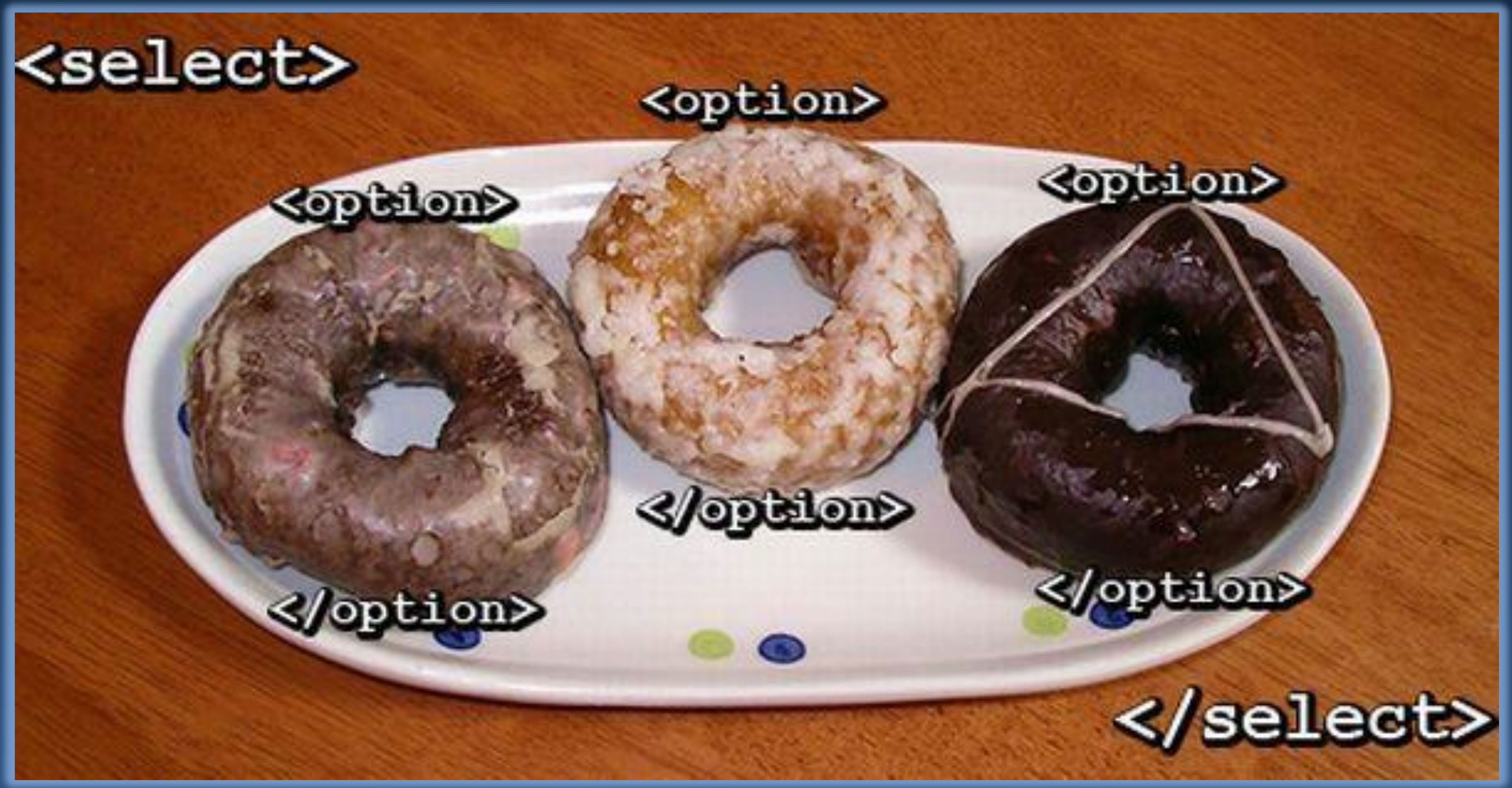
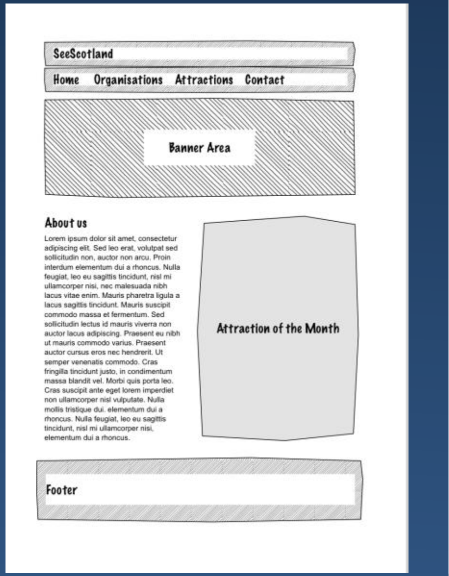


