Internet Technology - Session Seven
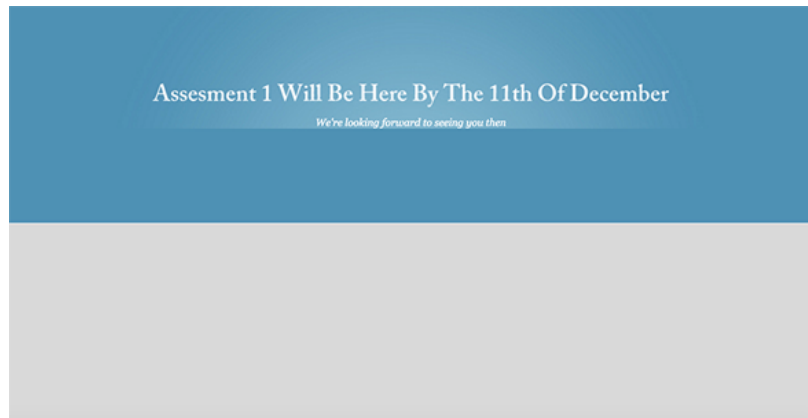
- [FTP](#)

- [Forms](#)

File Transfer Protocol (FTP)

FTP is a standard network protocol used for exchanging files between a computer and web server. **We're going to be using this protocol to upload assessment1 to the Edward2 web sever**

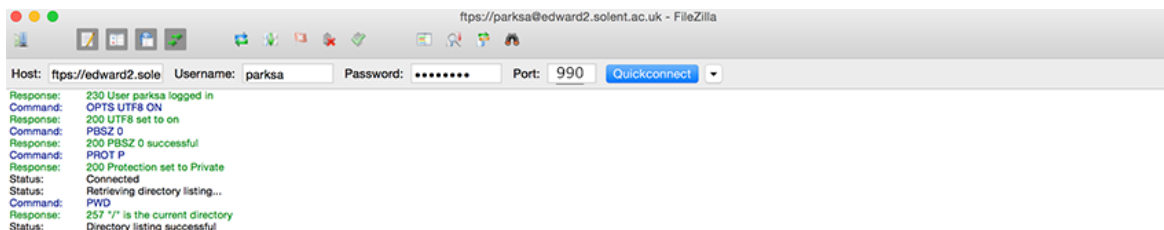## Session Practical 1 - Uploading Files to The Edward2 Server (30 mins)

The goal of this practical is understand the process of transferring files between your computer and a web server, in this case edward2.

1. Your FTP account security credentials should have been emailed to you, check your university account for an email from Joe Appleton, titled **Internet Technology (SWD400) - Sever Access**. Open up this email and take a note of your username and password. **Speak to the tutor if you don't have the email**

2. Ok, now we need a website to actually transfer to the server. Download the **ftp demo site** zip file from myCourse. Unzip the file and open the `index.html` file in your browser, the website should look like the below image:

1. Next we need to connect to our web server. In order to do this we need to use a FTP client, we're going to use [FileZilla](). FileZilla should already be installed on the university computers. Open up FileZilla, **note how the port in this case is set to 990**. Enter your security credentials along with the host information and click quick connect. You should see "Directory listing successful" in the command prompt.

<div align="center">

host: ftps://edward2.solent.ac.uk
username: <your username emailed to you>
password: <your password>
port: 990

</div>

1. Now we need to actually transfer our site to the web server. **Important** the site needs to be transferred to `public_html` , not the root directory. Click on `public_html` and locate your site in the left hand pane, drag across **all** your html files and assets into `public_html` .

2. Now try and access your website, the url is **http://edward2.solent.ac.uk/~<username emailed to you>**. When the security dialog box pops up enter your **university user name and password**. You should now have a live holding page, if everything worked correctly.

3. **Note**, you can start playing with and uploading assessment1 from now. You can delete remote files just like local versions. Also, you'll be able to access edward2 from any location, please take advantage of this.

Forms

- We've all encountered forms, they're how we gather information from users. This information can then be sent to a web server.

- The form controls (e.g. text field, checkbox, radio buttons etc) must live inside `<form>` ... `</form>` element

- The `<input>` Element, handles most form controls

We use the `type` attribute to specify the type of control e.g. type = resetlfilelhiddeimagelbuttonltext"
`<input type="text"></input>`

- The `<input>` Element, usually has a corresponding `<label>` .. `</label>` element.

