PHP

Array Functions

Useful Array function

- PHP made a bunch of functions that will make arrays easier to use and more useful.

- There's so many. For a complete list:
  http://www.w3schools.com/PHP/php_ref_array.asp

array_diff

- Compares two arrays and returns the differences

```
$sarasCheckList = array('tent', 'sleeping bag', 'pillow', 'food', 'clothes');
$tammysCheckList = array('vaccuum', 'pillow', 'food', 'sleeping bag', 'rain coat');

//will tell us what tammy remembed and sara didn't
$saraForgot = array_diff($tammysCheckList, $sarasCheckList);
```

Contains Vaccuum and RainCoat

array_intersect()

- Returns all the items that are both in array 1 and array 2

$sarasCheckList = array('tent', 'sleeping bag', 'pillow', 'food', 'clothes');
$tammysCheckList = array('vaccuum', 'pillow', 'food', 'sleeping bag', 'rain coat');

//tells us what they both remembered
$weRemembered = array_intersect($tammysCheckList, $sarasCheckList);

Contains 'pillow', 'food', 'sleeping bag'

array_keys()

- Takes an array as a parameter and returns the keys:

```
$capitols['Washington'] = 'Olympia';

$capitols['Florida'] = 'Tallahassee';

$capitols['Texas'] = 'Austin';

$capitols['NY'] = 'Albany';


$states = array_keys($capitols);
```

Contains 'Washington', 'Florida', 'Texas', 'NY'

array_merge

- Combines two or more arrays into one
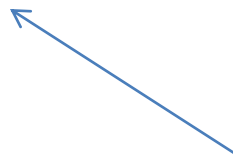- If two items share the same key. The item in the second array overwrites it and wins.

array_merge (cont...)

$capitols = array('Washington' => 'Seattle', 'Flordia' => 'Tallahouse', 'Texas' => 'Austin');

$moreCapitols = array('NY' => 'Albany','Washington' => 'Olympia', 'Flordia' => 'Tallahouse');

//merge the two arrays

$correctCapitols = array_merge($capitols, $moreCapitols);

```
Washington => Olympia
Flordia => Tallahouse
Texas => Austin
NY => Albany
```

array_pop

- Think of this as a stack of books.

- I just put a book on the top.
- In order to get at any books I pop the item off the top

array_pop (cont…)

- Removes the item off the end of the array
- Reducing the size of the array by 1
- And returns the item to you

array_pop (cont...)

```php
//create an array of books
$books[0] = 'Rudolf went to camp';
$books[1] = 'Skate Skate Skate';
$books[2] = 'Birthday Wishes';
$books[3] = 'Reading is fun';
$books[4] = 'Fun with Math 1, 2, 4';

$lastBook = array_pop($books);

echo("last book is $lastBook");    //displays fun with math
```

array_push

- Opposite of array_pop
- Takes in a item and pushes it on to the end of the array
- Increasing the size of the array by 1.

array_push

```php
//create an array of books
$books[0] = 'Rudolf went to camp';
$books[1] = 'Skate Skate Skate';
$books[2] = 'Birthday Wishes';
$books[3] = 'Reading is fun';


$item = 'Fun with Math';


array_push($books, $item);        //fun with math is now
                                              in the array
```

array_unique

Array_unique($myArray)

- Removes duplicates from the array

asort

- Sorts an array by it's values

  $colors = array('green', 'brown', 'red', 'purple')

  asort($colors);   //brown, green, purple, red

- arsort($myarray) will reverse sort it.

ksort

- sorts an array by it's keys

$capitols['Washington'] = 'Olympia';
$capitols['Flordia'] = 'Tallahousse';
$capitols['Texas'] = 'Austin';

ksort($capitols);   //flordia, texas, washington

- krsort reverse sorts by capitols

implode

- Opposite of the string function explode
- Converts an array into a string with the specified token to separate the elements

$colors = array('green', 'brown', 'red', 'purple');

$colorString = implode('! ', $colors);

echo($colorString);   //green! brown! red! purple!

In_array()

in_array($item, $myArray);

```
$people= array('Fred', 'Jon', 'Sara', 'Jen');

If(in_array('Sara', $people) {
    echo('you can proceed!');
}
```