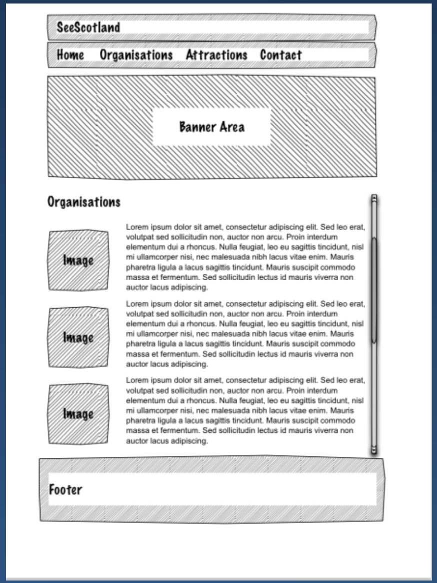
Web Technology

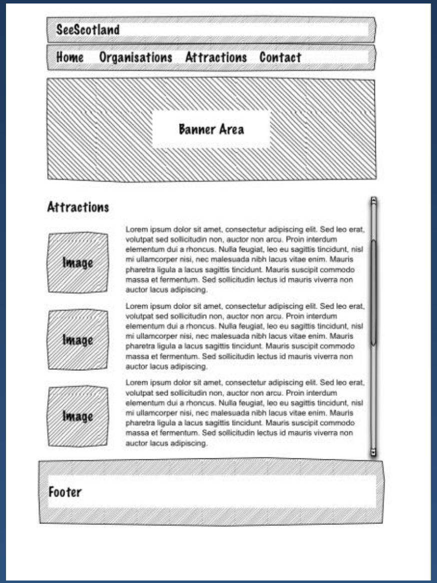
Lecture 8: HTML Forms

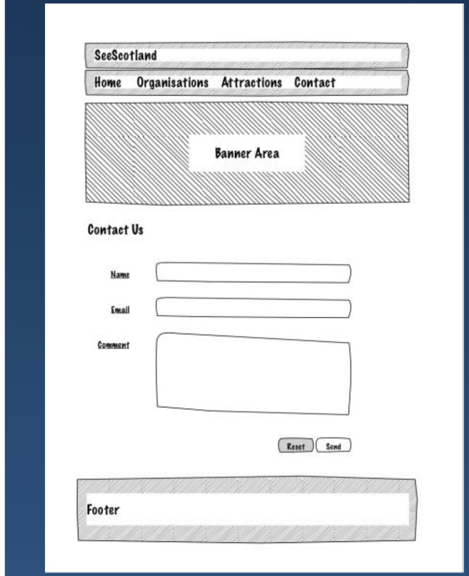
Continuous Assessment

- Now available on MyAberdeen
- Due 1st April, 16:00



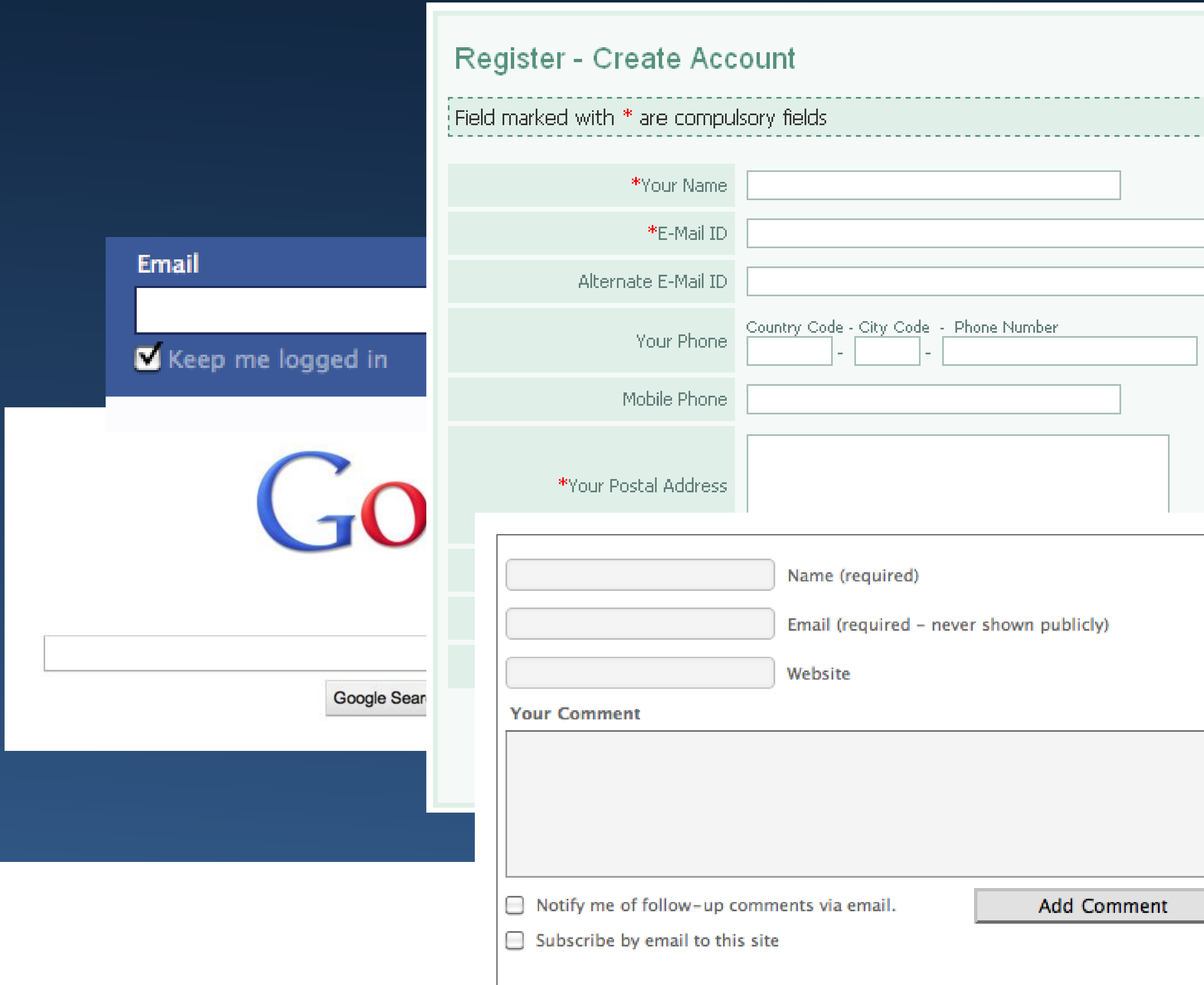






Web Forms: Usage

- Search engine
- Login page
- Register to a Website
- New Blog Post
- Many more



Email

☑ Keep me logged in

Google Sear

Register - Create Account

Field marked with * are compulsory fields

*Your Name

*E-Mail ID

Alternate E-Mail ID

Your Phone Country Code - City Code - Phone Number

Mobile Phone

*Your Postal Address

Name (required)

Email (required – never shown publicly)

Website

Your Comment

□ Notify me of follow-up comments via email.

□ Subscribe by email to this site

Add Comment

HTML Forms

- HTML forms are used to pass data to a server.
- A form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain select lists, textarea, fieldset, legend, and label elements.
- The `<form>` tag is used to create an HTML form:

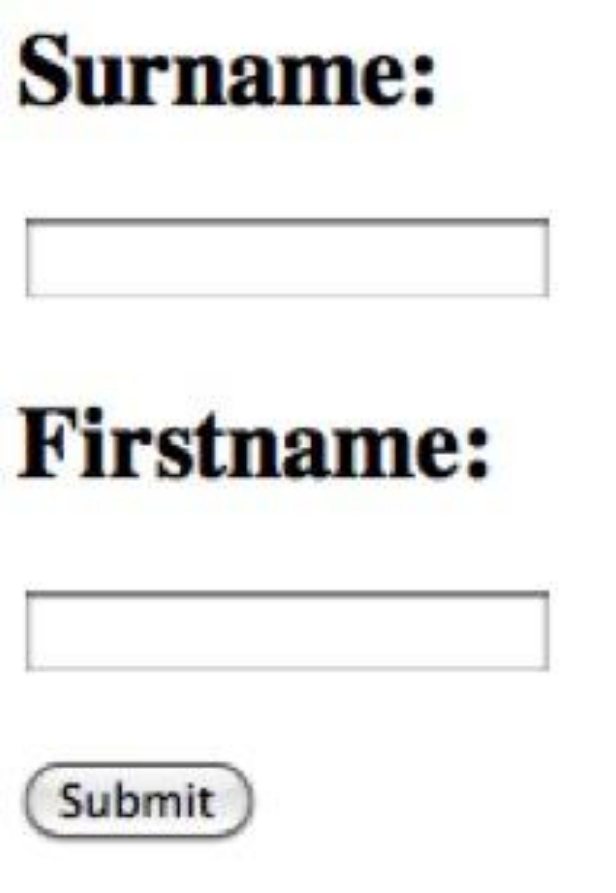
```
<form>
    input elements
</form>
```

