tags can be mixed in with the form elements
 - A form usually contains a `Submit` button to send the information in the form elements to the server
 - The form's *parameters* tell JavaScript how to send the information to the server (there are two different ways it could be sent)
 - Forms can be used for other things, such as a GUI for simple programs

Forms/CGI script interaction

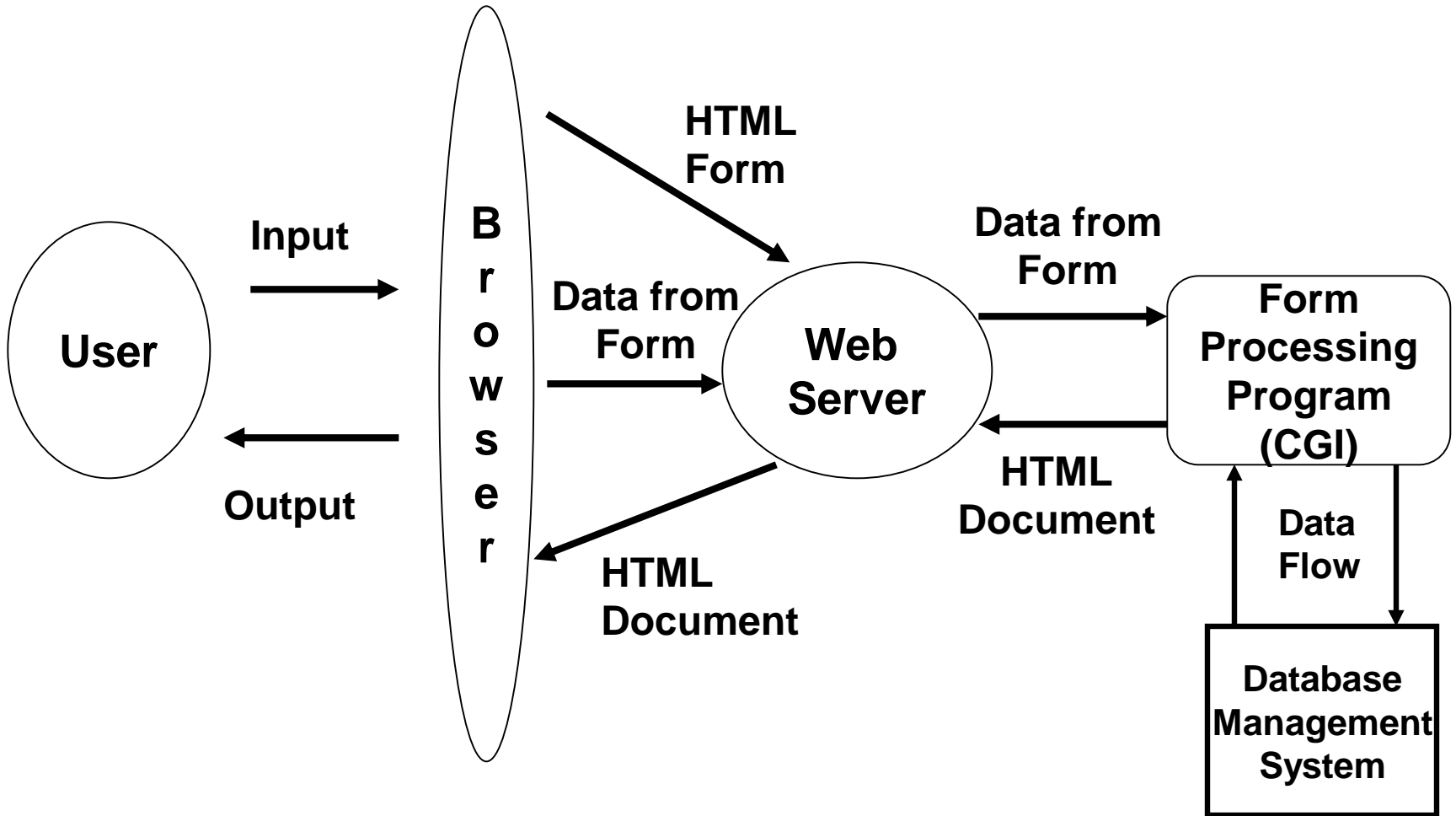


CGI script languages

- AppleScript
- C/C++
- Perl
- The Unix shell
- Visual Basic
- ASP

**USER
INTERACTIONS:
FORMS**

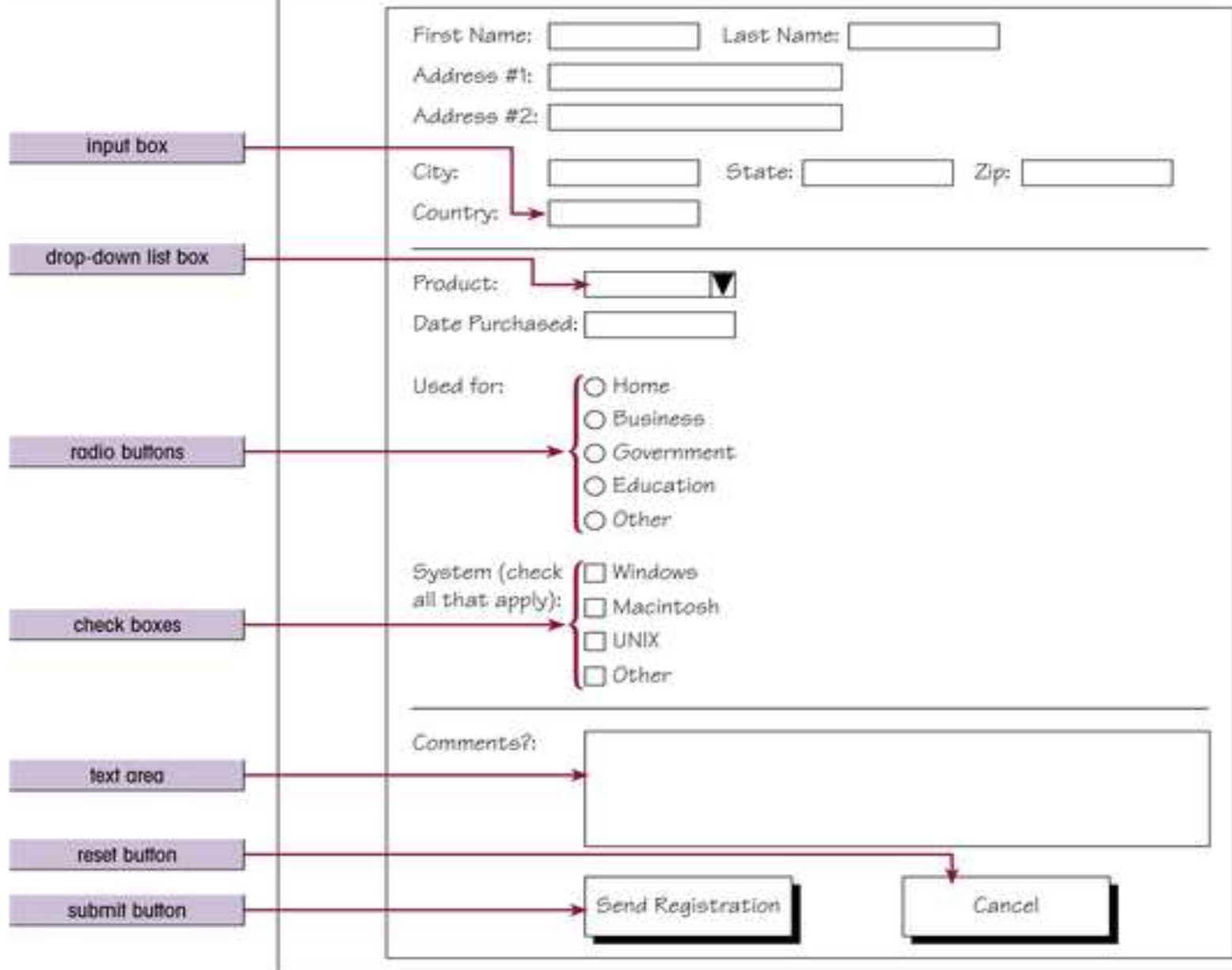
Form Processing



Flow of Information for Forms

Figure 6-3

PARTS OF A FORM



Forms and JavaScript

- The JavaScript language can be used to make pages that “do something”
 - You *can* use JavaScript to write complete programs, but...
 - Usually you just use snippets of JavaScript here and there throughout your Web page
 - JavaScript code snippets can be attached to various form elements
 - For example, you might want to check that a **zipcode** field contains a 5-digit integer before you send that information to the server
- Microsoft sometimes calls JavaScript “active scripting”
- HTML forms can be used without JavaScript, and JavaScript can be used without HTML forms, but they work well together

Forms

- HTML Forms are used to select different kinds of user input.
- A form is an area that can contain form elements.
- **Field:** Each element in which user can enter information
- **Field value:** Data entered in a field
- **<FORM> ...</FORM>:** Encloses the form elements and layout tags
- **<INPUT>:** For each field you have to specify the type of input (input box, radio button etc)

