core WEB programming

HTML Forms

Sending Data to Server-Side Programs

Agenda

- Sending data from forms
- The FORM element
- Text controls
- Push buttons
- Check boxes and radio buttons
- Combo boxes and list boxes
- File upload controls
- Server-side image maps
- Hidden fields
- Grouping controls
- Tab ordering

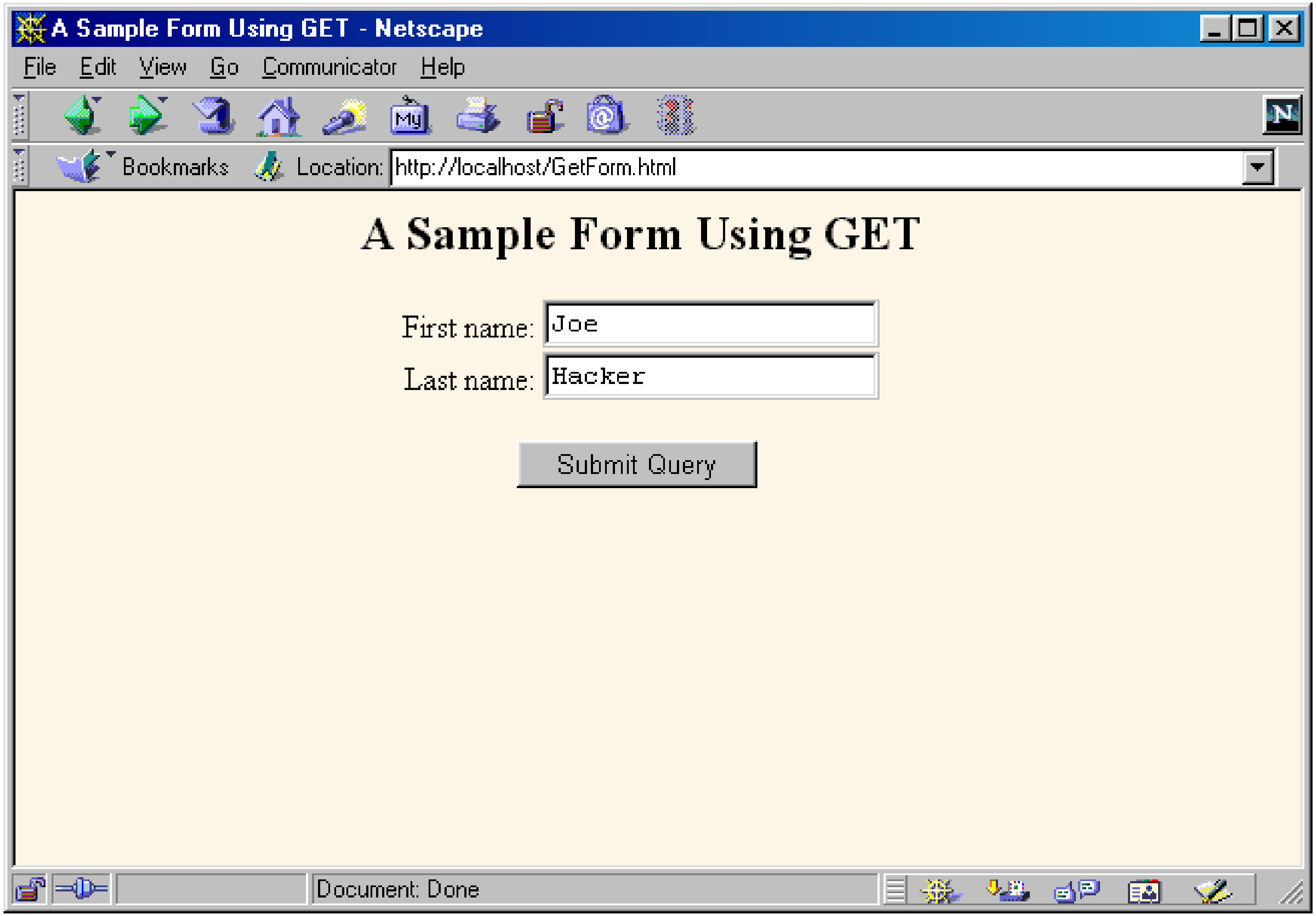
Sending Data with GET

```
…
<BODY BGCOLOR="#FDF5E6">
<H2 ALIGN="CENTER">A Sample Form Using GET</H2>

<FORM ACTION="http://localhost:8088/SomeProgram">
  <CENTER>
  First name:
  <INPUT TYPE="TEXT" NAME="firstName" VALUE="Joe"><BR>
  Last name:
  <INPUT TYPE="TEXT" NAME="lastName" VALUE="Hacker"><P>
  <INPUT TYPE="SUBMIT">
  </CENTER>
</FORM>

</BODY></HTML>
```