Simple Form Creation

```
<form>
   <h1>Surname: </h1>
   <input type="text" name="surname" id="surname" />
   <h1> Firstname:  </h1>
   <input type="text" name="firstname‟ id="firstname" />
   <br /> <br />
   <input type="submit" />
</form>
```

Example:

http://jsfiddle.net/edo77uk/4eBAm/



Surname:

Firstname:

HTML Form Variables

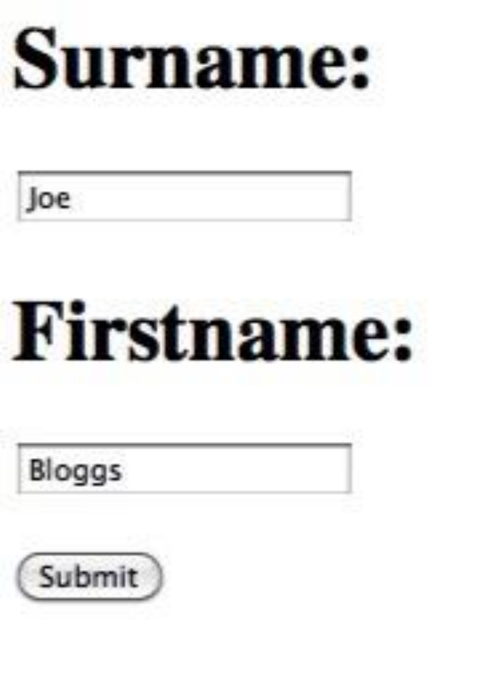
- A form contains **one or more variables**. When the user **fills in** the form, **values are assigned** to these **variables**.

```
<input type="text" name="surname" id="surname" />
<input type="text" name="firstname" id="firstname" />
```

- When the user clicks the **submit button**, the **set of variable names** and **corresponding values** are sent to the Web server in a HTTP request.
- The Web server can **extract the set of variables and values** from the HTTP request, and can do something with them.

Variables: names and values

```
<input type="text" name="surname" id="surname" />
<input type="text" name="firstname" id="firstname" />
```



Surname:

Joe

Firstname:

Bloggs

Variable Name	Variable Value
`surname`	Bloggs
`firstname`	Joe

HTML Forms

- There may be several forms on a web page
- You cannot nest forms within forms
- The form area of a page enclosed in

  `<form> </form>`

  tags
- The form tag has several attributes

Attribute	Variable Value
`name`	Name of the form
`autocomplete`	Specifies whether autocomplete can be used (from browser)
`novalidate`	Says that the form should not be validated before sending to server
`id`	Unique identifier of the form
`method`	How the form will be sent to the server
`action`	Specify the script on the server which will process the form

Form Methods

- The **method** attribute on the **form** tag specifies **how** the Web Browser should **send the data** to the Web server.
- **Two options**:
  - **GET**: pass the data in a **HTTP GET** request
  - **POST**: pass the data in a **HTTP POST** request
- In a **HTTP GET** request, the browser appends the form **data** to a URL. For example:

http://homepages.abdn.ac.uk/m.dennis/pages/teaching/echo.php?name=joe+bloggs&id=1234

Form Methods

```
<form method="post" action= "mail.php">
```

- This form uses the **HTTP POST** method: instead of sending the data appended to the URL, the data is sent in a **separate part of the HTTP message**.
- **HTTP GET vs POST** methods for forms:
  - **GET** allows a **user** to type in the **URI manually, including the data**
  - **GET** has security issues
  - **POST** is better for **large sets of form data** and for processes that effects the "state of the world" e.g. a database
  - Most forms should use **POST**

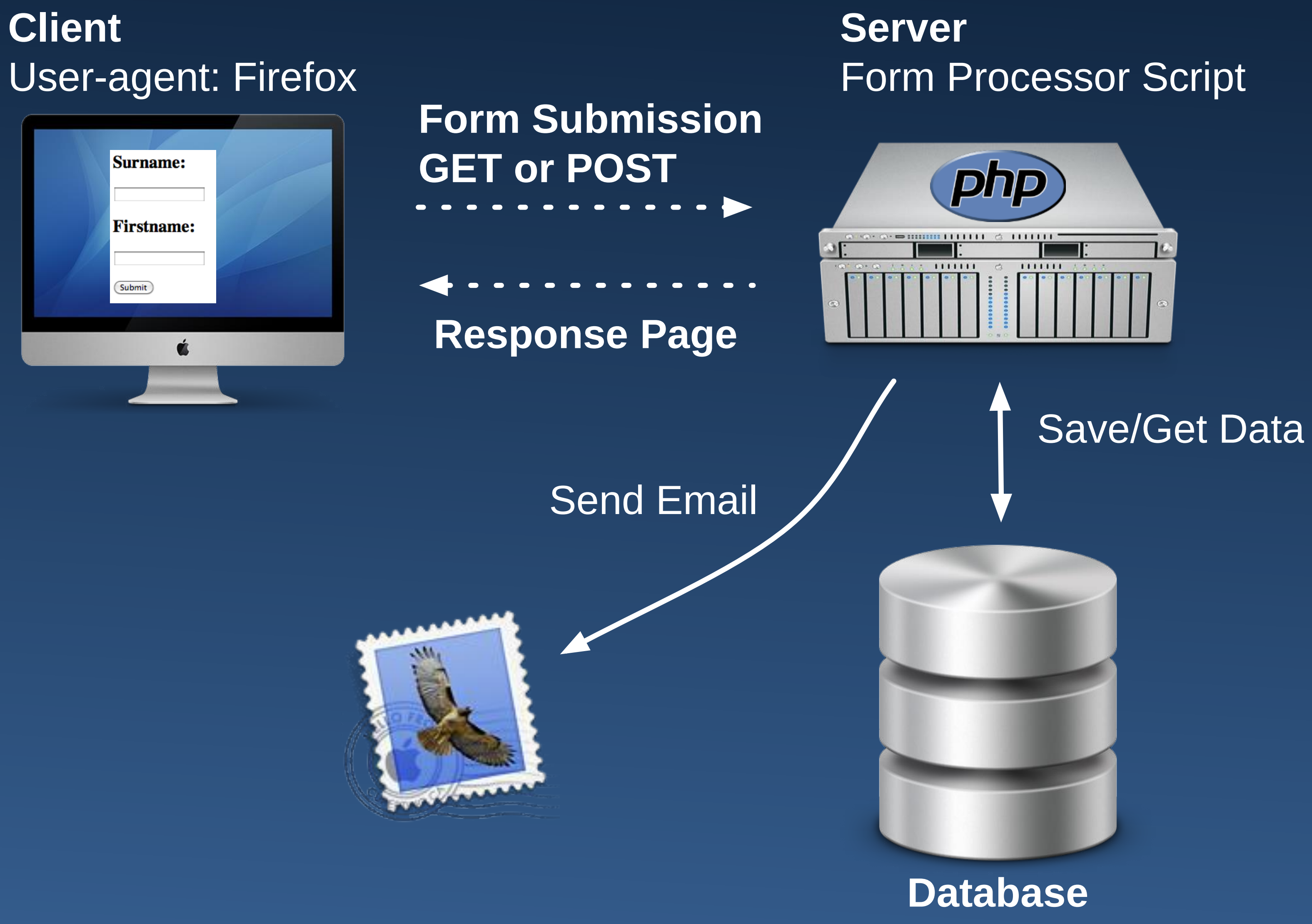
Form Actions

```
<form method="get" action= "doquery.php">
```

