Todd Smith
tbsmith@fullsail.com

Welcome to SSL Day 3!

Forms and Server-Side Validation

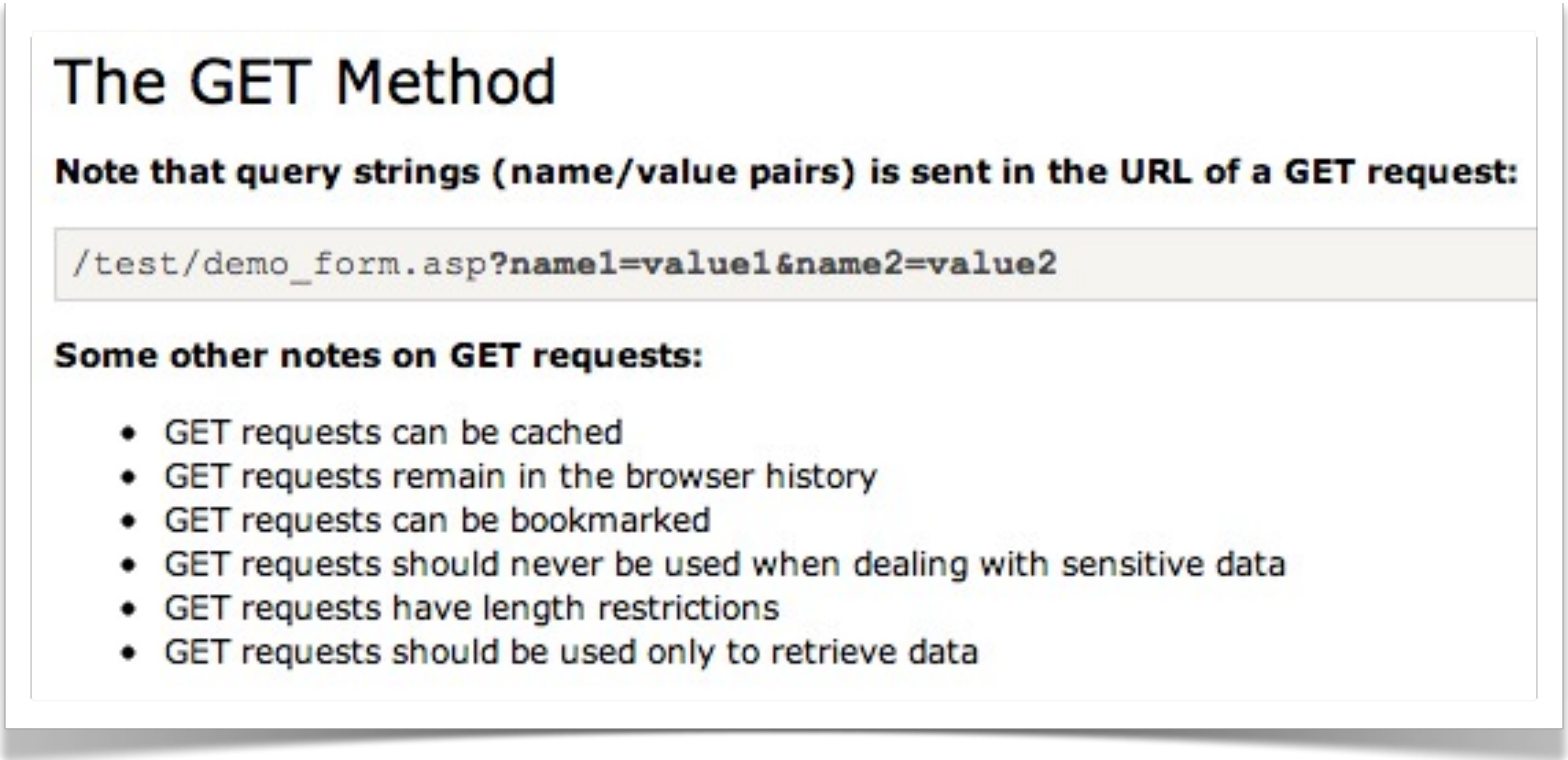
Day 3

Overview

- Questions about Day2 or Lab2?
- Forms
- Server-Side Validation

Server-Side Languages

Sending information to the server-side script



The GET Method

Note that query strings (name/value pairs) is sent in the URL of a GET request:

```
/test/demo_form.asp?name1=value1&name2=value2
```

Some other notes on GET requests:

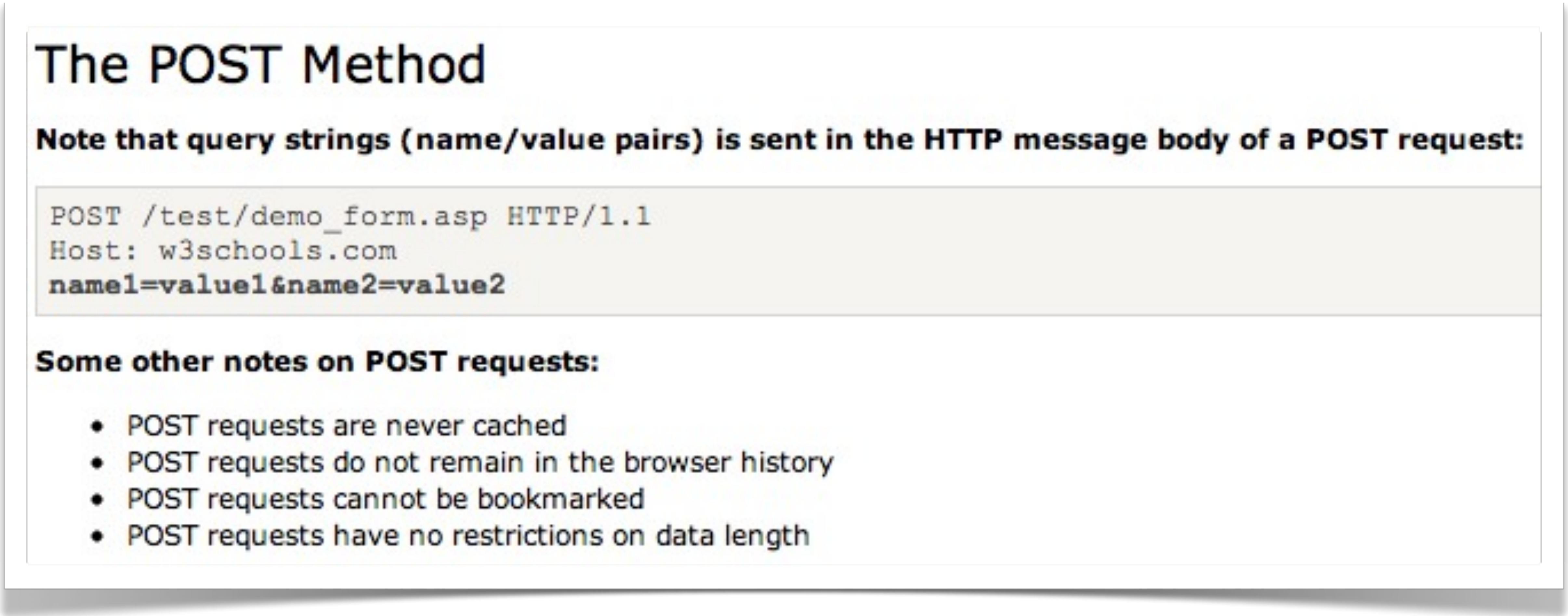
- GET requests can be cached
- GET requests remain in the browser history
- GET requests can be bookmarked
- GET requests should never be used when dealing with sensitive data
- GET requests have length restrictions
- GET requests should be used only to retrieve data

http://www.w3schools.com/tags/ref_httpmethods.asp

Day 3

Server-Side Languages

Sending information to the server-side script



The POST Method

Note that query strings (name/value pairs) is sent in the HTTP message body of a POST request:

```
POST /test/demo_form.asp HTTP/1.1
Host: w3schools.com
name1=value1&name2=value2
```

Some other notes on POST requests:

- POST requests are never cached
- POST requests do not remain in the browser history
- POST requests cannot be bookmarked
- POST requests have no restrictions on data length

http://www.w3schools.com/tags/ref_httpmethods.asp

Day 3

Responding to the URL in PHP

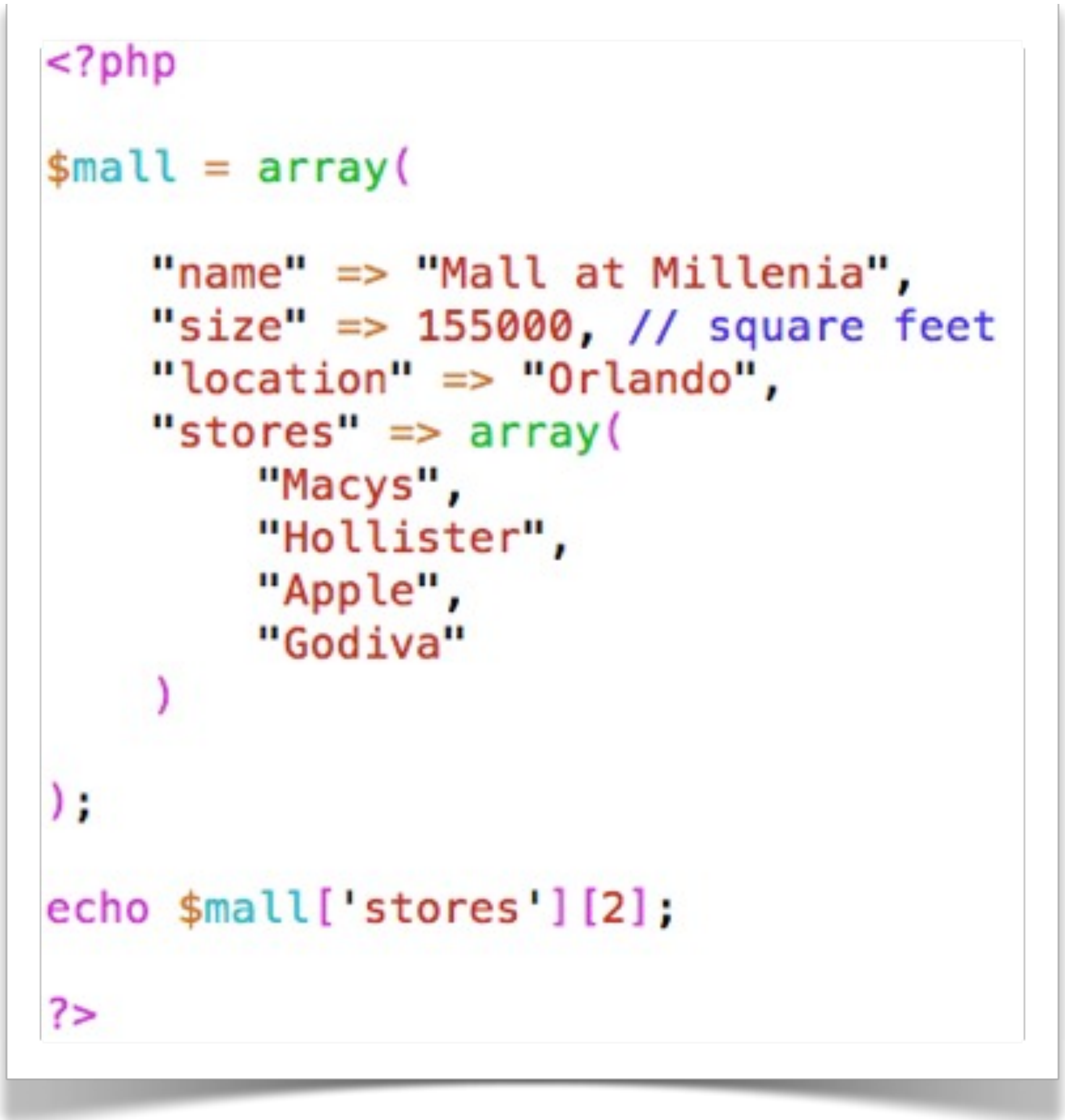
PHP scripts will make information from the URL available in the $_GET associative array.

