Web Programming
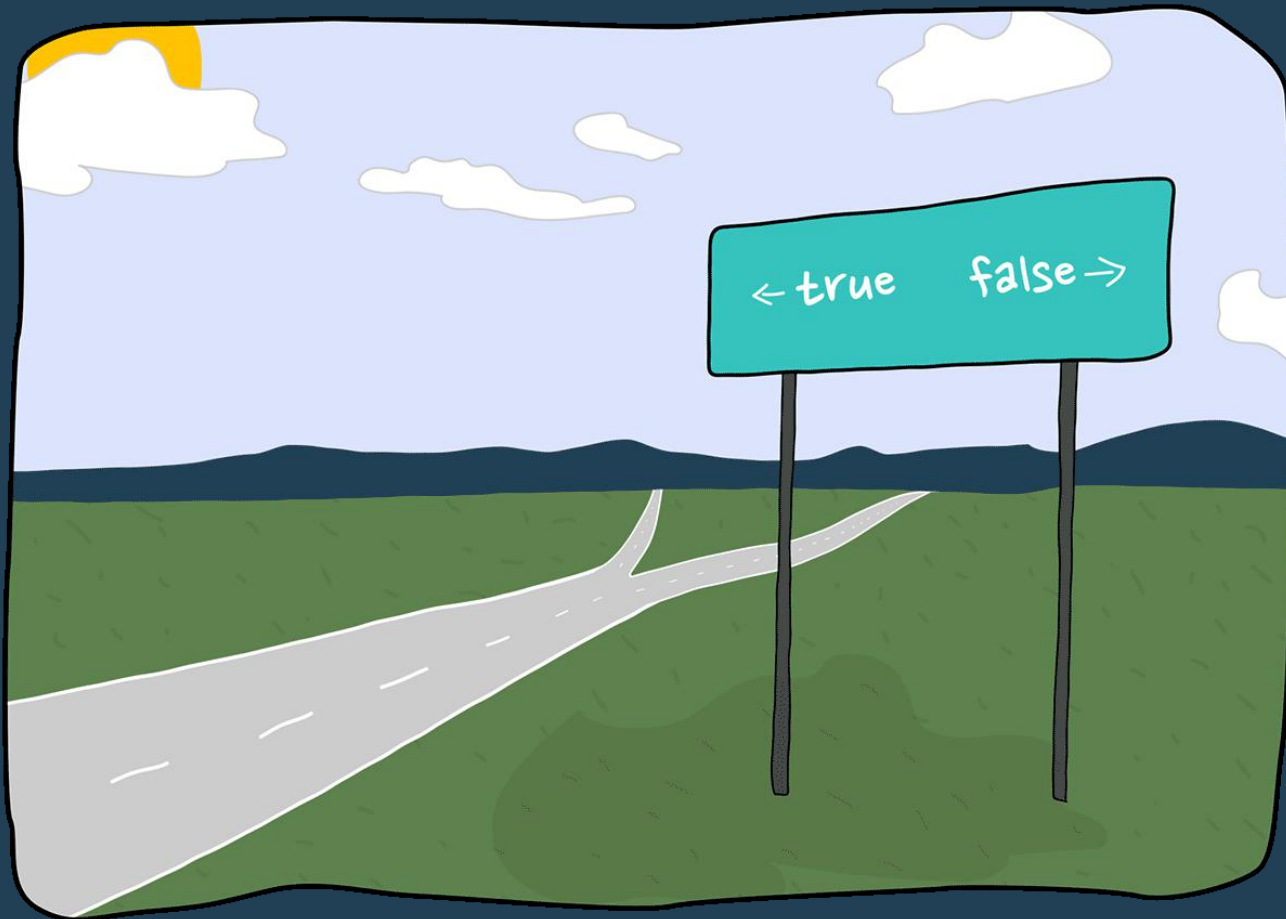## #6 PHP - Conditionals

Herman Kabetta, M.T.

Computers aren't intelligent,in order for them to make decisions
they have to be programmed with a set of rules to follow.

```
if (/*some condition*/) {
  // Do something...
}
```

```php
<?php
    $learner_one = ["is_correct"=>FALSE, "box"=>["shape"=>"none", "color"=>"none"]];
    $learner_two = ["is_correct"=>TRUE, "box"=>["shape"=>"none", "color"=>"none"]];

    function markAnswer ($is_answer_correct, &$box){
        if ($is_answer_correct){
            $box["shape"] = "checkmark";
            $box["color"] = "green";
        } else {
            $box["shape"] = "x";
            $box["color"] = "red";
        }
    }

    markAnswer($learner_one["is_correct"], $learner_one["box"]);
    markAnswer($learner_two["is_correct"], $learner_two["box"]);

    print_r($learner_one["box"]);
    print_r($learner_two["box"]);
?>
```

1. Write a function `chooseCheckoutLane()`. This function should take in a single number argument representing the number of items a customer has. If the customer has 12 items or fewer, the function should return "express lane". Otherwise, the function should return "regular lane".