- The action attribute on the form tag specifies what the Web server should do with the form data.
  - Pass the data to a script or program
- In the example, the action is to pass the data to a script called doquery.php [this is a PHP script].
- The script is passed the variables and values, and can then process them.

http://homepages.abdn.ac.uk/f.cerutti/pages/teaching/echo.php?name=joe&id=1234

HTTP Form Processing



Client
User-agent: Firefox

Form Submission
GET or POST

Response Page

Server
Form Processor Script

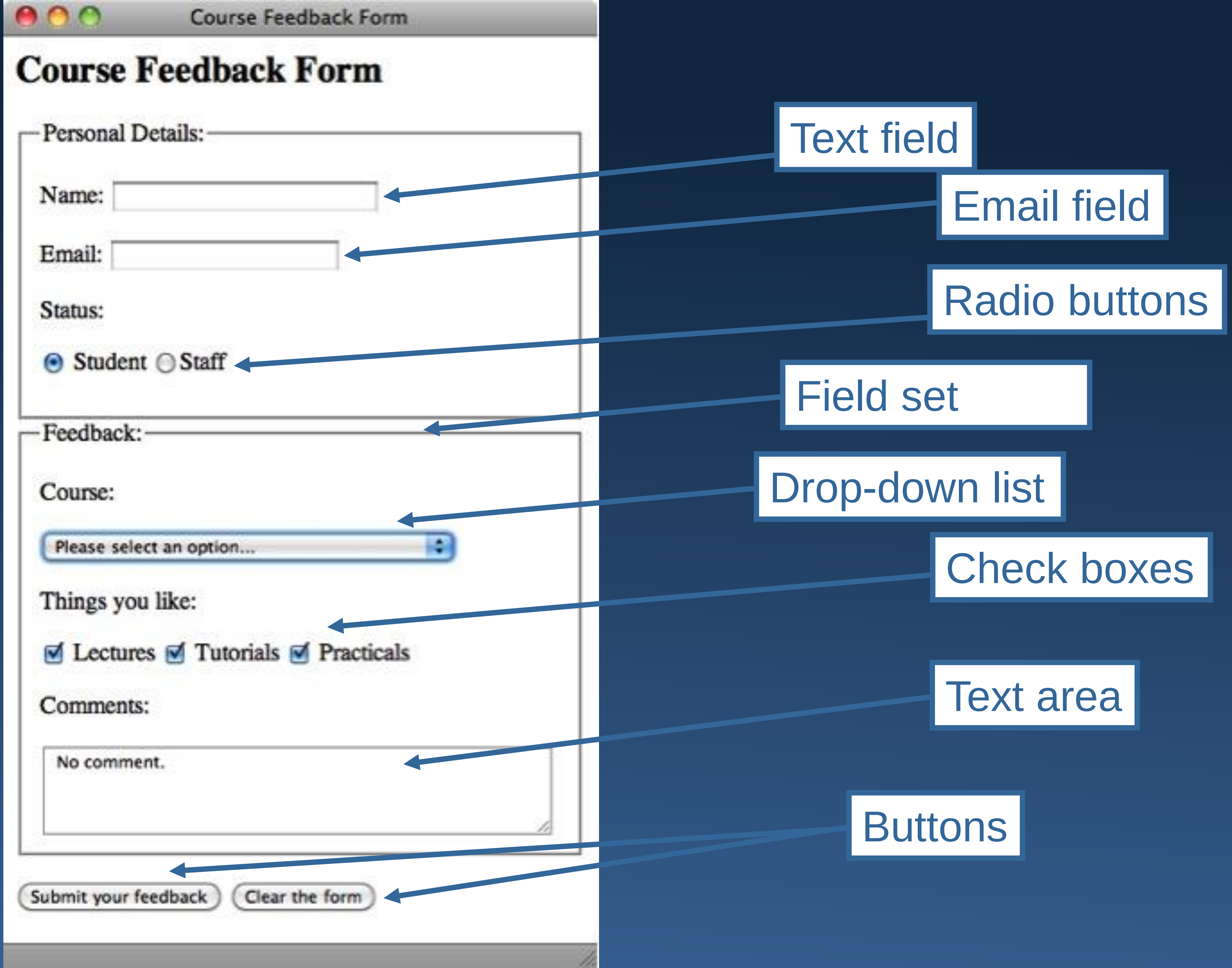
Send Email

Save/Get Data

Database

Form Tags

Tag	Description
`<form>`	Defines an HTML form for user input
`<input />`	Defines an input control
`<textarea>`	Defines a multi-line text input control
`<label>`	Defines a label for an input element
`<fieldset>`	Defines a border around elements in a form
`<legend>`	Defines a caption for a fieldset element
`<select>`	Defines a select list (drop-down list)
`<optgroup>`	Defines a group of related options in a select list
`<option>`	Defines an option in a select list
`<button>`	Defines a push button



Course Feedback Form

Course Feedback Form

Personal Details:

Name:

Email:

Status:

● Student ○ Staff

Feedback:

Course:

Please select an option...

Things you like:

☑ Lectures ☑ Tutorials ☑ Practicals

Comments:

No comment.