PHP scripts will make information from posted forms available in the $_POST associative array.

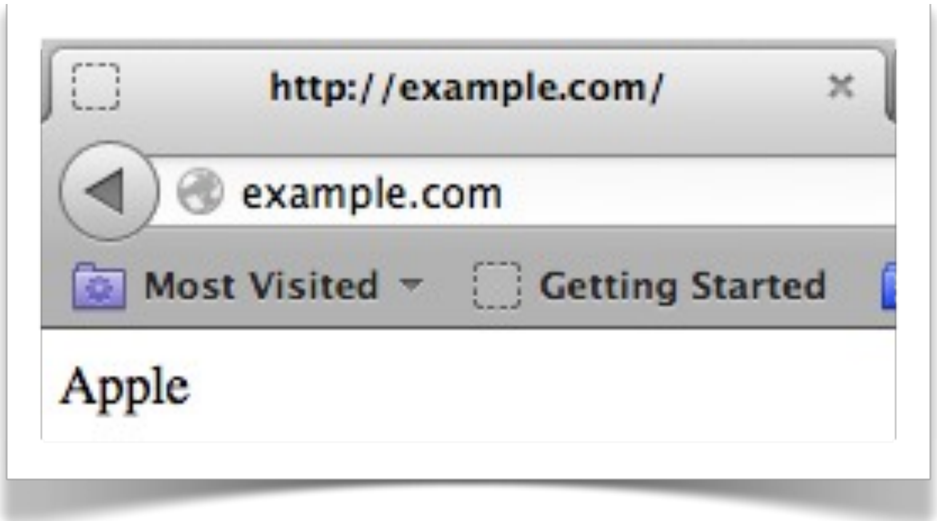
Server-Side Languages

PHP Associative Arrays

Associative arrays can be built with the array() function and the => symbols, and can be accessed with square brackets.



```
<?php

$mall = array(

    "name" => "Mall at Millenia",
    "size" => 155000, // square feet
    "location" => "Orlando",
    "stores" => array(
        "Macys",
        "Hollister",
        "Apple",
        "Godiva"
    )

);

echo $mall['stores'][2];

?>
```