Initial Result



A Sample Form Using GET - Netscape
File Edit View Go Communicator Help
Bookmarks
Location:

http://localhost/GetForm.html

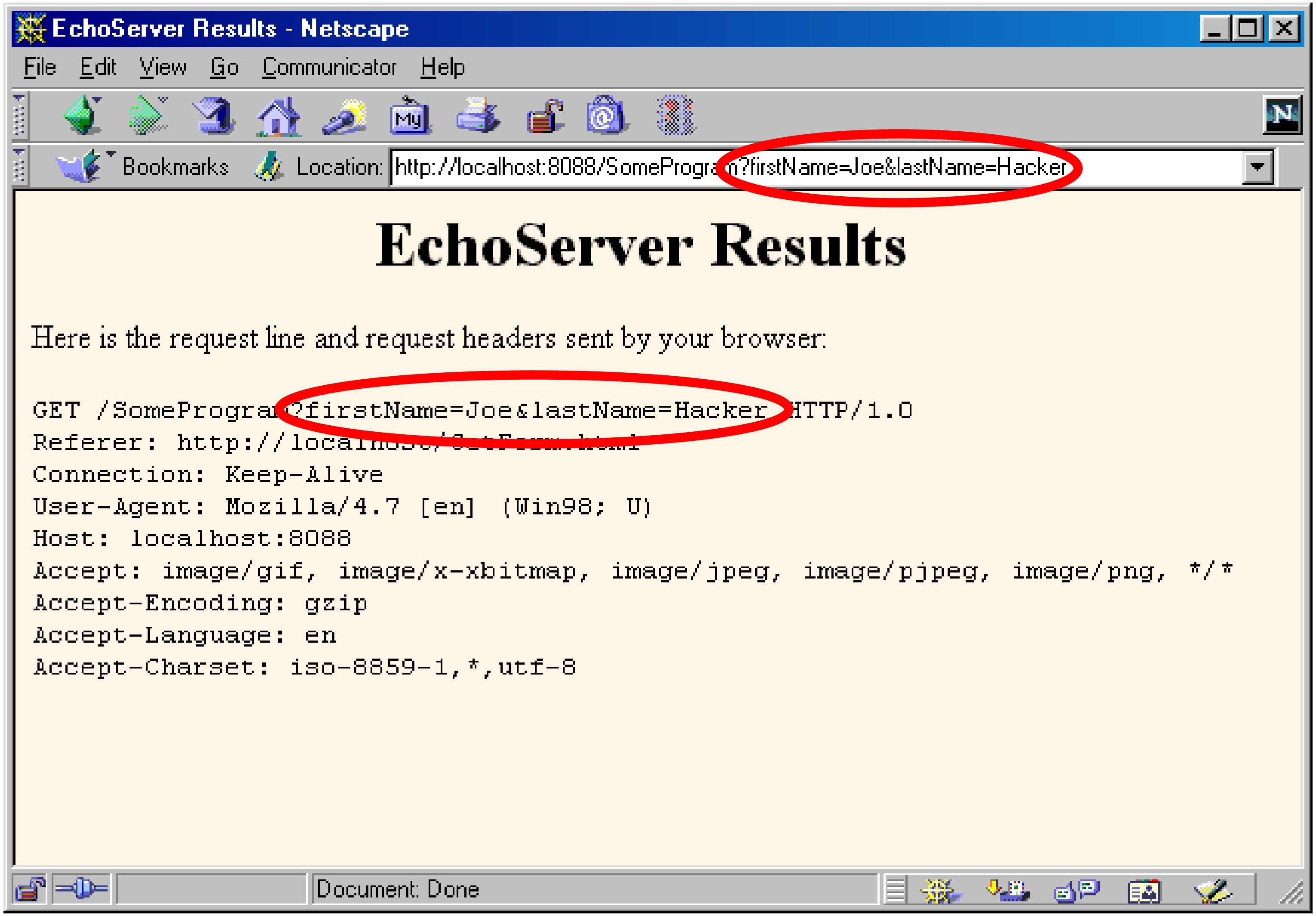
A Sample Form Using GET

First name: Joe

Last name: Hacker

Document: Done

Submission Result


EchoServer Results - Netscape
File Edit View Go Communicator Help
Bookmarks
Location: http://localhost:8088/SomeProgram?firstName=Joe&lastName=Hacker
Document: Done

EchoServer Results

Here is the request line and request headers sent by your browser:

```
GET /SomeProgram?firstName=Joe&lastName=Hacker HTTP/1.0
Referer: http://localhost/GetForm.html
Connection: Keep-Alive
User-Agent: Mozilla/4.7 [en] (Win98; U)
Host: localhost:8088
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
Accept-Encoding: gzip
Accept-Language: en
Accept-Charset: iso-8859-1,*,utf-8
```