Submit your feedback

Clear the form

Text field

Email field

Radio buttons

Field set

Drop-down list

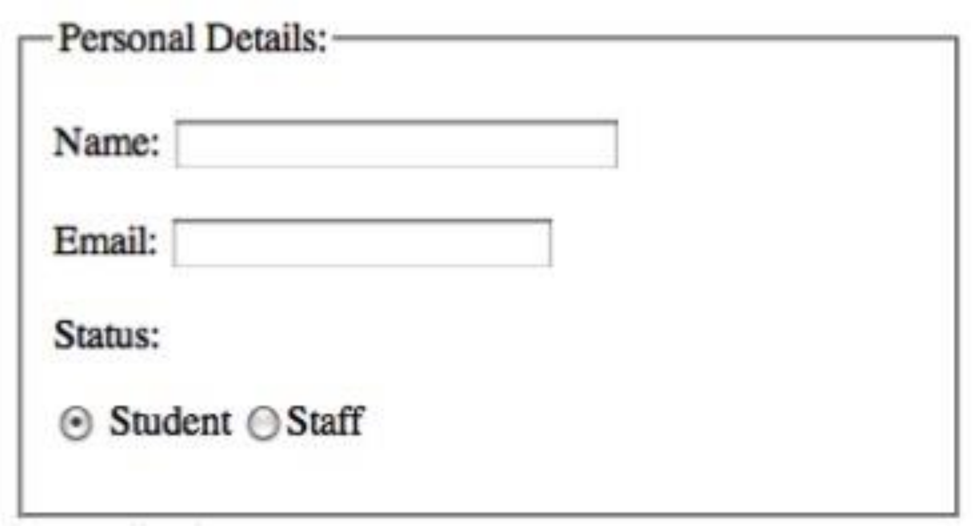
Check boxes

Text area

Buttons

Example form [1]

```
<form method="post" >
<fieldset>
 <legend>Personal Details:</legend>
 <p>Name: <input name="name" id="name" type="text" size="30" /></p>
 <p>Email: <input name="email" id="email" type= "email" size="25" />
 </p>
 <p>Status:</p>
 <p>
 <input name="status" id="status" type="radio" value="student"
                                       checked  = "checked" /> Student
 <input name="status" id="status" type="radio"
 </p>
</fieldset>
```



Personal Details:

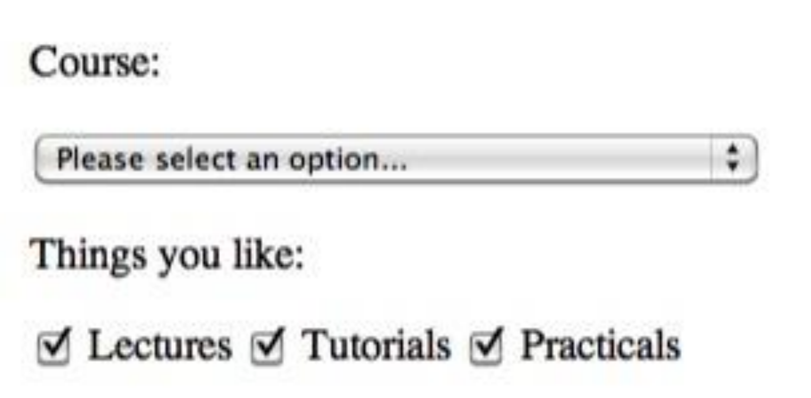
Name:

Email:

Status:

● Student ○ Staff

Example form [2]



Course:

Please select an option...

Things you like:

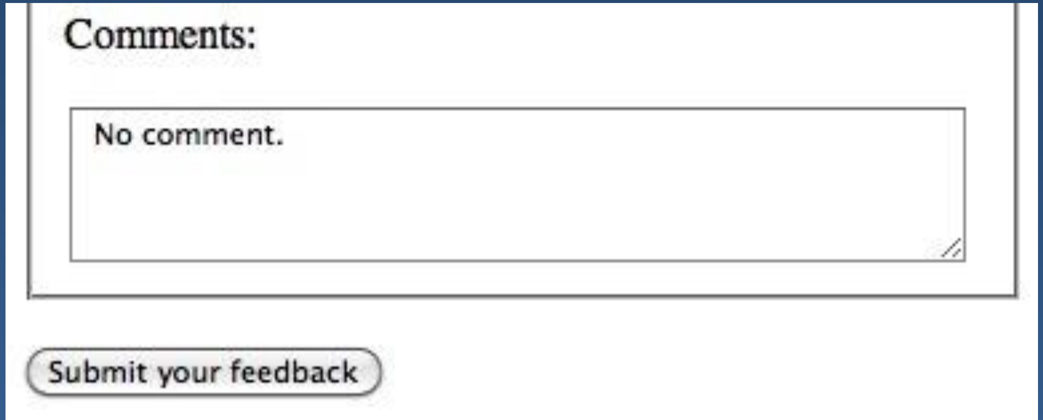
☑ Lectures ☑ Tutorials ☑ Practicals

```
<fieldset>
    <legend>Feedback:</legend>
<p>Course:</p>
<select name="course" id="course">
  <option value  = "" disabled selected>Please select an option... </option>
  <option value ="CS5020">CS5020 Internet Information Systems </option>
  <option value ="CS5035">CS5035 Introduction to Database Systems </option>
  <option value ="CS5036">CS5036 Introduction to Programming in Java </option>
  <option value ="CS5037">CS5037 System analysis and Design </option>
</select>
</p>

<p>Things you like:</p>
<input name="lectures" id="lectures" type="checkbox" checked = "checked" value = "yes"/>
Lectures
<input name="tutorials" id="tutorials" type="checkbox" checked = "checked" value =
"yes"/> Tutorials
<input name="practicals"  id="practicals" type="checkbox"  checked = "checked" value =
"yes"/> Practicals
</p>
```

Example form [3]

```
<p>Comments:</p>
<textarea name="comments" id="comments" rows="4" cols="45">
  No comment.
</textarea>
</fieldset>
<p>
<input type="submit" value="Submit your feedback" />
<!-- <input type="reset" value="Clear the form" /> -->
</p>
<input type="hidden" name="recipient" id="recipient"
                                        value="csc204" />
<input type="hidden" name="subject" id="subject"
                                 value="Course Feedback" />

</form>
</body>
</html>
```

Why would anyone want this?!


Comments:

No comment.

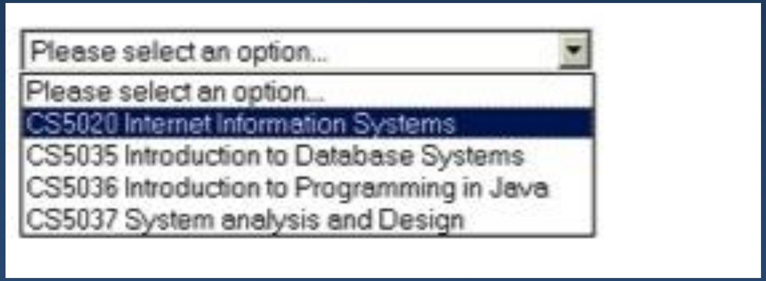
The `<fieldset>` Tag

```
<fieldset>
 <legend>Personal Details:</legend>
Name: <input name="name" id="name" type="text"
                                         size="30" />
Email: <input name="email" id="email" type="password"
                                         size="25" />
<input name="status" id="status" type="radio" value="student"
                         checked  = "checked" /> Student
<input name="status" id="status" type="radio"
                               value="staff" /> Staff
</fieldset>
```

- The `<fieldset>` tag is used to logically group together elements in a form.
- The `<fieldset>` tag draws a box around the related form elements.
- The `<legend>` tag defines a caption for the fieldset element.