- An example form using `<input type="text">`

```
<h1>Form</h1>
<form>
    <p> <label>Name:</label> <input type="text" name="Name"> </p>
    <p> <label>Email:</label> <input type="text" name="Email"> </p>
    <input type="submit" name="Submit" value="Submit">
</form>
```

Form

Name: [   ]

Email: [   ]

[Submit]

Input - Radio Box


```
<label>  Please enter rating    </label>
 <input type="radio" name="rating" value="excellent"> Excellent <br/>
 <input type="radio" name="rating" value="good"> Good <br/>
 <input type="radio" name="rating" value="bad"> Bad <br/>
```

- The buttons in the same group have the same name

- If the user selects excellent, the server will receive: rating="excellent"
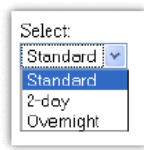
- Radio buttons let a user select ONE choice

Input - Check Box


```
<label>  How do you get to university?    </label>
<input type="checkbox" name="vehicle" value="Bike" checked="checked"> By Bike<br>
<input type="checkbox" name="vehicle" value="Car">  By Car <br>
<input type="checkbox" name="vehicle" value="Bus">  By Bus <br>
```

- Similar to check boxes, however they allow MULTIPLE choices

Drop Down List


```
<label>Shipping method: </label>
<select name="shipping">
              <option value="1"> Standard</option>
              <option value="2"> 2-day</option>
              <option value="3"> Overnight</option>
</select>
```
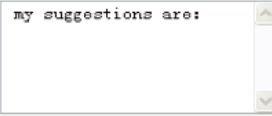
- **The buttons in the same group have the same name**

- **If the user selects excellent, the server will receive: rating="excellent"**

- **Radio buttons let a user select ONE choice**

- **You can create a list using the `<select>…</select>` tag, giving the items in the list by using the**

- **If the user selects "Standard", the server side will receive: shipping=1**

- **`<option value="1" selected="selected">Standard</option>`**

Text Area

Enter your suggestions here:

my suggestions are:

```
<label > Enter a suggestion </label>
<textarea name="suggestions" rows="5" cols="25"> my suggestions are:</textarea>
```

- Used for multiple line inputs

- The attributes `rows` and `cols` specify the size

HTML5 Form Validation

- HTML5 allows us to instruct the browser to validate the form before it's sent to the server

Session Practical 2 - Form Validation (15 mins)

The goal of this practical is to appreciate just how easy it is to use HTML5 for form validation.

1. Create a new file called `session_7_form_validation.html`, and copy in a sample html form from [here](https://raw.githubusercontent.com/sirus21/Internet_technology/master/session7/session_practicals/session_7_form_validation.html)
   `https://raw.githubusercontent.com/sirus21/Internet_technology/master/session7/session_practicals/session_7_form_validation.html`.

2. Press the submit button, you should observe that there is currently no validation.

3. To make a input required, you just need to add `required` to the element. Add `required` to `Name` and `Email`, press **submit**, on the empty form. You should now see a error prompt.

   ```
   <input type="text" name="Name" required>
   ```

4. Let's take this a little further, at the moment a user can still enter a invalid email. HTML5 allows us to be more specific with the `type` element and solve this issue. Try changing `type="text"` to `type="email"`. Test your form to check the validation works.

Some other common HTML5 input types are:

- color
- date
- datetime
- datetime-local
- month
- search
- tel
- time
- week

Weekly Task - HTML5 Forms and validation

The purpose of this task is to increase your understanding of HTML5 forms and validation. Today we'll be focusing on HTML only, there will be no need to style the forms using CSS.

1 Setup

Create the file `form_task.html` and place it in your `internet_tech/session_7` folder. [Copy the starting HTML source code from here](#).

2 Populate the form with the correct inputs

Using the comments as guidance complete the form. It should resemble the below:

Student Survey

Name*: [_____]
Email*: [_____]
Number*: [_____]
What year are you in*: Select ▼
Which electronic devices do you use?
☐ Computer
☐ Ebook reader
☐ Smart phone
How many hours a day do you spend using electronic devices?
○ 1 hours
○ 2 Hours
○ 3 Hours +
Any other comments

[_____]

[Submit]

3 No need to validate your HTML this week

As we're not focusing on CSS this week, we've had to use a few tags that will break validation. Therefore, you're not required to validate your work.

4 Update your learning log