CENTER FOR TEXTUAL STUDIES AND
DIGITAL HUMANITIES

DIGH 401 - Introduction to Computing

Fall Semester 2013

Week 9

Today's Class

- XAMPP and editors demo
- manipulating data

Manipulating Data - Mathematics

A few PHP Mathematics functions - rocket thrust exercise!

- abs() finds the absolute value of a number

- round() rounds a number to the nearest integer

- ceil() rounds a number upwards to the nearest integer

- floor() rounds a number downwards to the nearest integer

- min() returns the smaller of the two arguments passed

- max () returns the larger of the two arguments passed

- rand(min, max) simply returns a random integer
    - Example

```
<?php
echo '<p>'.rand().'</p>';
echo '<p>'.rand(1, 49).'</p>';
?>
```

Further information: PHP Mathematics functions

Manipulating Data - Mathematics

```
<?php
$total = 0;
$booster1 = 50;
$booster2 = 100;
$booster3 = rand();

echo '<p>Original Total = '.$total.'</p>';

function rocketThrust()
{
global $total, $booster1, $booster2, $booster3;
        $total = $booster1 + $booster2 + $booster3;
}

rocketThrust();
echo '<p>Rocket thrust is currently stable at '.$total.' newtons</p>';
?>
```

- rocketThrust Random Example

Manipulating Data - Working with Strings

- many programming languages include built-in string functions such as
    - counting the number of characters in a string
    - removing characters from a string
    - comparing strings
    - splitting strings...

Common functions can include options such as

length(x) = counts the number of characters in a string (x) including spaces
trim(x,y) = removes characters from a string
index(x,y) = returns the positon of a string within another string
compare(x,y) = compares two strings to see where they differ
replace(x,y,z) = replace one string with another string within a main string

Manipulating Data - Working with Strings

PHP - some common functions

- strlen() finds the character length of a string
    strlen("DIGH 401");

- strpos() searches for a given character, word, phrase etc within a string
    strpos("Hello World, again!", "World");

- str_replace() replaces specified characters in a string with new characters, case-sensitive
    - str_replace('World', 'Jim', 'Hello World, again!');

Example 1

Example 2

Manipulating Data - Working with Strings

Regular expressions

- before you can manipulate a string you need to find it, eg: a substring
- some languages include string searching functions
- these functions tend to be fairly limited and normally require exact matches
- regular expression is a series of symbols that tells the computer how to find a given pattern in a string
- regular expressions offer a more powerful way of searching strings

Manipulating Data - Working with Strings

Regular expressions - Single character (.) wildcard

- simplest way to search for a pattern is by using a single character
- single character wildcard (.)

b.t

- use the number of wildcards (.) to match the number of characters searched

b..t would return boat, boot, bolt...

- this expression will only match the exact number of wildcards

Manipulating Data - Working with Strings

Regular expressions - Specific character matching

- search for any specific character using [ ] symbols
- enclose the characters to search for within the brackets
- add multiple characters in the brackets to search for different alternatives

b[aou]t

- this expression will search for words that contain either a, o, or u between b and t
- you can also use the not ^ character to specify characters to ignore

b[^ao]t

- ^ can also be used in regular expression to define an anchor normally at the start of a target string

q[^x]

Manipulating Data - Working with Strings

Regular expressions - Pattern matching with ranges

- match characters within a range of specified characters
- search for just letters, eg: bu[a-z]
- restrict range as much as is required, eg: bu[r-u]
- you can also check numeric ranges, eg 15[0-9]
- numbers can also be restricted to a shorter range, eg: 15[2-5]
- combine ranges to check wider selection of characters and values

[0-9a-fA-F]

[0-9a-fxA-FX]

Manipulating Data - Working with Strings

Regular expressions - Non-Printable characters

- use special characters to add non-printable characters to an expression

\t = tab
\r = carriage return
\n = line feed or new line

\xFF eg: \xA9 uses the hexadecimal index to match the copyright symbol in the Latin-1 character set

\uFFFF eg: \u20AC uses Unicode to match the Euro currency symbol

- all non-printable characters can be used in regular expressions

Manipulating Data - Working with Strings

Regular expressions - Anchors

- do not directly match any characters
- they match a position in the given string

^ matches at the start of the string
$ matches at the end of the string
\b matches at a word boundary

\w word characters

\W
\B

Manipulating Data - Working with Strings

Regular expressions - Alternation

- character used is a pipe |
- similar to OR operator in programming languages

    book | volume

- finds either everything to left or everything to the right

    (book) | (volume)

Manipulating Data - Working with Strings

Regular expressions - Repetition

- three basic ways to enable repetition in regular expressions
    - ?, * and + operators

- ? makes the preceding token in the regular expression optional

    colou?r

- * attempts to match the preceding token zero or more times

    <[A-Za-z] [A-Za-z0-9]*>

- + attempts to match the preceding token once or more

    a+b

Manipulating Data - Working with Strings

Regular expressions - Combining expressions

- combine multiple expressions to search for a variety of different string patterns

b[aei].[0-9]

- this example will search through a string looking for matches based on each of the four character requirements

Try www.regular-expressions.info for further info, tutorials, examples etc...

Javascript examples available at the W3 Schools

Manipulating Data - Comparison Operators

- comparison of two data values to see which is bigger than the other
- comparison operator returns either true or false
- often used in conditional expressions

==	equal to	x == y
===	identical and of same type	5 === 5
<> or !=	not equal	x != y
!==	not identical	x !== y
<	less than	x < y
<=	less than or equal to	x <= y
>	greater than	x > y
>=	greater than or equal to	x >= y

Manipulating Data - Comparison Operators

String comparison

- comparing strings using this method can be slightly confusing
- computers use a number to represent a letter
- ASCII table contains numbers for characters

A > a

- compare multiple characters in a string

aA > aa

aA > a

Aa > a

- as soon as the computer finds values that differ, it will cease the comparison and return a boolean result

Manipulating Data - Using Boolean Operators

- comparison operators return Boolean values, either True or False
- as with numbers and strings, Boolean values can also be manipulated
- boolean operators also return either True or False results
    - Not, And, Or, Xor
- PHP known as 'Logical Operators'
    - And, Or, Xor, Not

    x and y
    x or y
    x xor y

Manipulating Data - Using Boolean Operators

Java

!	NOT operator
&	AND operator
|	OR operator
^	XOR operator
||	short-circuit OR operator
&&	short-circuit AND operator
==	EQUAL TO operator
!=	NOT EQUAL TO operator

12 < 6 && 18 > 1
12 < 6 || 18 > 1

Manipulating Data - Using Boolean Operators

NOT operator

- takes a boolean value and converts it to the opposite
- True to False, and vice-versa

Not(3 > 2)

Manipulating Data - Using Boolean Operators

AND operator

- takes two boolean values and converts them into a single boolean value
- if boolean values are true, the And operator will return true
- if a False boolean value exists, the And operator will always return false

True	True	
True	False	
False	True	
False	False	

(3 > 2) AND (3 >= 18)

Manipulating Data - Using Boolean Operators

OR operator

- takes two boolean values and converts them into a single boolean value
- both boolean values are false, it will return a false value
- otherwise it will return a true value

True	True	
True	False	
False	True	
False	False	

(3 > 2) OR (3 >= 18)

Manipulating Data - Using Boolean Operators

XOR operator

- XOR is an exclusive OR operator
- converts two boolean values into a single boolean value
- both values True or False, XOR returns a False boolean value
- if one value is True and the other is false, XOR returns a True boolean value

True	True	
True	False	
False	True	
False	False	

(3 > 2) XOR (3 >= 18)