The `<select>` Form Element

- The select form element provides a pull-down list of choices.
- The variable name is given as an attribute to the select tag.
- Each option tag specifies a different possible value for the variable.
- The default choice is marked with a selected attribute with the value selected.

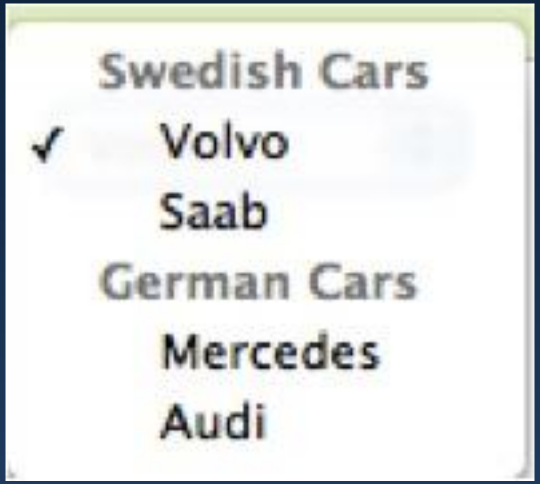


Please select an option...

Please select an option...
CS5020 Internet Information Systems
CS5035 Introduction to Database Systems
CS5036 Introduction to Programming in Java
CS5037 System analysis and Design

```
<select name="course" id="course">
  <option value  = "" disabled selected>Please select an option... </option>
  <option value ="CS5020">CS5020 Internet Information Systems </option>
  <option value ="CS5035">CS5035 Introduction to Database Systems </option>
  <option value ="CS5036">CS5036 Introduction to Programming in Java </option>
  <option value ="CS5037">CS5037 System analysis and Design </option>
</select>
```

The `<optgroup>` Tag



Swedish Cars

✓ Volvo

Saab

German Cars

Mercedes

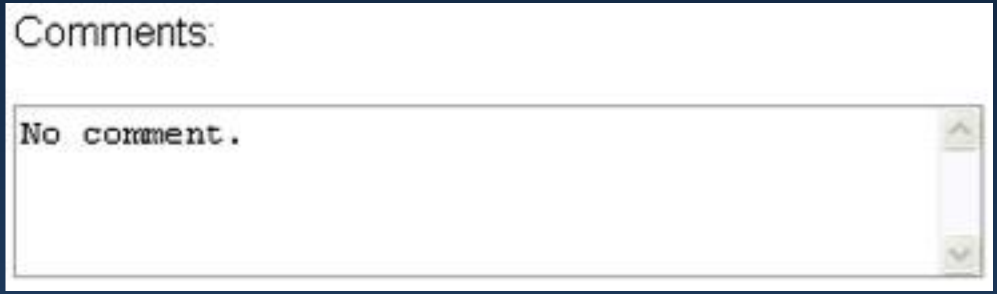
Audi

```
<select>
  <optgroup label="Swedish Cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
  </optgroup>
  <optgroup label="German Cars">
    <option value="mercedes">Mercedes </option>
    <option value="audi">Audi</option>
  </optgroup>
</select>
```

- The `<optgroup>` tag is used to group together related options in a select list.
- If you have a long list of options, groups of related options are easier to handle for the user.

The `<textarea>` Form Element

```
<textarea name="comments"
    id ="comments" rows="4" cols="45">
   No comment.
</textarea>
```



Comments:

No comment.

- The textarea form element is similar to the text element, except the user can enter multiple lines of text.
- A default value can be specified by putting it as the content of the textarea element.
- In addition to specifying the size of the text area [rows & cols attributes], attributes can be used to specify whether the text wraps, and the maximum size of the text.

The "password" Form Element Type

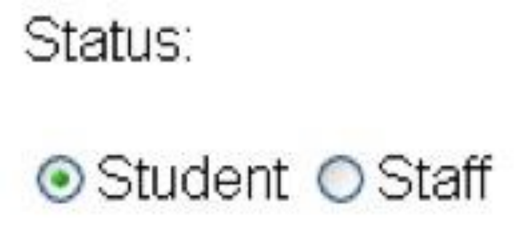
```
<input name= "password" id= "password"
type="password" size="25" />
```

- The **password** form element is similar to the text element, except the **data** the user types is **not shown,** only asterisks.

- But it is sent to the web server in **plain text**, so hackers could intercept it!
  - Assuming you’re using HTTP rather than HTTPS

The "radio" Form Element Type

```
<input name="status"  id="status" type="radio"
value="student"
checked = "checked" /> Student
<input name="status"  id="status" type="radio"
value="staff" /> Staff
```



Status:

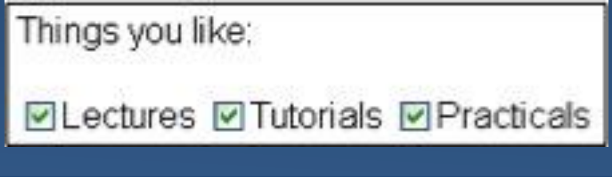
● Student ○ Staff

- The radio form element is used to create a set of radio buttons where only one option can be selected at a time.
- If one of the buttons is to be selected as the default, it needs to have the attribute checked with the value checked.

The "checkbox" Element Type

```
<input name="lectures" id="lectures" type="checkbox"  value
= yes" checked="checked"/>  Lectures
<input name ="tutorials" id="tutorials" type="checkbox"
value = "yes" checked="checked"/> Tutorials
<input name ="practicals" id="practicals" type="checkbox"
value = "yes" checked="checked"/>  Practicals
```

- The checkbox form element type is similar to the radio element type, except that the user can check multiple boxes.
- As the choices are effectively independent, they are usually but not always given different variable names.
- The checked attribute may be set to "checked" to show the item has been selected.



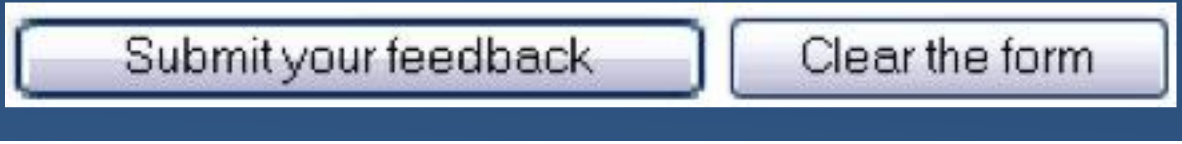
Things you like:

☑Lectures ☑Tutorials ☑Practicals

Button Form Elements

```
<input type="submit" value="Submit your feedback" />
<input type="reset" value="Clear the form" />
```

- There are two kinds of button:
  - The submit type triggers the form action
  - The reset type simply clears the form and resets any default values
- A form can have multiple submit buttons, each with a different value - the value is passed to the Web server with the submit variable, just like any other variable-value pair.