Apple

Day 3

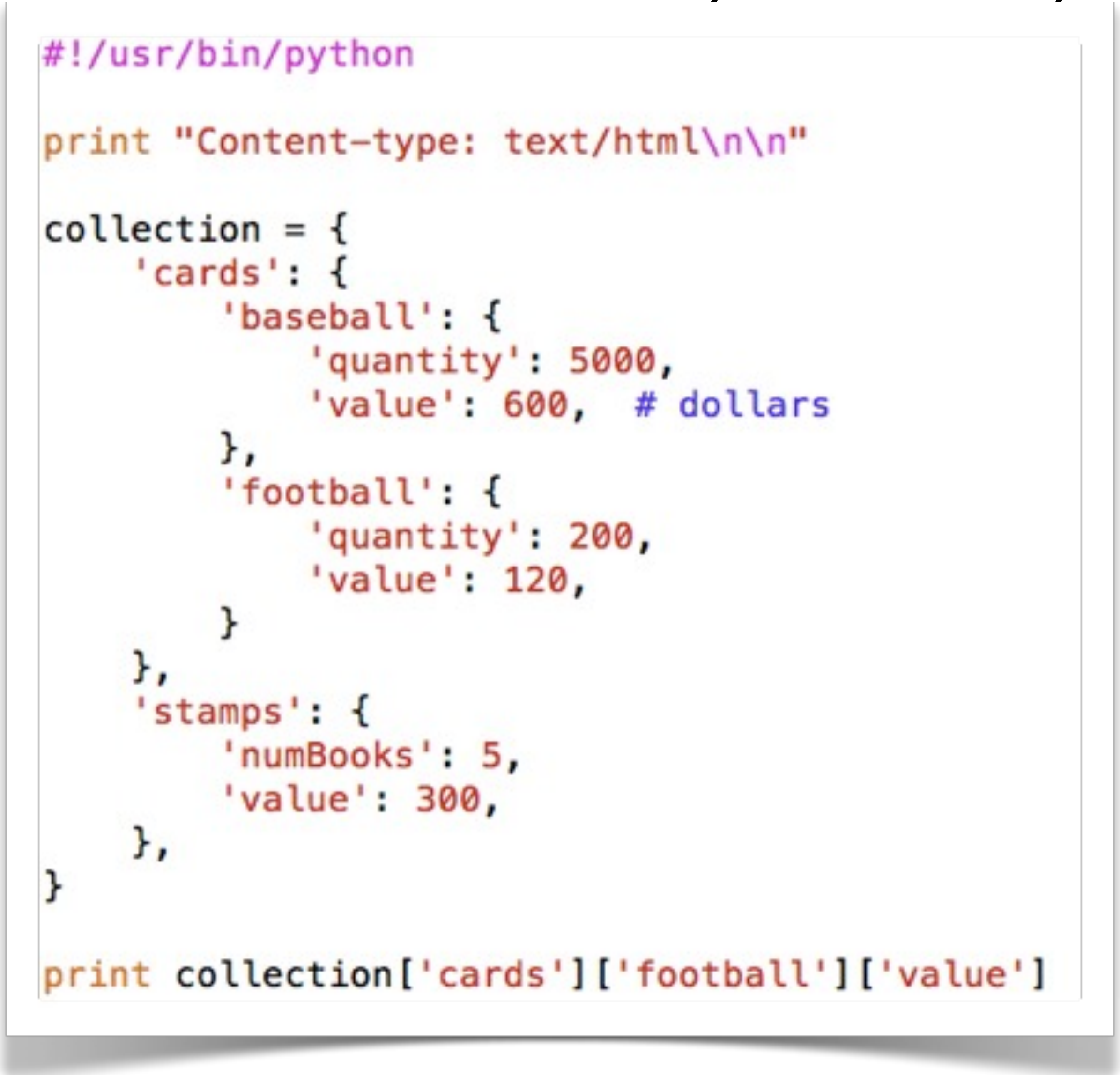
Responding to the URI in Python

The script must import the 'cgi' package.

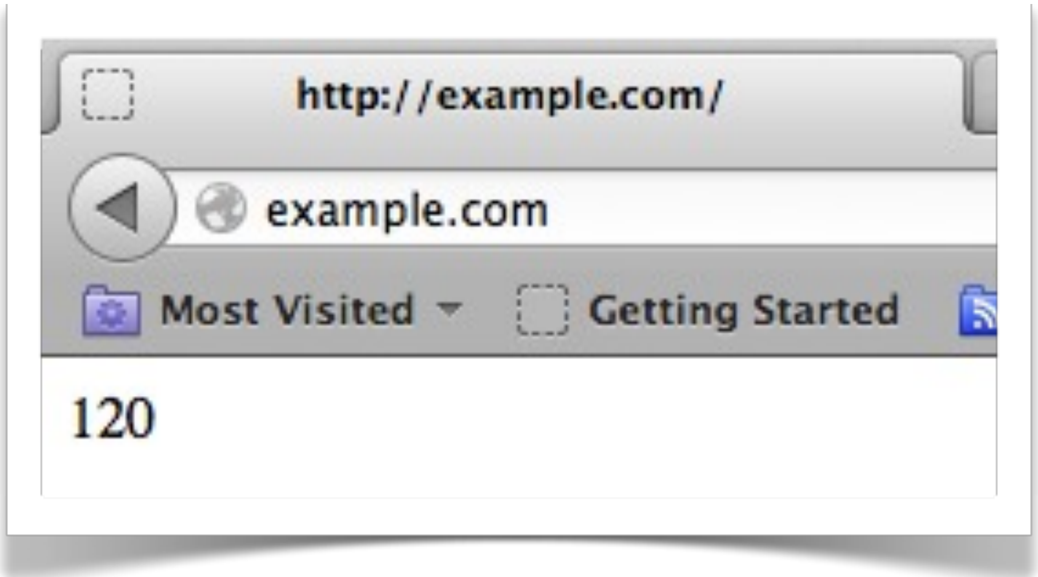
Python scripts will make information from the URL and from posted forms available in the dictionary returned by the cgi.FieldStorage() function.

Python Dictionaries

Dictionaries can be built with curly braces and colons.
The syntax is exactly the same as JSON.