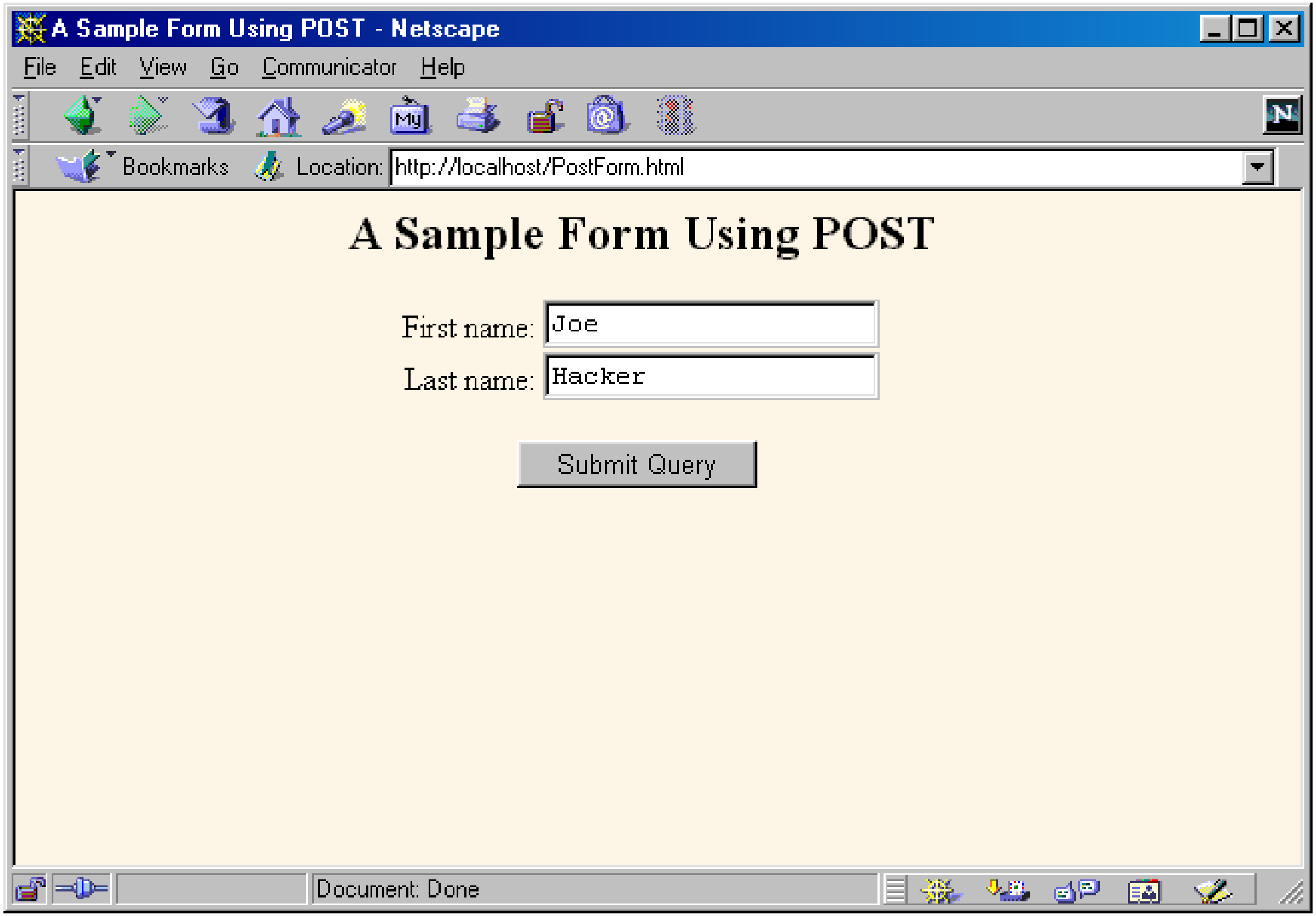
Sending Data with POST

```
…
<BODY BGCOLOR="#FDF5E6">
<H2 ALIGN="CENTER">A Sample Form Using POST</H2>

<FORM ACTION="http://localhost:8088/SomeProgram"
      METHOD="POST">
  <CENTER>
  First name:
  <INPUT TYPE="TEXT" NAME="firstName" VALUE="Joe"><BR>
  Last name:
  <INPUT TYPE="TEXT" NAME="lastName" VALUE="Hacker"><P>
  <INPUT TYPE="SUBMIT">
  </CENTER>
</FORM>

</BODY></HTML>
```

Initial Result



A Sample Form Using POST - Netscape
File Edit View Go Communicator Help
Bookmarks
Location: http://localhost/PostForm.html

