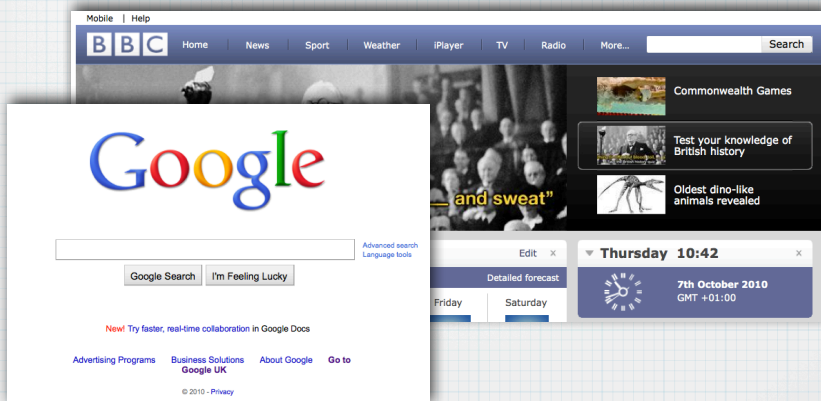


PHP

forms

## Web forms

Most common way of getting data from user



## Web forms

### TV feedback

This form will allow you to provide feedback for your favourite soap operas

#### Provide your feedback

How many times do you watch soaps a week:

Thank you for your feedback

Demo

## Web forms

### TV feedback

This form will allow you to provide feedback for your favourite soap operas

#### Provide your feedback

How many times do you watch soaps a week:

Thank you for your feedback

Input field - submit button

Input field - text

## Web forms

No PHP in here - can be saved as an **.html** file

```
<body>
<h1>TV feedback</h1>
<p>This form will allow you to provide feedback for your favourite soap operas</p>
<h2>Provide your feedback</h2>

<form name="soapform" action="response1.php" method="get">

<p>How many times do you watch soaps a week:<input type="text" name="timesaweek" /></p>
<p>Thank you for your feedback <input type="submit" name="continue" /></p>

</form>

</body>
```

## Web forms

form must be indicated by **form** element

```
<body>
<h1>TV feedback</h1>
<p>This form will allow you to provide feedback for your favourite soap operas</p>
<h2>Provide your feedback</h2>

<form name="soapform" action="response1.php" method="get">

</form>

</body>
```

Needs at least three attributes **name**, **action** and **method**

## Web forms

form must be indicated by **form** element

```
<body>
<h1>TV feedback</h1>
<p>This form will allow you to provide feedback for your favourite soap operas</p>
<h2>Provide your feedback</h2>
```

```
<form name="soapform" action="response1.php" method="get">
```

Use **name** to give each form  
on the page a unique name

```
</form>
```

```
</body>
```

## Web forms

form must be indicated by **form** element

```
<body>
<h1>TV feedback</h1>
<p>This form will allow you to provide feedback for your favourite soap operas</p>
<h2>Provide your feedback</h2>
```

```
<form name="soapform" action="response1.php" method="get">
```

Use **action** to indicate the  
'next' page where the form  
will be processed

```
</form>
```

```
</body>
```

## Web forms

form must be indicated by **form** element

```
<body>
<h1>TV feedback</h1>
<p>This form will allow you to provide feedback for your favourite soap operas</p>
<h2>Provide your feedback</h2>
```

```
<form name="soapform" action="response1.php" method="get">
```

Use **method** to indicate the  
way that data will be  
transferred

```
</form>
```

```
</body>
```

## Web forms

### Two ways of passing information between pages

get	form information is passed through the URL
post	form information is embedded in the HTTP stream

..more on this in a minute

## form elements

### TV feedback

This form will allow you to provide feedback for your favourite soap operas

#### Provide your feedback

How many times do you watch soaps a week:

Thank you for your feedback

```
<input type="text" name="timesaweek" />  
<input type="submit" name="continue" />
```

Give every form element a unique **name** - these are used to pass the data values and in the processing page

## method="get"

The form values are passed in the URL using **name = value** pairs

```
responseget1.php?timesaweek=4&continue=Submit+Query
```

**method="get"**

The form values are passed in the URL using **name = value** pairs

```
responseget1.php?timesaweek=4&continue=Submit+Query
```

Special encoding has to occur for spaces, =, & etc.

Called **URL encoding** and done automatically by the browser - note that **name** comes from the form

**method="get"**

The form values are passed in the URL using **name = value** pairs

```
responseget1.php?timesaweek=4&continue=Submit+Query
```

The continue button is also a form element - here with the default value

**\$\_GET['...']**

php creates a variable for each form element passed through using this notation

```
$_GET['form_element_name']
```

Best to create a variable to get values out at the top of the page

```
$timesaweek = $_GET['timesaweek'];
```

## Web forms

### responseget1.php page

```
<?php
$timesaweek = $_GET['timesaweek'];
?>

<html>
<head>
</head>
<body>
<h1>Responses</h1>
<p>Here are the results</p>
<p>
<?php
print "<p>The amount watched was ".$timesaweek."</p>";
?>
</p>
</body>
</html>
```

## Web forms - POST example

### TV feedback

This form will allow you to provide feedback for your favourite soap operas

#### Provide your feedback

How many times do you watch soaps a week:

Thank you for your feedback

Looks the same in the browser

## method="post"

```
<body>
<h1>TV feedback</h1>
<p>This form will allow you to provide feedback for your favourite soap operas</p>
<h2>Provide your feedback</h2>

<form name="soapform" action="response2.php" method="post">
<p>How many times do you watch soaps a week:<input type="text" name="timesaweek" /></p>
<p>Thank you for your feedback <input type="submit" name="continue" /></p>
</form>
</body>
```

Small change here

But how is the data passed to the next page when using POST?

## Web forms

### Two ways of passing information between pages

get	form information is passed through the URL
post	form information is embedded in the HTTP stream

## Web forms

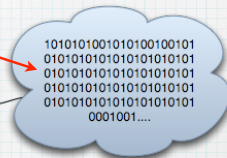
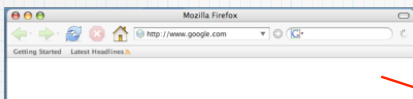
### Two ways of passing information between pages

get	form information is passed through the URL
post	form information is embedded in the HTTP stream

## How POST works ...

User clicks on form

Page request AND data is sent to server using HTTP



```
POST /responses.php HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0
Content-Length: 27
Content-Type: application/x-www-form-urlencoded

timesaweek=4&continue=Submit+Query
```

**`$_POST['...']`**

php creates a variable for each form element passed through using this notation

```
$_POST['form_element_name']
```

Best to create a variable to get values out at the top of the page

```
$timesaweek = $_POST['timesaweek'];
```

## Web forms

responsepost1.php page

```
<?php
$timesaweek = $_POST['timesaweek'];
?>

<html>
<head>
</head>
<body>
<h1>Responses</h1>
<p>Here are the results</p>
<p>
<?php
print "<p>The amount watched was ".$timesaweek."</p>";
?>
</p>
</body>
</html>
```

Uses the `_POST` array

## Method choice - GET vs POST

GET - visible, bookmarkable

POST - not seen on screen, not bookmarkable