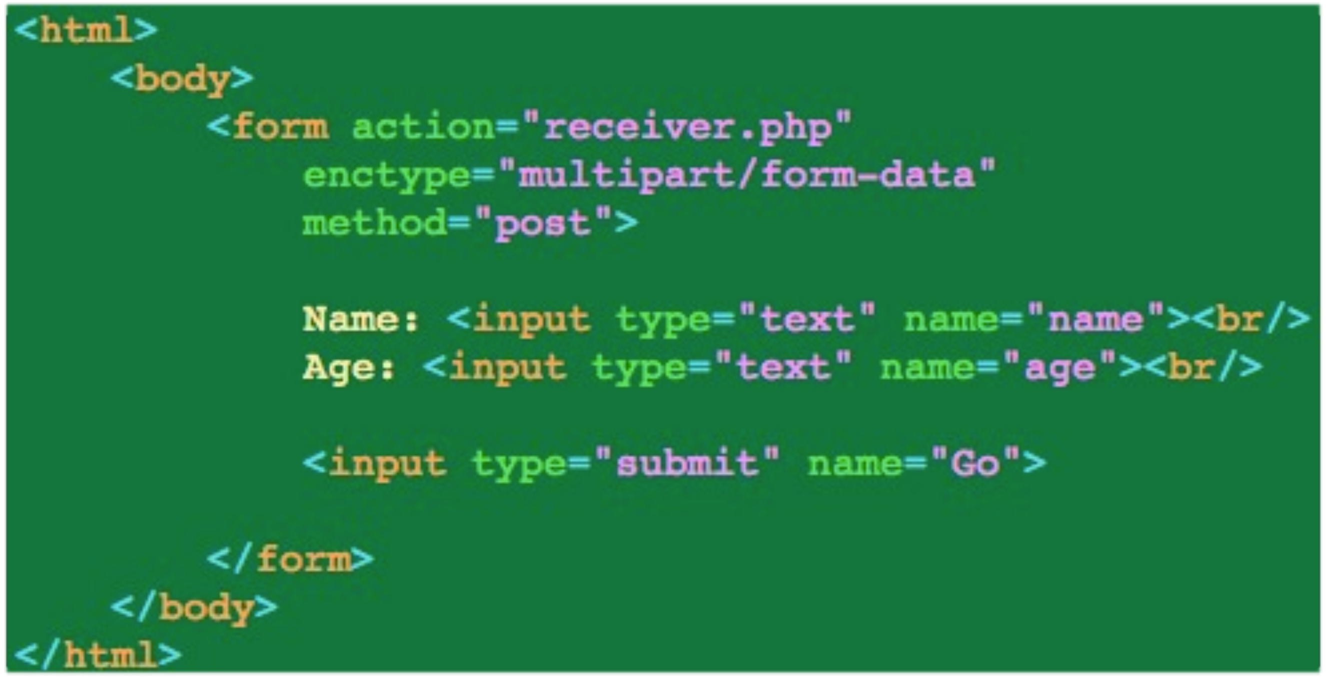


```
#!/usr/bin/python

print "Content-type: text/html\n\n"

collection = {
    'cards': {
        'baseball': {
            'quantity': 5000,
            'value': 600,  # dollars
        },
        'football': {
            'quantity': 200,
            'value': 120,
        }
    },
    'stamps': {
        'numBooks': 5,
        'value': 300,
    },
}

print collection['cards']['football']['value']
```