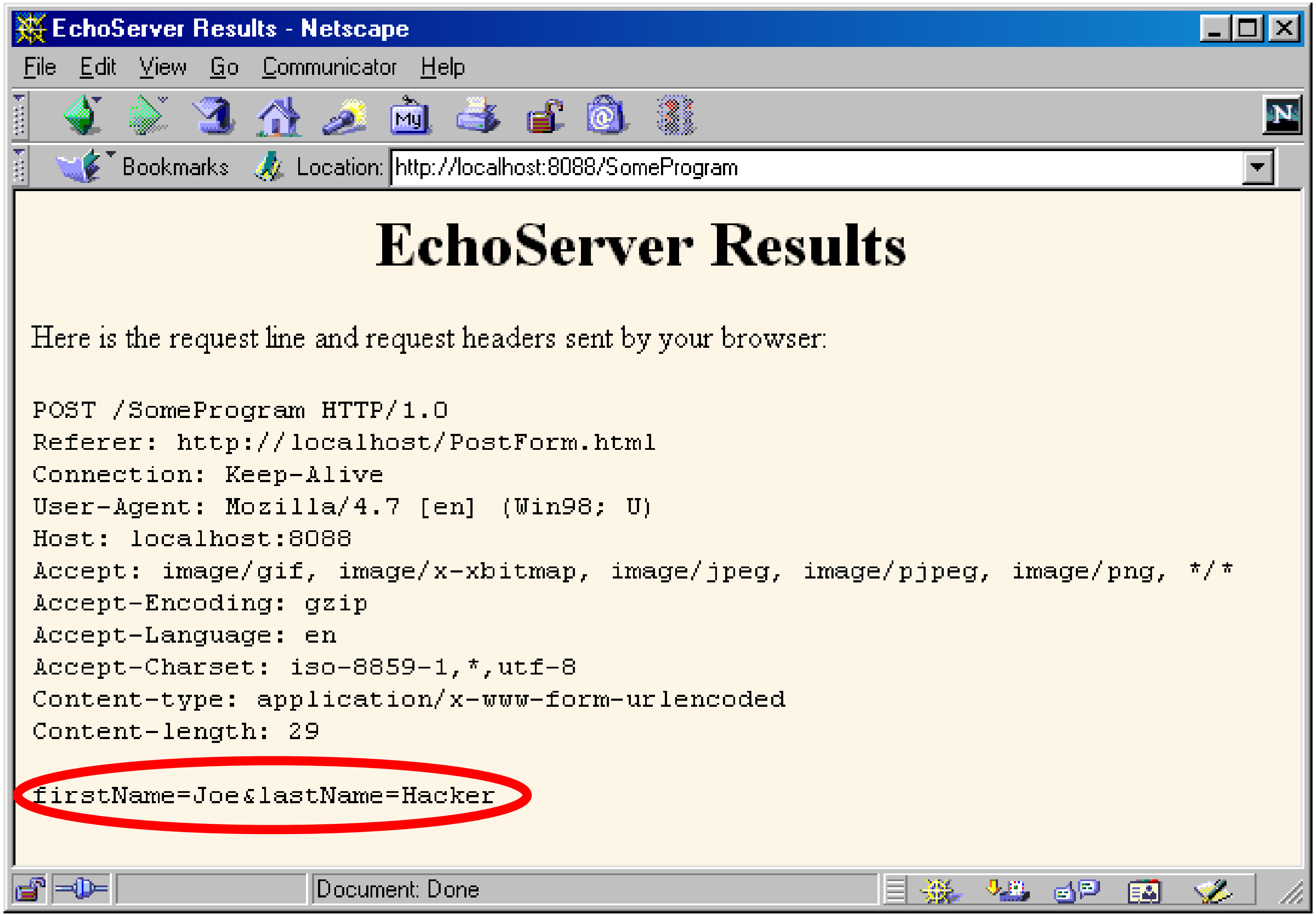
A Sample Form Using POST

First name: Joe

Last name: Hacker

Submit Query
Document: Done

Submission Result


EchoServer Results - Netscape
File Edit View Go Communicator Help
Bookmarks
Location: http://localhost:8088/SomeProgram
Document: Done

