Processing webforms

Server-Side Web Languages

Uta Priss
School of Computing
Napier University, Edinburgh, UK

Outline

Directory/File permissions for Server-Side Materials

A Simple Webform

HTML pages

- ▶ HTML pages must be stored in a dedicated directory on the webserver. The name of the directory depends on server settings. A common name is "public_html".

HTML pages

- ▶ HTML pages must be stored in a dedicated directory on the webserver. The name of the directory depends on server settings. A common name is "public_html".
- ▶ File permissions must be set to allow other users to view the HTML pages.
  - ▶ The user's home directory must be executable by others.
  - ▶ The public_html directory and its subdirectories must be readable by others.
  - ▶ Each HTML page must be readable by others.

Viewing HTML pages

The URL of a HTML page often consists of the server, a tilde, the
username, and the path and filename of the html page starting
below "public_html".

Example: A file

/home/username/public_html/hello.html

would be available at the URL

http://servername/∼username/hello.html

Script pages

- ▶ Script pages are usually stored in a dedicated directory under the public_html directory. The name of this directory depends on the language used and on the server settings. A common name is "cgi-bin".

Script pages

- ▶ Script pages are usually stored in a dedicated directory under the public_html directory. The name of this directory depends on the language used and on the server settings. A common name is "cgi-bin".
- ▶ File permissions must be set to allow script pages to be executed. Depending on the server settings, scripts are either executed
    - ▶ as a general www user ($\rightarrow$ set permissions to 755) or
    - ▶ as the owner of the CGI script ($\rightarrow$ set permissions to 700).

Viewing script pages

The URL of a script page is formed in the same manner as for
other HTML pages. In this case, the directory name (cgi-bin) is
included.

Example: A file

/home/username/public_html/cgi-bin/hello.pl

would be available at the URL

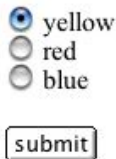http://servername/~username/cgi-bin/hello.pl

HTML content can be dynamically generated in response to a form which a user has filled in.

For example, a user could be asked to check one out of three checkboxes and then click a submit button:

HTML content can be dynamically generated in response to a form which a user has filled in.

For example, a user could be asked to check one out of three checkboxes and then click a submit button:

◉ yellow
◯ red
◯ blue

submit

After submitting the button the user could then be presented with:

Hello! Your choice was yellow

Implementing this example

Two HTML pages need to be created for this example:

Implementing this example

Two HTML pages need to be created for this example:

▶ An HTML page which contains the webform (i.e., the radio
buttons and the submit button).

Implementing this example

Two HTML pages need to be created for this example:

► An HTML page which contains the webform (i.e., the radio buttons and the submit button).

► A dynamically generated HTML page, which uses the colour which the user had selected.

Implementing this example

Two HTML pages need to be created for this example:

► An HTML page which contains the webform (i.e., the radio buttons and the submit button).

► A dynamically generated HTML page, which uses the colour which the user had selected.

The second page is generated by a script. The script stores the parameters from the webform as variables, which can then be used within the HTML:

```
<p>Hello! Your choice was
<font color=$formparameter>$formparameter</font><p>
```