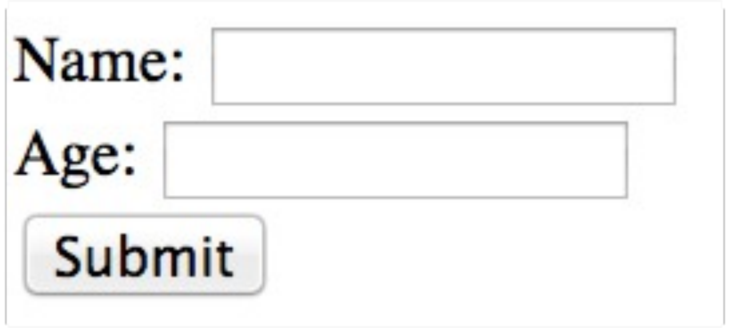


```
120
```

Sending HTTP requests: The form



```
<html>
    <body>
        <form action="receiver.php"
            enctype="multipart/form-data"
            method="post">

            Name: <input type="text" name="name"><br/>
            Age: <input type="text" name="age"><br/>

            <input type="submit" name="Go">

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



Name:

Age:

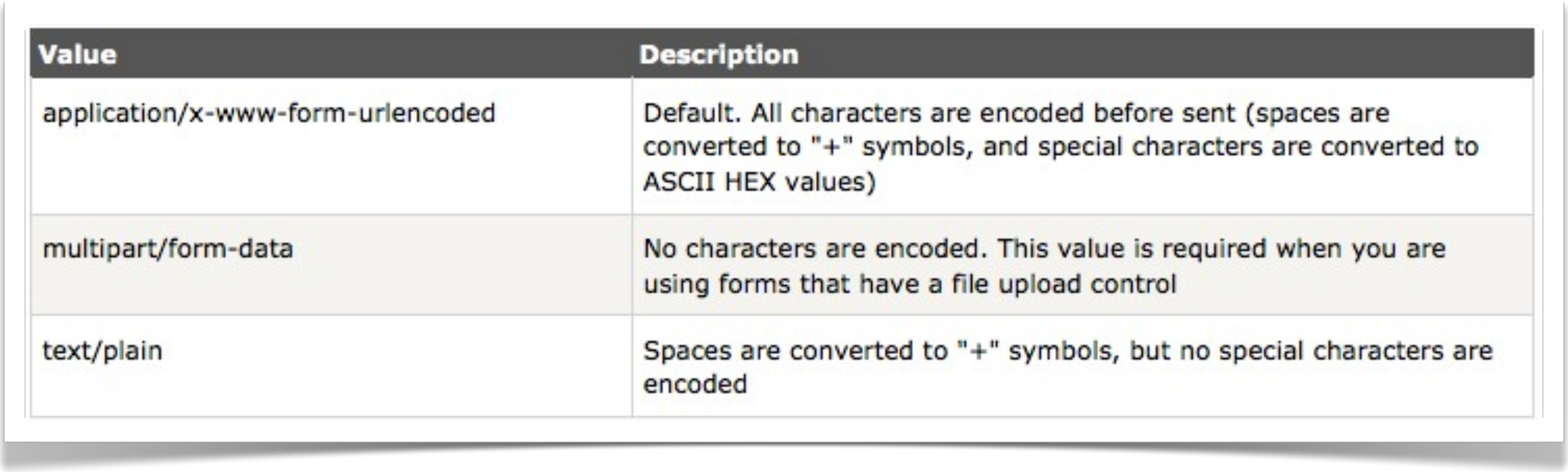
Server-Side Languages

Read about the different kinds of form input here:

http://www.w3schools.com/html/html_forms.asp

Day 3

Form Enctype



Value	Description
application/x-www-form-urlencoded	Default. All characters are encoded before sent (spaces are converted to "+" symbols, and special characters are converted to ASCII HEX values)
multipart/form-data	No characters are encoded. This value is required when you are using forms that have a file upload control
text/plain	Spaces are converted to "+" symbols, but no special characters are encoded

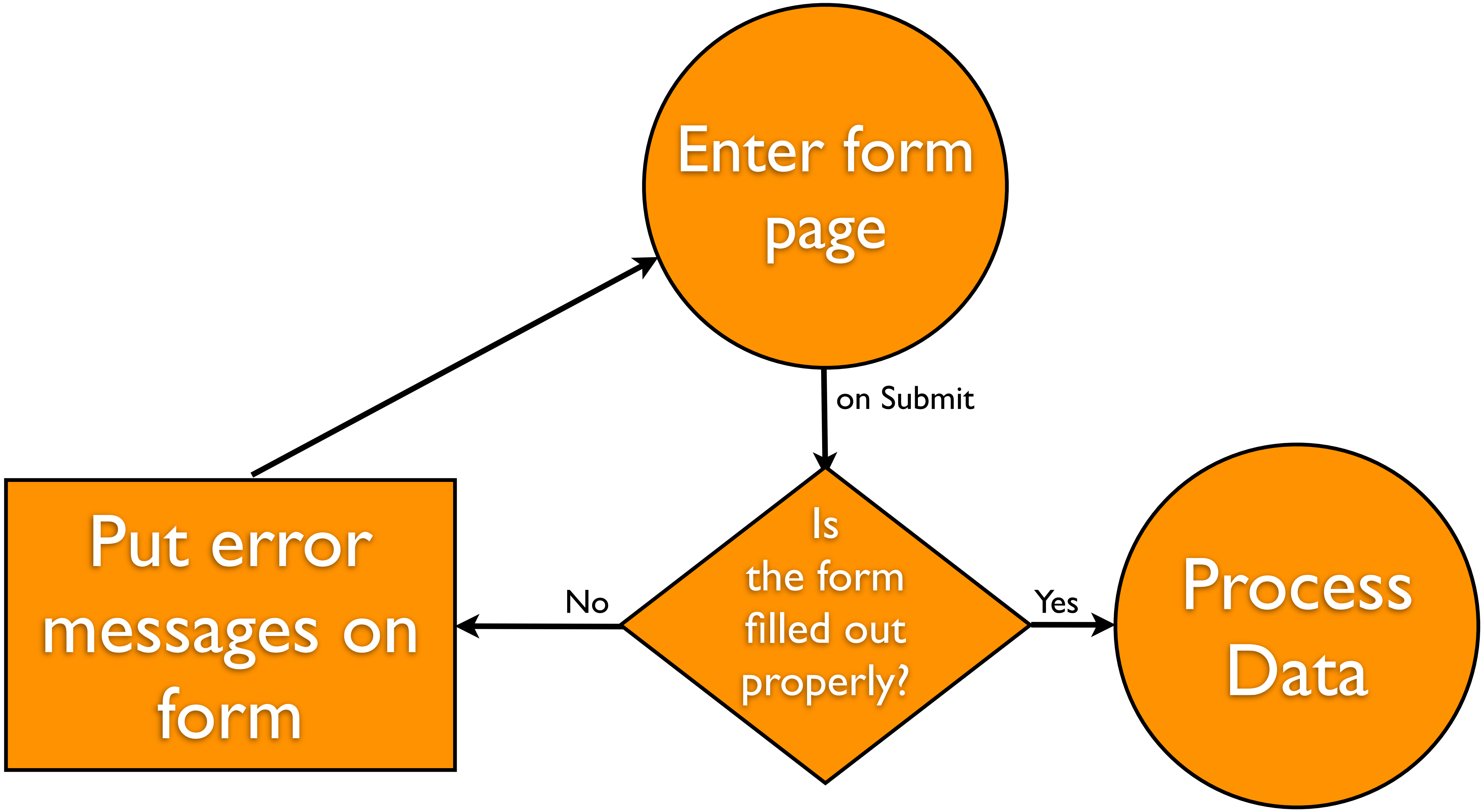
http://www.w3schools.com/tags/att_form_enctype.asp

In-class Assessment: Forms

- Build a form that sends name, email address, and phone number to a script which then displays the information on the screen.
- Add two fields with radio buttons and two fields with checkboxes, and show their values with your script.

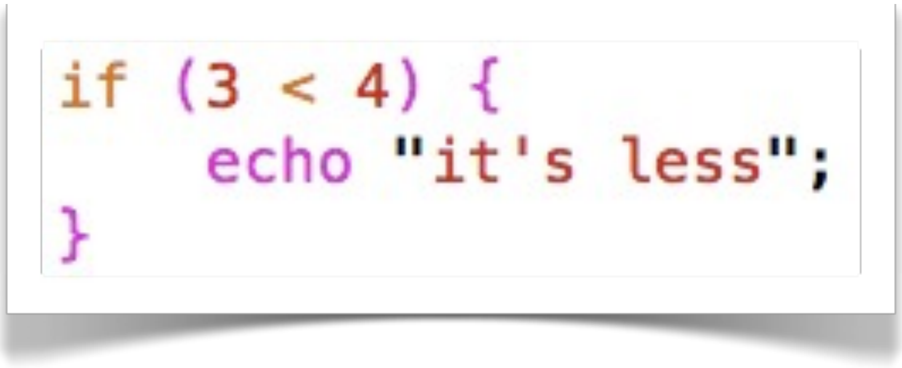
Day 3

Server-Side Validation



Day 3

PHP Conditionals

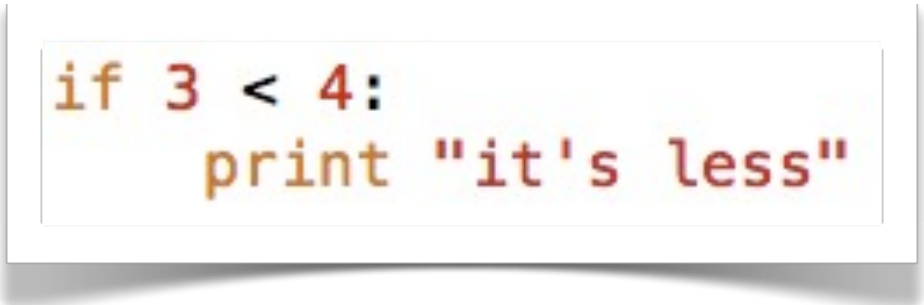