2. In the U.S., citizens can vote if they are 18 years old or older. Write a function `canIVote()` that takes in a number representing an age, and return the string "yes" if they can vote, and the string "no" if they cannot.

3. Tested your functions to make sure they work with various inputs. Test each of your functions twice with FALSE and TRUE conditions.

1. Write a function `agreeOrDisagree()` that takes in two strings, and returns "You agree!" if the two strings are the same and "You disagree!" if the two strings are different.

2. Test your function! Invoke it once with a value that will result in the if block running and once with a value that won't.

- Make programs far more complex.
- We can write conditionals with multiple if statements using the elseif construction

```php
$grade = 88;
if ($grade < 60) {
    echo "You got an F";
} elseif ($grade < 70) {
    echo "You got a D";
} elseif ($grade < 80) {
    echo "You got a C";
} elseif ($grade < 90) {
    echo "You got a B";
} else {
    echo "You got an A";
}
```

Switch statement,
an alternate syntax
for conditional.

```php
if ($letter_grade === "A"){
    echo "Terrific";
} elseif ($letter_grade === "B") {
    echo "Good";
} elseif ($letter_grade === "C") {
    echo "Fair";
} elseif ($letter_grade === "D") {
    echo "Needs Improvement";
} elseif ($letter_grade === "F") {
    echo "See me!";
} else {
    echo "Invalid grade";
}
```

```php
switch ($letter_grade){
    case "A":
        echo "Terrific";
        break;
    case "B":
        echo "Good";
        break;
    case "C":
        echo "Fair";
        break;
    case "D":
        echo "Needs Improvement";
        break;
    case "F":
        echo "See me!";
        break;
    default:
        echo "Invalid grade";
}
```

Switch Statements: Fall through

- The keyword break tells the computer to break out of the switch statement, without it, it will fall through.

- But it can be useful when we want multiple cases to execute the same code.

```php
switch ($day_of_week) {
    case "Monday":
    case "Tuesday":
        echo "Just getting started!";
        break;
    case "Wednesday":
        echo "Humpday!";
        break;
    case "Thursday":
    case "Friday":
        echo "Almost the weekend!";
        break;
    case "Saturday":
    case "Sunday":
        echo "Enjoy!";
        break;
}
```

Write a function, `returnSeason()`. Your function should take in a string representing a month (e.g. "January"), and it should return which season that month falls in.

- "December", "January", and "February" are "winter" months.
- "March", "April", and "May" are "spring" months.
- "June", "July", and "August" are "summer" months.
- And "September", "October", and "November" are "fall" months.

Your function should use a switch statement and fall through to accomplish this task.

PHP offers a short-hand syntax to conditionally return a value

```php
$isClicked = FALSE;
if ( $isClicked ) {
    $link_color = "purple";
} else {
    $link_color = "blue";
}
```

```php
$isClicked = FALSE;
$link_color = $isClicked ? "purple" : "blue";
```

1. In a previous exercise, you wrote a function `chooseCheckoutLane()` using an if/else. Write a version of this function called `ternaryCheckout()`

2. In a previous exercise, you wrote a function `canIVote()` using an if/else. Write a version of this function called `ternaryVote()`