The <form> tag

- The <form *arguments*> ... </form> tag encloses form elements (and probably other HTML as well)
- The arguments to **form** tell what to do with the user input
 - **action="url"** (required)
 - Specifies where to send the data when the **Submit** button is clicked
 - **method="get"** (default)
 - Form data is sent as a URL with **?form_data** info appended to the end
 - Can be used *only* if data is all ASCII and not more than 100 characters
 - **method="post"**
 - Form data is sent in the body of the URL request
 - Cannot be bookmarked by most browsers
 - **target="target"**
 - Tells where to open the page sent as a result of the request
 - **target= _blank** means open in a new window
 - **target= _top** means use the same window

Form structure

<FORM>

<TABLE>

<TR>

<TD>First Name: </TD>

<TD> <INPUT NAME=FirstName> </TD>

</TR>

<TR>

<TD>Address: </TD>

<TD> <INPUT NAME=Address> </TD>

</TR>

</TABLE>

</FORM>

First Name:	<input type="text"/>
Address:	<input type="text"/>

<FORM> Attributes

```
<FORM NAME=text ACTION=URL METHOD=GET/POST>
```

- **NAME**: Identifies the form
- **ACTION**: Identifies the CGI script that will process the form
- **METHOD**: Controls how the browser sends data to the Web server for the CGI Script.
 - **GET**: Packages the form data by appending it to the end of the URL: specified in the ACTION property
 - **POST**: Send form information in a separate data stream

The <input> tag

- Most, but not all, form elements use the `input` tag, with a `type="..."` argument to tell which kind of element it is
 - `type` can be `text`, `checkbox`, `radio`, `password`, `hidden`, `submit`, `reset`, `button`, `file`, or `image`
- Other common `input` tag arguments include:
 - `name`: the name of the element
 - `value`: the “value” of the element; used in different ways for different values of `type`
 - `readonly`: the value cannot be changed
 - `disabled`: the user can’t do anything with this element
 - Other arguments are defined for the `input` tag but have meaning only for certain values of `type`

Text input

A text field:

```
<input type="text" name="textfield" value="with an initial value">
```

A text field:

A multi-line text field

```
<textarea name="textarea" cols="24" rows="2">Hello</textarea>
```

A multi-line text field:

A password field:

```
<input type="password" name="textfield3" value="secret">
```

A password field:

- Note that two of these use the `input` tag, but one uses `textarea`

<INPUT> attributes

```
<INPUT NAME=text VALUE=text/# SIZE=#  
MAXLENGTH=# TYPE=predefined>
```

- **NAME**: Identifies the field
- **SIZE**: Of the input box, in characters, default is 20, does **not** limit the input
- **VALUE**: Default text or number that will appear in the field
- **MAXLENGTH**: Max # of characters user can input
- **TYPE**: text box is the default

<INPUT TYPE = >

- **PASSWORD**: Characters typed by user are displayed as bullets or asterisk
- **HIDDEN**: used to pass hidden information to the CGI script (like the email of the person that should receive the form).
 - The user does not see the content of the field
 - The user is not allowed to change the value of the hidden field
 - Usually all hidden fields are placed right after the <FORM> tag

Radio buttons

```
<INPUT TYPE=RADIO NAME=text VALUE=value CHECKED>
```

- Display a list of choices from which the user makes a selection
- **NAME**: the field containing the radio button. The same may be used for many buttons.
- **VALUE**: the value sent to the CGI script
- **CHECKED**: makes the particular radio button the default choice

Check Boxes

```
<INPUT TYPE=CHECKBOX NAME=text VALUE=value CHECKED>
```

- It is either selected or not. There is only one check box per field
- **NAME**: the field containing the check box
- **VALUE**: the value sent to the CGI script if the check box is selected (yes/no)
- **CHECKED**: makes the particular check box the default choice

Text area

```
<TEXTAREA NAME=text COLS/ROWS=value WRAP=option>  
Default text </TEXTAREA>
```

- Allows users to enter comments
- **NAME**: the field containing the text area
- **ROWS**: number of available lines
- **COLS**: number of characters in each line
- **WRAP**:
 - **OFF**: turns off text wrapping
 - **SOFT (VIRTUAL)**: turn text wrapping on

Selection lists

```
<SELECT NAME=text SIZE=value MULTIPLE>  
  <OPTION> Option 1  
  <OPTION> Option 2  
</SELECT>
```

- A list box from which the user selects a particular (set of) value(s). (Drop-down menus)
- **SIZE**: # of items the list displays in the form
- **MULTIPLE**: using the CTR or SHFT key the user can select more than one item

Form Buttons

```
<INPUT TYPE=text VALUE="text">
```

- Form fields that perform an action when activated (by the user clicking)
- **VALUE**: text that appears on the button
- **TYPE=**
 - **SUBMIT**: Submits the form to the CGI script
 - **RESET**: Cancels or resets the appearance of a form
 - **BUTTON**: Performs an action within the page by running a script