```
if (3 < 4) {
     echo "it's less";
}
```

Read More:

http://php.net/manual/en/language.operators.comparison.php

Python Conditionals



```
if 3 < 4:
    print "it's less"
```

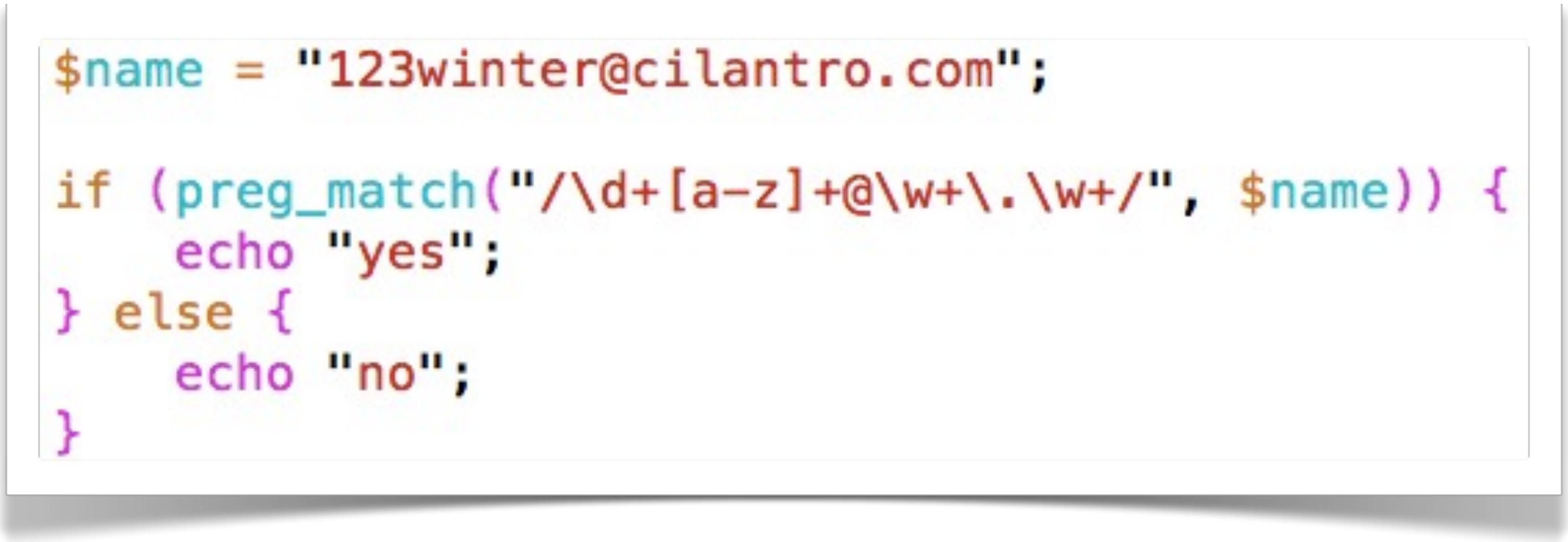
Read More:

http://docs.python.org/2/reference/expressions.html#not-in

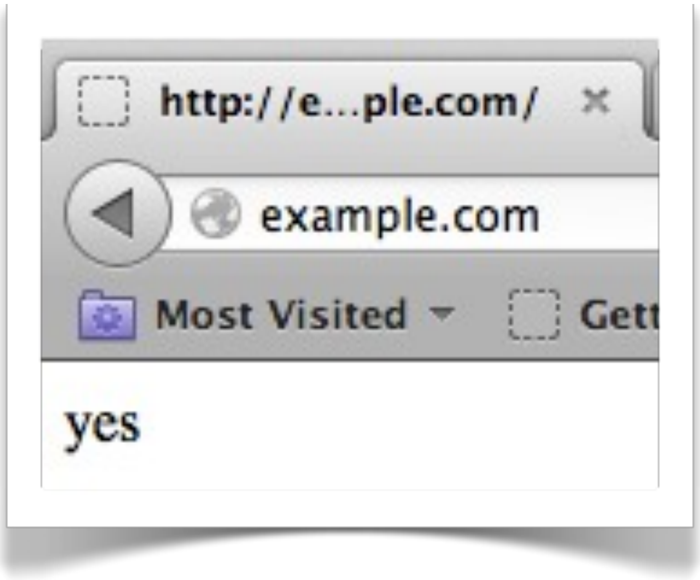
Regular Expressions Quick Guide

\n	Matches a newline
\w	Alphanumeric character
\d	Digit
^	Beginning of line
$	End of line
[a-z]	Lowercase letter
.	Any single character
\.	A literal dot / period
+	Matches preceding character one or more times

Regular Expressions in PHP

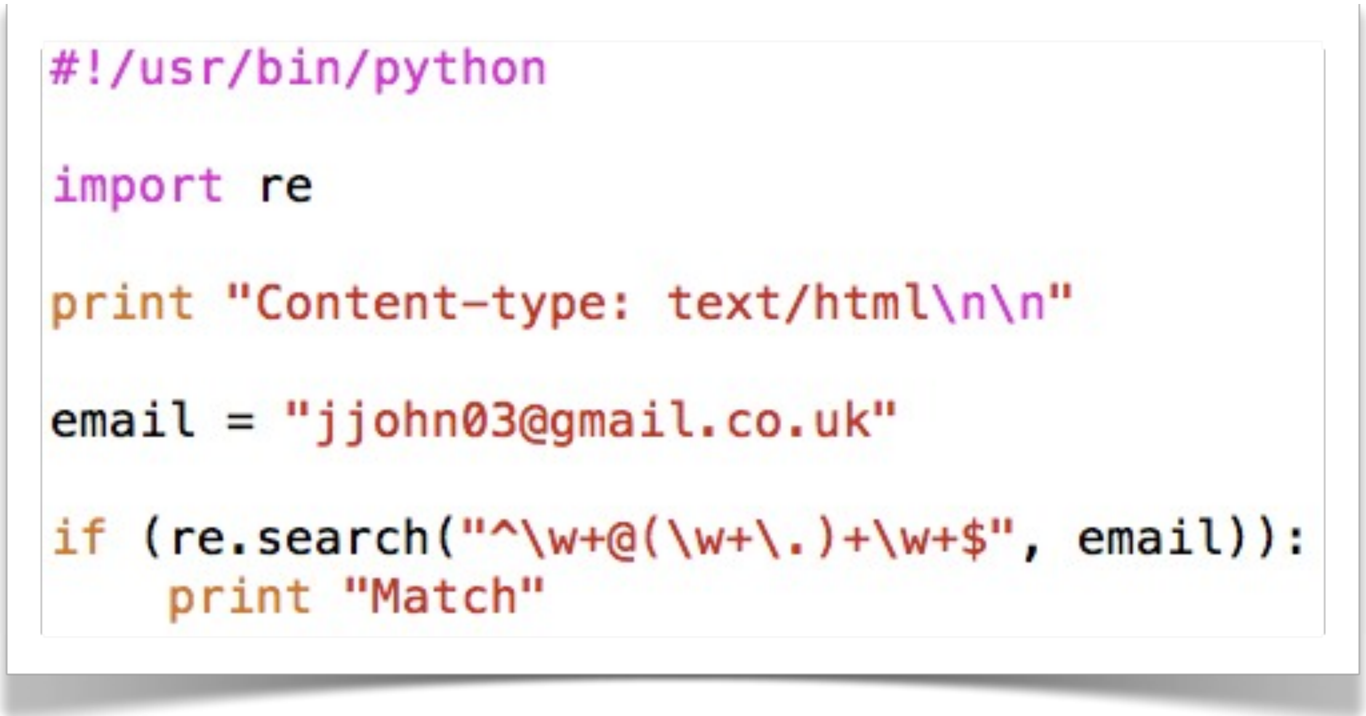