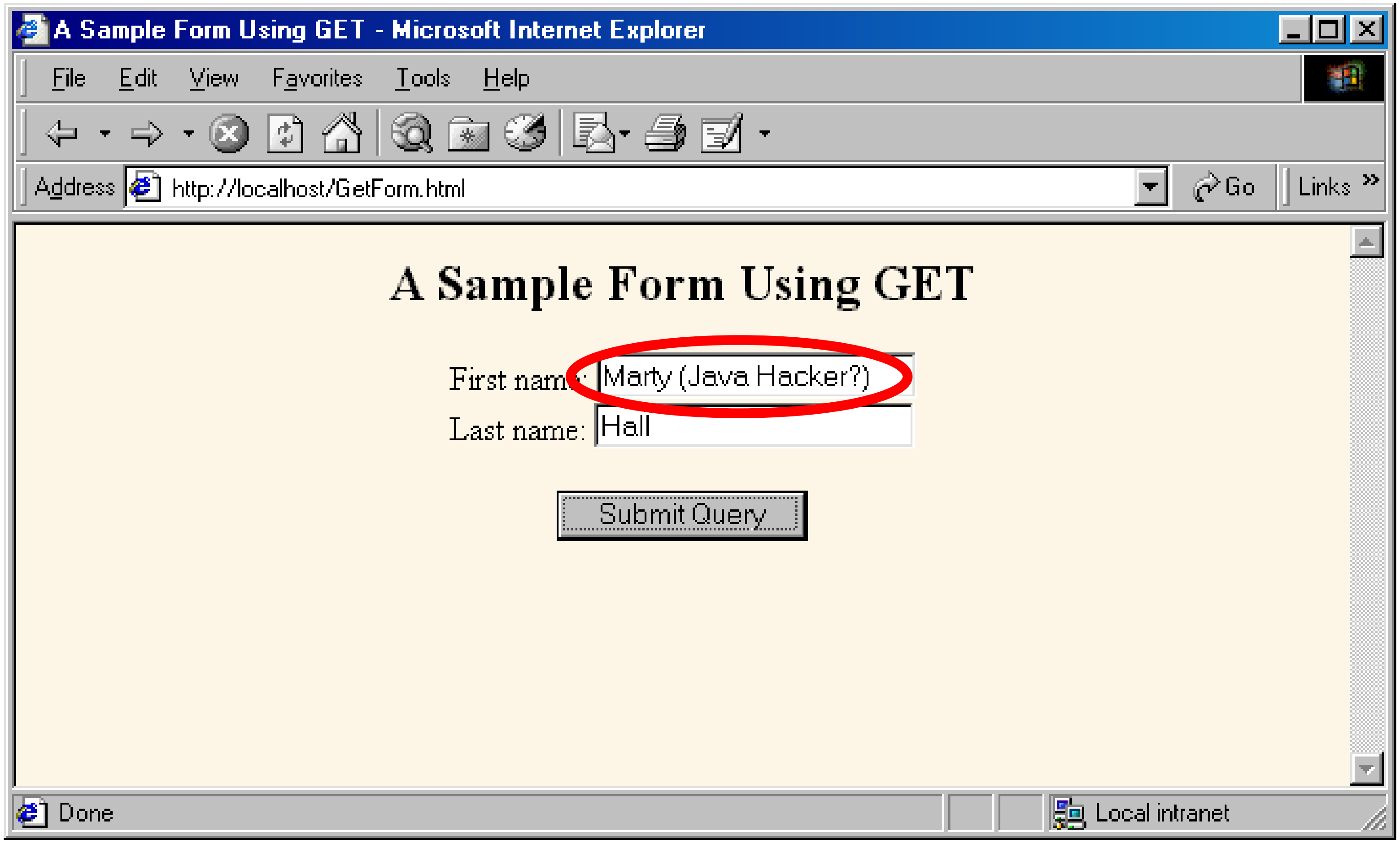
EchoServer Results

Here is the request line and request headers sent by your browser:

```
POST /SomeProgram HTTP/1.0
Referer: http://localhost/PostForm.html
Connection: Keep-Alive
User-Agent: Mozilla/4.7 [en] (Win98; U)
Host: localhost:8088
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
Accept-Encoding: gzip
Accept-Language: en
Accept-Charset: iso-8859-1,*,utf-8
Content-type: application/x-www-form-urlencoded
Content-length: 29

firstName=Joe&lastName=Hacker
```

URL Encoding: Original Form



A Sample Form Using GET - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address
Go
Links
Done
Local intranet