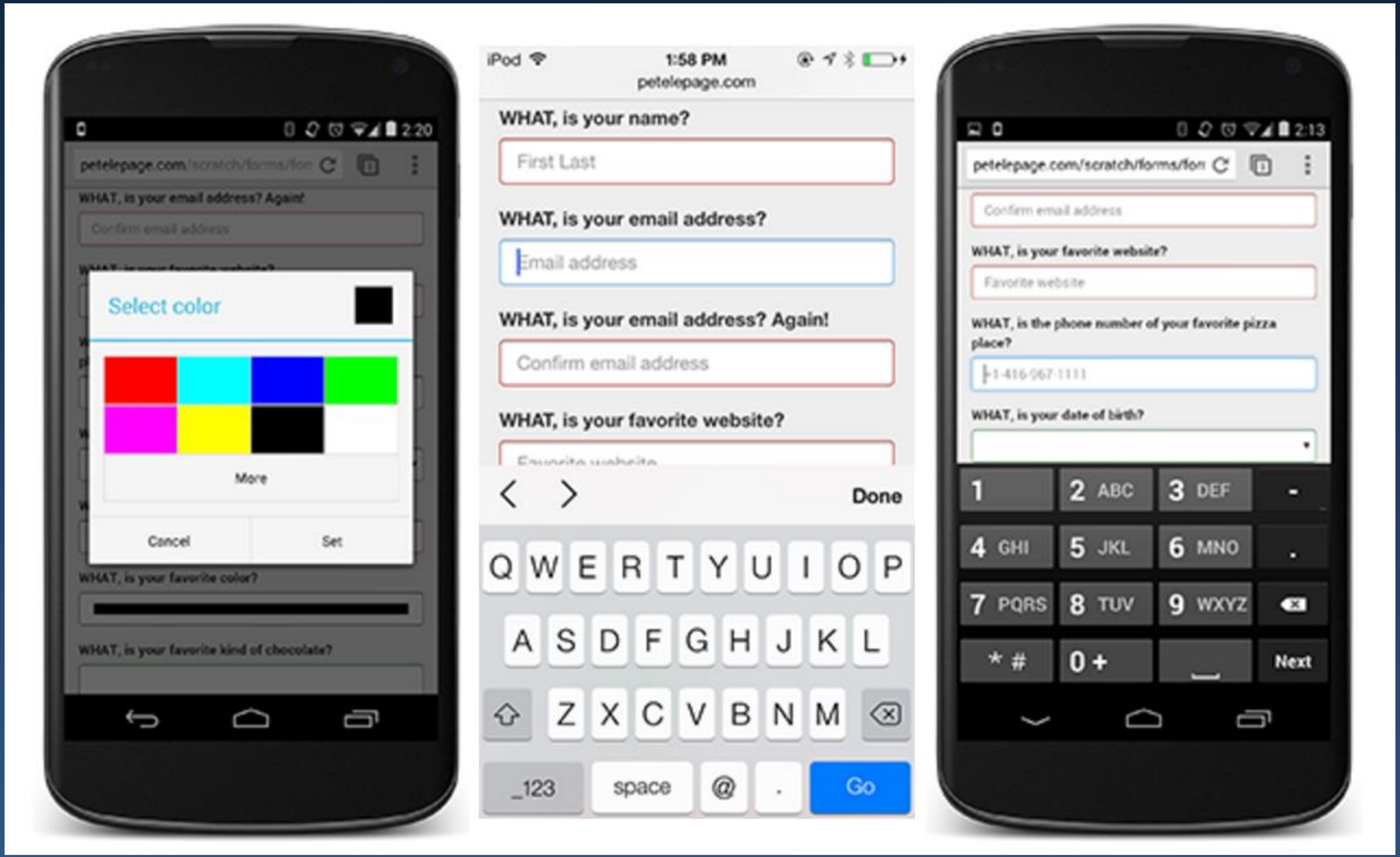


<input> types

- HTML5 has added many new <input> types to make everyone's lives easier
- http://www.w3schools.com/tags/att_input_type.asp

<input type =" ">

button	Defines a clickable button	password	Defines a password field (characters are masked)
checkbox	Defines a checkbox	radio	Defines a radio button
color	Defines a color picker	range	Defines a control for entering a number whose exact value is not important (like a slider control)
date	Defines a date control (year, month and day)	reset	Defines a reset button
datetime-local	Defines a date and time control	search	Defines a text field for entering a search string
email	Defines a field for an e-mail address	submit	Defines a submit button
file	Defines a file-select field and a "Browse..." button (for file uploads)	tel	Defines a field for entering a telephone number
hidden	Defines a hidden input field	text	Default. Defines a single-line text field (default width is 20 characters)
image	Defines an image as the submit button	time	Defines a control for entering a time (no time zone)
month	Defines a month and year control (no time zone)	url	Defines a field for entering a URL
number	Defines a field for entering a number	week	Defines a week and year control (no time zone)



WHAT, is your email address? Again!

Confirm email address

WHAT, is your favorite color?

WHAT, is your favorite kind of chocolate?

WHAT, is your name?

First Last

WHAT, is your email address?

Email address

WHAT, is your email address? Again!

Confirm email address

WHAT, is your favorite website?

Favorite website

Confirm email address

WHAT, is your favorite website?