```
$name = "123winter@cilantro.com";

if (preg_match("/\d+[a-z]+@\w+\.\w+/", $name)) {
    echo "yes";
} else {
    echo "no";
}
```

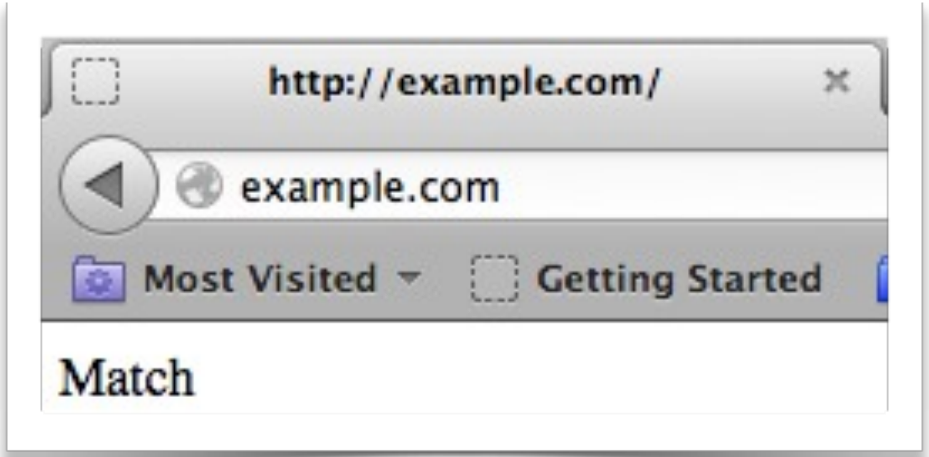


```
yes
```

Regular Expressions in Python



```
#!/usr/bin/python

import re

print "Content-type: text/html\n\n"

email = "jjohn03@gmail.co.uk"

if (re.search("^\w+@(\w+\.)+\w+$", email)):
    print "Match"
```



```
Match
```

In-class Assessment: Regular Expressions

Make your form receiver scripts from the previous assessment validate input.

Validate email address, name, and phone number.

Print an error if the input is wrong, and show the form again.

Day 3

Server-Side Languages

Lab 3

Incorporate a registration page with server-side validation into both of your websites.

Day 3