http://localhost/GetForm.html

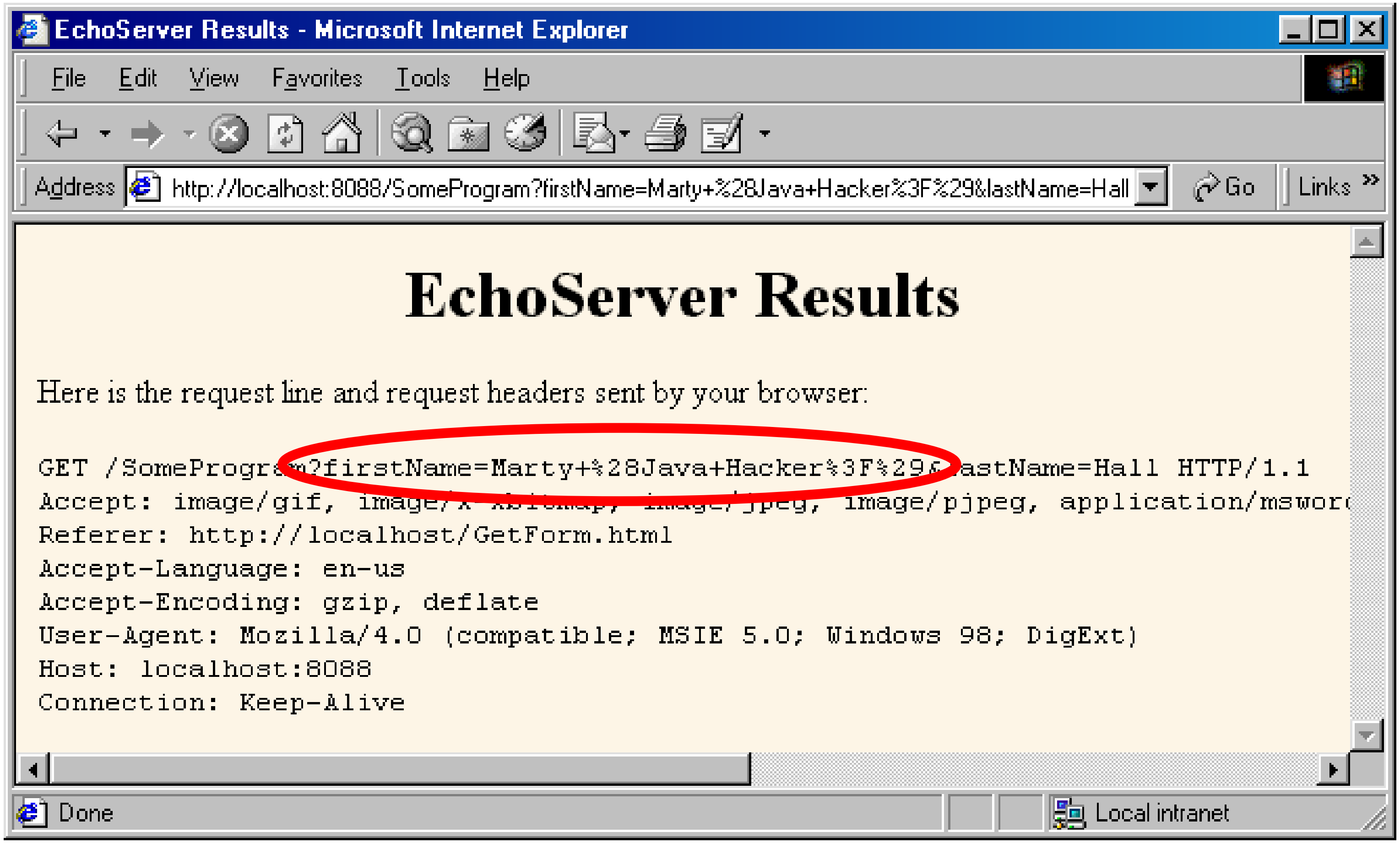
A Sample Form Using GET

First name: Marty (Java Hacker?)

Last name: Hall

Submit Query

URL Encoding: Result



http://localhost:8088/SomeProgram?firstName=Marty+%28Java+Hacker%3F%29&lastName=Hall

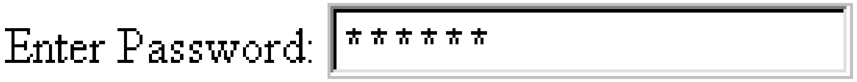
EchoServer Results

Here is the request line and request headers sent by your browser:

```
GET /SomeProgram?firstName=Marty+%28Java+Hacker%3F%29&lastName=Hall HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/mswor
Referer: http://localhost/GetForm.html
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 5.0; Windows 98; DigExt)
Host: localhost:8088
Connection: Keep-Alive
```

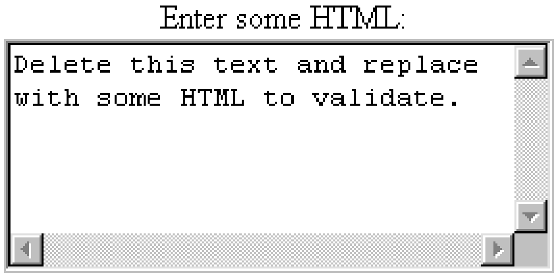
Text Controls

- **Textfields**
  - <INPUT TYPE="TEXT" NAME="…" …>
    - VALUE can give original value
- **Password Fields**
  - <INPUT TYPE="PASSWORD" NAME="…" …>
    - *Always* use POST



Enter Password: ******

- **Text Areas**
  - <TEXTAREA NAME="…"
    ROWS="…" COLS="…">
    …
    </TEXTAREA>



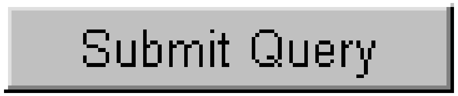
Enter some HTML:

Delete this text and replace with some HTML to validate.