3. Test each of your functions twice—once with an input will fulfill the condition and once with an input that won't.

Logical Operators

Logical Operators

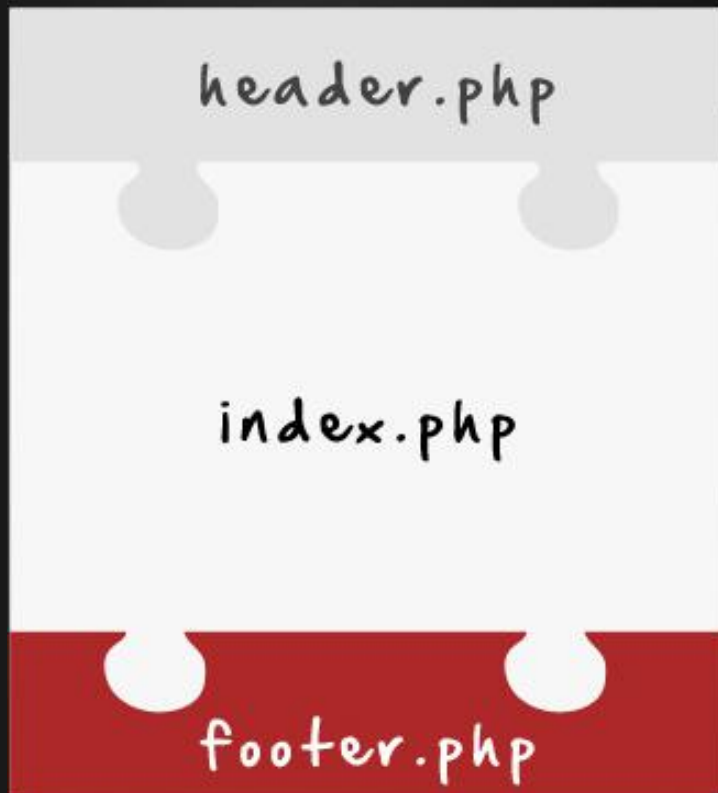| Example	Name	Result
$a and $b	And	TRUE if both $a and $b are TRUE.
$a or $b	Or	TRUE if either $a or $b is TRUE.
$a xor $b	Xor	TRUE if either $a or $b is TRUE, but not both.
! $a	Not	TRUE if $a is not TRUE.
$a && $b	And	TRUE if both $a and $b are TRUE.
$a || $b	Or	TRUE if either $a or $b is TRUE.

```php
<?php
    $banana_cream = ["whole milk", "egg yolks", "bananas", "heavy cream", "powdered sugar"];
    $experimental_pie = ["whole milk", "sugar", "bananas", "chicken", "salmon", "garlic"];

    // Write your code below:
    function eatPie($ingredients){
        if (in_array("chicken", $ingredients) xor in_array("bananas", $ingredients)){
            return "Delicious pie!";
        } else {
            return "Disgusting!";
        }
    }

    echo eatPie($banana_cream);
    echo "\n\n";
    echo eatPie($experimental_pie);
?>
```

Multi-File Programs: include


```php
// header.php
echo "This is Header";
```
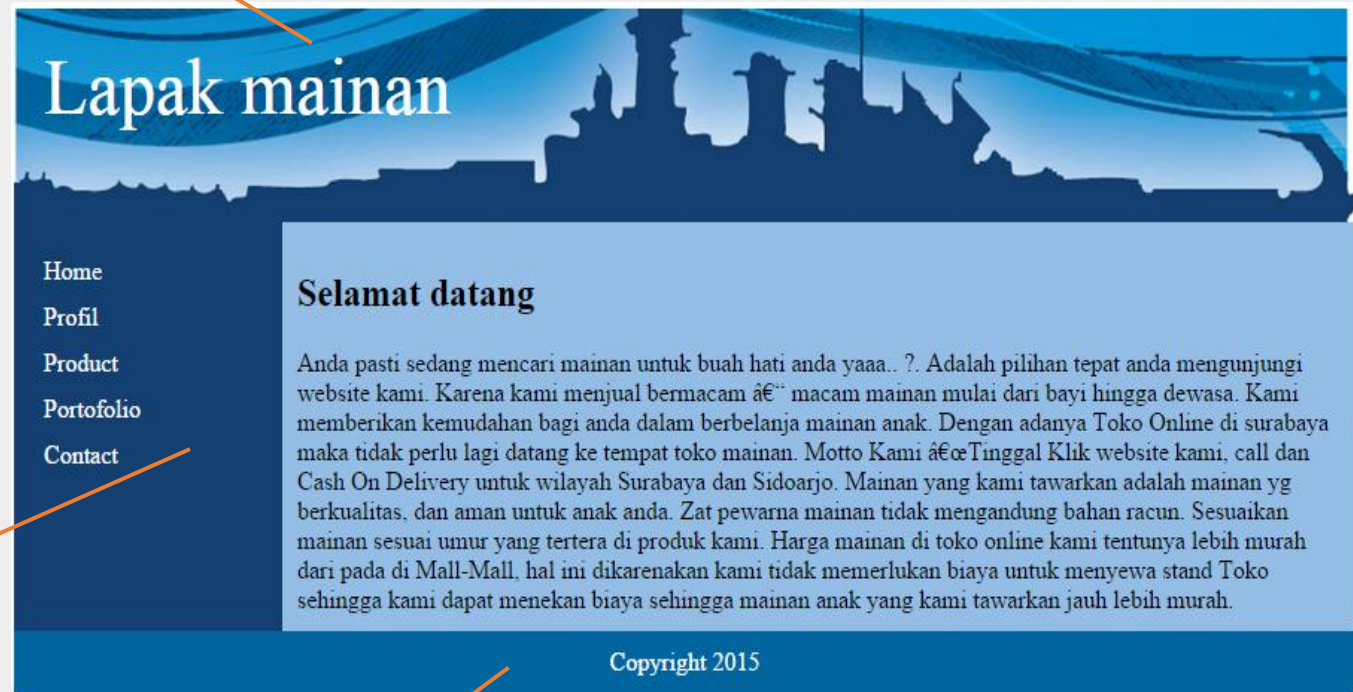
```php
// index.php
include "header.php";
echo "This is The Content";
include "footer.php";
```

```php
// footer.php
echo "This is Footer";
```

Assignment

header.php

menu.php

footer.php