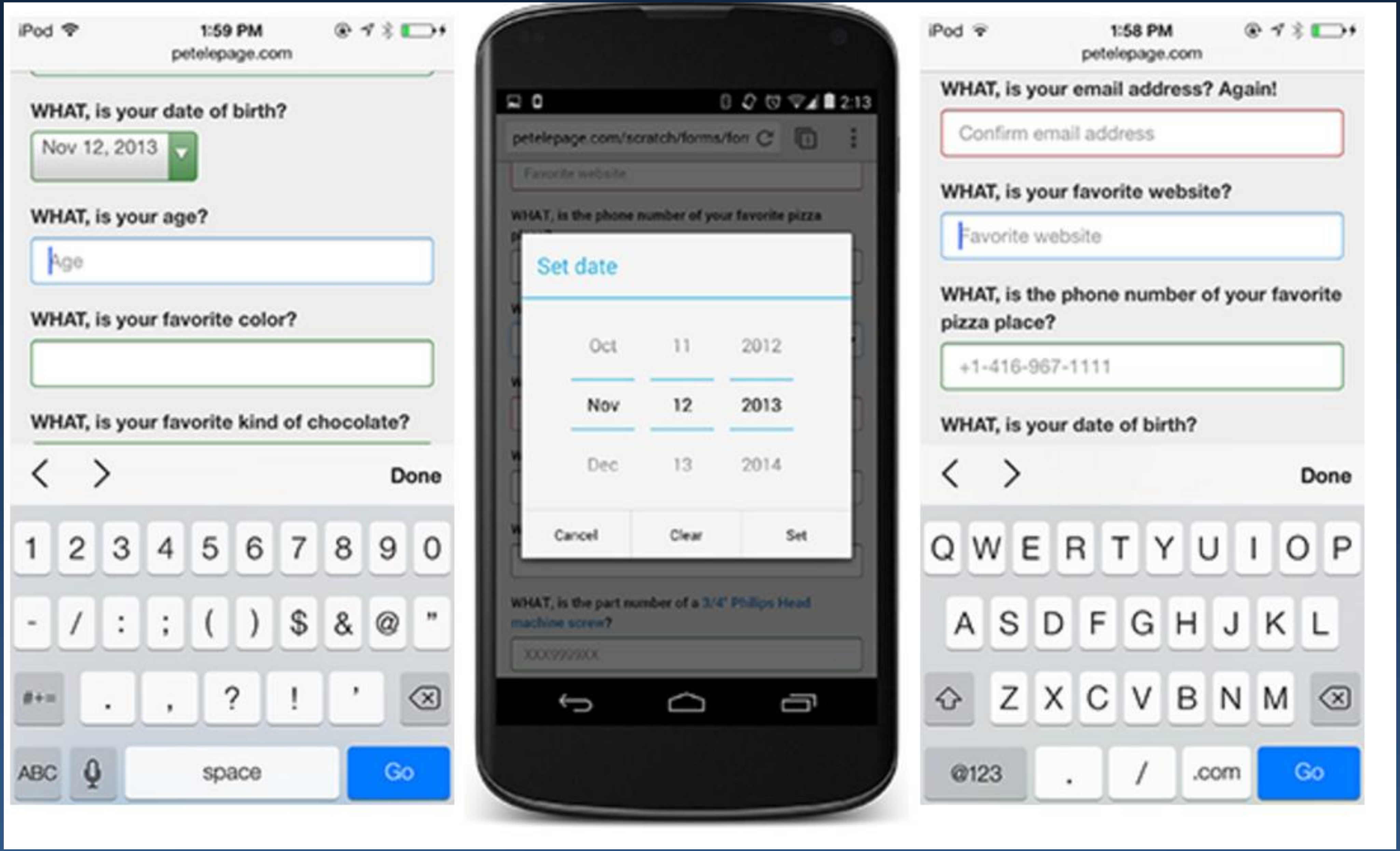
Favorite website

WHAT, is the phone number of your favorite pizza place?

+1-416-967-1111

WHAT, is your date of birth?

http://www.html5rocks.com/en/tutorials/forms/html5forms/



iPod 1:59 PM
petelepage.com

WHAT, is your date of birth?

Nov 12, 2013

WHAT, is your age?

Age

WHAT, is your favorite color?

WHAT, is your favorite kind of chocolate?

petelepage.com/scratch/forms/for

Favorite website

WHAT, is the phone number of your favorite pizza place?

Set date

Oct	11	2012
Nov	12	2013
Dec	13	2014

Cancel Clear Set

WHAT, is the part number of a 3/4" Philips Head machine screw?

XXX999XX

iPod 1:58 PM
petelepage.com

WHAT, is your email address? Again!

Confirm email address

WHAT, is your favorite website?

Favorite website

WHAT, is the phone number of your favorite pizza place?

+1-416-967-1111

WHAT, is your date of birth?

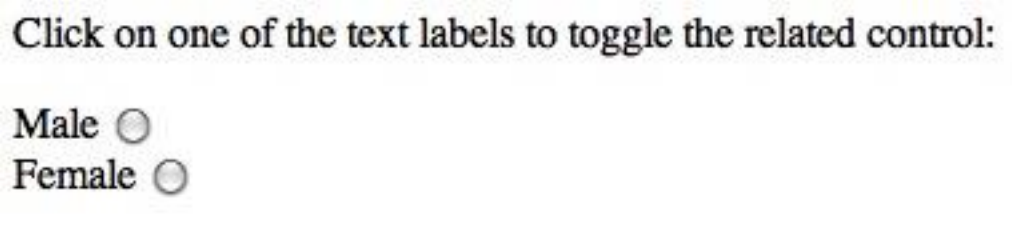
http://www.html5rocks.com/en/tutorials/forms/html5forms/

New Input Types

- Older browsers (looking at you IE 9) don't support them
- However, they degrade gracefully into text fields
- So they can be safely used

The <label> Tag

```
<form>
  <label for="male">Male</label>
  <input type="radio" name="sex" id="male" />
  <br />
  <label for="female">Female</label>
  <input type="radio" name="sex" id="female" />
</form>
```



Click on one of the text labels to toggle the related control:

Male ○
Female ○

- The <label> tag defines a label for an input element.
- The label element does not render as anything special for the user. However, it provides a usability improvement because it toggles the control when clicked.
- The for attribute of the <label> tag should be equal to the id attribute of the related element to bind them together.

Hidden Form Elements

- Often, it is useful to add hidden variables to a form, with pre-set values
- This technique is typically used for two purposes:
  - keeping track of previously-acquired information
  - supplying additional information to the server-side script

- The information is invisibly added to the form as variables of type hidden:

```
<input type="hidden" name="recipient" id="recipient"
    value= "csc204" />
<input type="hidden" name="subject" id="name"
    value="Course Feedback" />
```

- These variables are additional information needed by the mail.php script.

Form Validation

- You usually want to make some fields mandatory
- Username and password, name, etc.
- Some are optional (middle names, street(!))
- How do we achieve this?

Form Validation

- Client-side:
  - Check that required fields are completed prior to form submission
  - JavaScript
  - HTML5 (add "required" to the fields that are mandatory)
    - `<input type ="email" required />`
    - Can also add min and max values to numeric fields
    - `<input type = "number" min ="0" max ="100"/>`
- Server side
  - Check that the fields in the POST array are set before doing anything with the data
  - Redirect users to the back to the page containing the form if empty fields are present

Client Side or Server Side Validation?

- BOTH.
- Older browsers don't support the HTML5 required attribute
- People have JavaScript switched off
- Input fields can be manipulated by developer tools in browsers

- If you don't check server-side, you are vulnerable to hacking, data corruption, or worse

Summary

In this lecture

- HTML Forms

What next?

- HTML and CSS
- Intro to Javascript