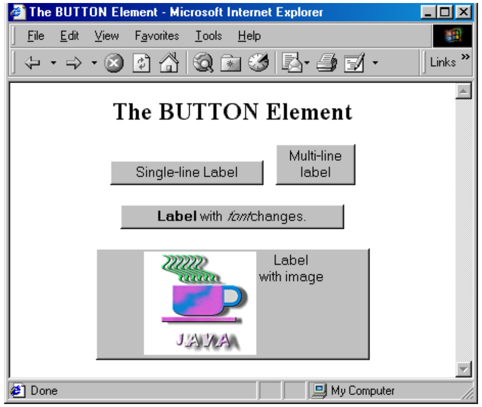
    - Interpretation of regular HTML tags turned off between <TEXTAREA...> and </TEXTAREA>

Push Buttons

- **Submit Buttons**
  - <INPUT TYPE="SUBMIT" …>
    - Use `NAME` if you have multiple buttons
    - Use `VALUE` to change button's label

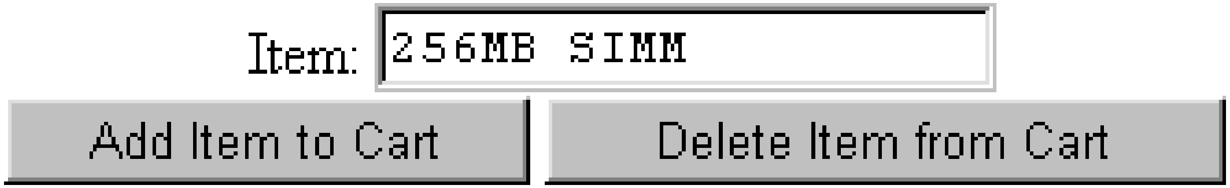


- **Reset Buttons**
  - <INPUT TYPE="RESET" …>
    - Use `VALUE` to change button's label
- **JavaScript Buttons**
  - <INPUT TYPE="BUTTON"
    onClick="someJavaScriptFunction()" …>
- **Fancy Buttons**
  - <BUTTON TYPE="SUBMIT" …>*HTML*</BUTTON>
    - Internet Explorer and Netscape 6 only



Using Multiple Submit Buttons

```
<CENTER>
Item:
<INPUT TYPE="TEXT" NAME="Item" VALUE="256MB SIMM"><BR>
<INPUT TYPE="SUBMIT" NAME="Add"
       VALUE="Add Item to Cart">
<INPUT TYPE="SUBMIT" NAME="Delete"
       VALUE="Delete Item from Cart">
</CENTER>
```

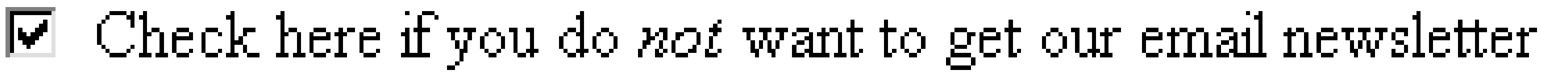

Item: 256MB SIMM

Check Boxes

- **Format**
  - <INPUT TYPE="CHECKBOX" NAME="…" …>
    - The `CHECKED` attribute makes it initially checked
    - Name/value pair sent only if checkbox is checked when form is submitted
- **Example code**

```
<P>
<INPUT TYPE="CHECKBOX" NAME="noEmail" CHECKED>
Check here if you do <I>not</I> want to
get our email newsletter
```

- **Example result**

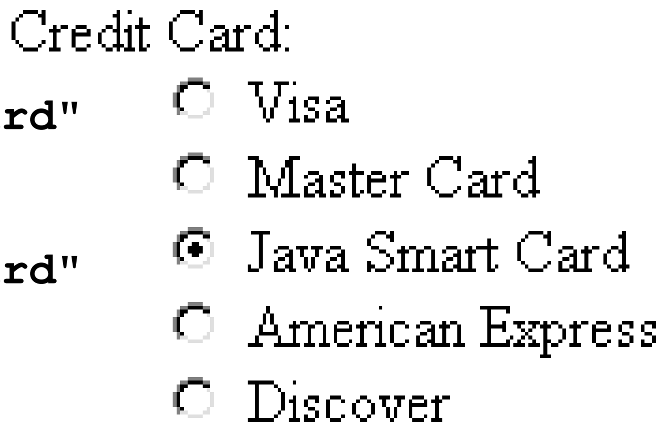


☑ Check here if you do *not* want to get our email newsletter

Radio Buttons

- **Format**
  - <INPUT TYPE ="RADIO"
    NAME="..." VALUE="..."...>
    - All radio buttons in a group should have same NAME
    - Only one button in a group can be pressed; pressing a different one causes previous one to pop out
- **Example**

```
<DL>
  <DT>Credit Card:
  <DD><INPUT TYPE="RADIO" NAME="creditCard"
             VALUE="visa">
      Visa
  <DD><INPUT TYPE="RADIO" NAME="creditCard"
             VALUE="mastercard">
      Master Card
  ...
</DL>
```



Credit Card:

○ Visa

○ Master Card

● Java Smart Card

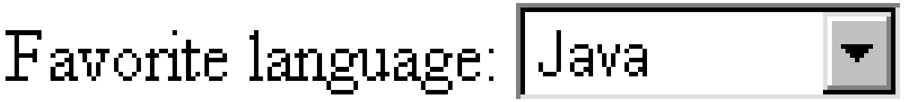
○ American Express

○ Discover

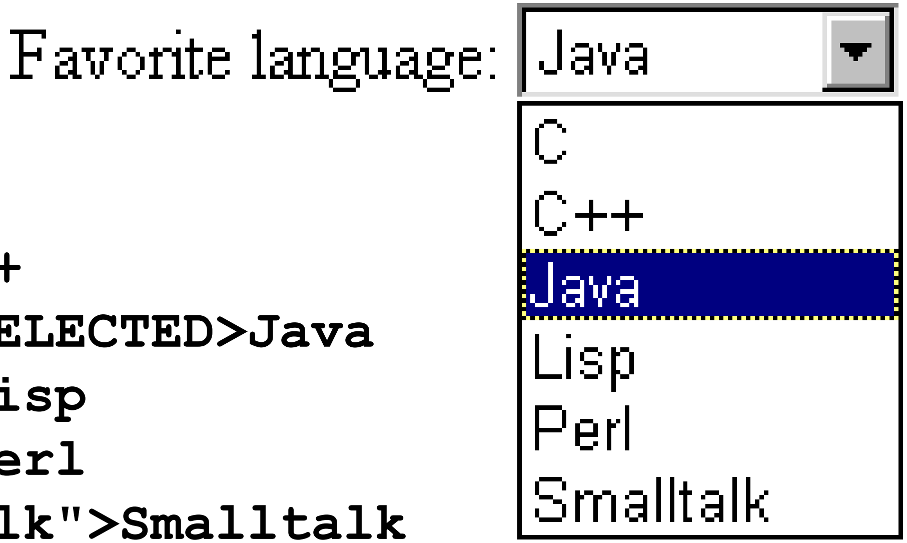
Combo Boxes

- **Format**
  - SELECT gives NAME
  - OPTION gives VALUE
- **Example**

```
Favorite language:
<SELECT NAME="language">
  <OPTION VALUE="c">C
  <OPTION VALUE="c++">C++
  <OPTION VALUE="java" SELECTED>Java
  <OPTION VALUE="lisp">Lisp
  <OPTION VALUE="perl">Perl
  <OPTION VALUE="smalltalk">Smalltalk
</SELECT>
```



Favorite language: Java



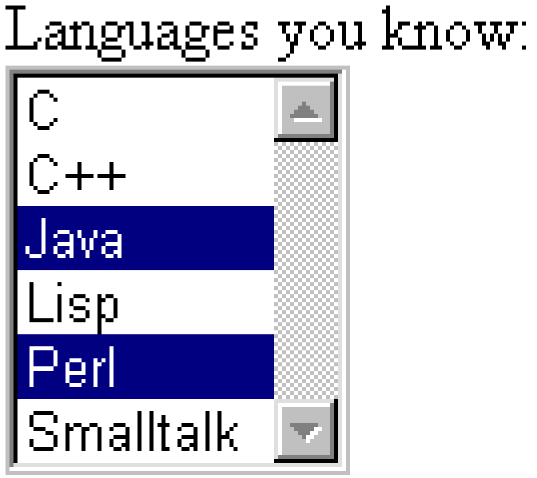
Favorite language: Java

C
C++
Java
Lisp
Perl
Smalltalk

List Boxes

- **Format**
  - Identical to combo boxes, but specify `MULTIPLE`
- **Example**

```
Languages you know:<BR>
<SELECT NAME="language" MULTIPLE>
  <OPTION VALUE="c">C
  <OPTION VALUE="c++">C++
  <OPTION VALUE="java" SELECTED>Java
  <OPTION VALUE="lisp">Lisp
  <OPTION VALUE="perl" SELECTED>Perl
  <OPTION VALUE="smalltalk">Smalltalk
</SELECT>
```



Languages you know:

C
C++
Java
Lisp
Perl
Smalltalk

Other Controls and Options

- **File upload controls**
  - Lets user select a file and send it to the server
- **Server-side image maps**
  - User clicks on an image and form gets submitted.
  - Form data gets sent as *name*.x=*x-pos*&*name*.y=*y-pos*
- **Hidden fields**
  - Preset `NAME` and `VALUE` sent with form submission..
- **Grouping Controls**
  - `FIELDSET` lets you visually group forms.
  - Internet Explorer and Netscape 6 only.
- **Tab order control**
  - `TABINDEX` (Internet Explorer and Netscape 6 only)

Summary

- **General process**
  - `FORM` uses `ACTION` to specify base URL
  - Input elements each have a `NAME`
  - User enters values
  - When form submitted, URL is baseURL?name1=value1&name2=value2&…
  - For `POST` requests, name/value pairs sent on separate line (not part of URL)
- **Textfields**
  - <INPUT TYPE="TEXT" …>
- **Submit Buttons**
  - <INPUT TYPE="SUBMIT" …>

core WEB programming

Questions?