Arrays in PHP

CIT 381
Starting PHP

- The php code starts with `<?php`
- and ends with `?>`
Variables

• All variables start with a dollar sign ...
• $state, $row, $name
• not typed - can be string or int or array
Arrays

- We will see a lot of arrays
- rows from result of SQL returned as array
- $row = mysqli_fetch_array($result, ...);
- $row = array(5, 4, 3, 2, 2, 5);
- the last example is a numeric array
Numeric Array

• $row = array(5, 4, 3, 2, 2, 5);
• $row[0] has value 5
• $row[1] has value 4
• indices are integers - hence numeric
• length of array is count($row), ...
• which returns 6
FOREACH Loop

• Loops through array

• `foreach($arrayname as $entryname)`

```php
foreach ($row as $value) {
    print $value;
    print "\n";
}
```
Another Array

- Arrays can be extended
- The following will have three elements
- in locations 0, 1, and 2

```php
$states = array();
$states[] = "FL";
$states[] = "IL";
$states[] = "LA";
```
More foreach

- Consider ...
- `foreach($states as $i => $state)`
- `$i` takes on all index values (0, 1, 2)
- `$state` each entry ("FL", "IL", "LA")
foreach($states as $i => $state) {
    print "$i\t";
    print $state;
    print "\n"; }

0 FL
1 IL
2 LA

note: \t is a tab
and \n a newline
Include html in output

- create a drop down box with all states

```php
print "<form action="yourFile.php">\n";
print "<select name = "state">\n";
foreach($states as $state) {
    $line = "<option value="\"$state\"">\n";
    $line = $line."$state</option>\n";
    print $line;
}
print "</select>";
```
<form action="yourFile.php">
<select name = "state">
<option value="FL">FL</option>
<option value="IL">IL</option>
<option value="LA">LA</option>
</select>
</form>
FOR Loops

- example: for($i=0; $i < 10; $i++)

```php
for($i=0; $i<count($states); $i++) {
    print "$i\t"
    print $states[$i];
    print "\n";
}
```
Associative Array

- These use names as an index
- $row = array('firstName' => 'Margaret', 'lastName' => 'Atwood');
- $row['firstName'] is 'Margaret'


```php
foreach($row as $fieldname => $name){
    print $fieldname;
    print "\t";
    print "$name";
    print "\n";
}
```

firstName Margaret
lastName Atwood
$header = "";
foreach($row as $fieldname => $name){
    $header = $header."\t".$fieldname;
}

$rowlist = "";
foreach($row as $fieldname => $name){
    $rowlist = $rowlist."\t".$name;
}

print $header;
print "\n";
print $rowlist;
Implosion

- format implode(glue-string, array)
- puts each entry of array into string ...
- with “glue” between each entry
- print implode("\n<p>\n", $states);