CS 547

Week 1 Day 2

More PHP Language Constructs

Agenda

- Announcements
- Brief Review
- PHP Language Continued

Announcements

- First project next details next class

Review

- Development Tools
- XAMPP from apachefriends.org

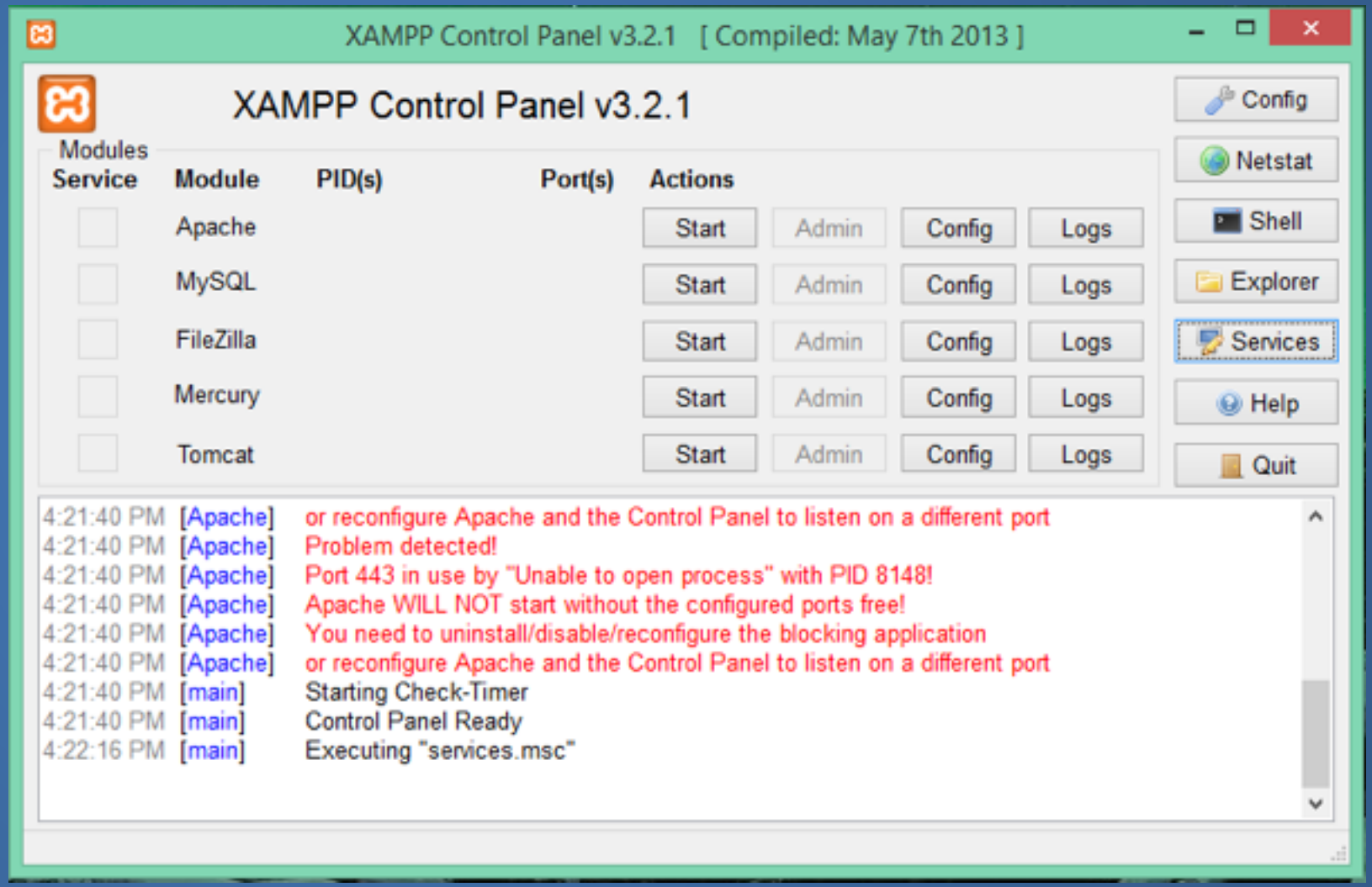
Dev Tools: Download

Download XAMPP from ApacheFriends.Org web site.

XAMPP Control Panel

Last time we talked about Apache

Today we will dive into MySQL



PHP Language

PHP Language

Review: Last session we discussed

Basics

Syntax

Simple variables

PHP Language

-
-

Today:

Variables Continuation…

Arrays

Operators

Control Structures

Special Functions

PHP Variables

PHP is a loosely typed language

```
$index = 1;
…
$index = “One”;
```

These variables are referred to as **Scalar** variables.
Scalar variables contain only one value

PHP Variables: Data Types

- PHP has the following Data Types
  - String
  - Integers
  - Floats or Doubles
  - Booleans: true , false
  - Arrays
  - Objects
  - Null
  - Resource

PHP Variables: String

- <?php
$a = "Hello world.";
- echo $a;
echo "
";
- ?>

PHP Variables: String Concatenation

- <?php
  $a = "Hello”
- $b = “world.";
- echo $a . $b;
  echo "
";
- ?>
- Helloworld

PHP Variables: String Concatenation

- <?php
  $x = "Hello”
- $b = “world.";
- echo $x .” “. $b;
  echo "
";
- ?>
- Hello world

PHP Variables: Integer

- An integer is a whole number (without decimals). It is a number between -2,147,483,648 and +2,147,483,647.
- Rules for integers:
- An integer must have at least one digit (0-9)
- An integer cannot contain comma or blanks
- An integer must not have a decimal point
- An integer can be either positive or negative

PHP Variables: Integer

- Integers can be specified in three formats:

- decimal (10-based) 1234567890
- hexadecimal (16-based - prefixed with 0x) 0x00af
- octal (8-based - prefixed with 0) 0777

PHP Variables: Integer

- <?php
  $a = “101";
- echo $a;
  echo "
";
- var_dump($a);
- ?>
- 101
- Int(101)

PHP Variables: Float

- A number that contains a decimal
- <?php
  $a = “101.01";
-
  echo $a;
  echo "
";
- var_dump($a);
-
  ?>
- 101.01
- Float(101.01)

PHP Variables: Boolean

- Binary states : true or false
- <?php
  $a = true;
-
  echo $a;
  echo "
";
- echo !$a;
-
  ?>
- true
- false

PHP Variables: Arrays

- An array stores multiple values in one single variable.

-
```
<?php
$fruits = array(“Apple",“Banana",“Kiwi");
var_dump($fruits);
?>
```

- array(3) { [0]=> string(5) "Apple" [1]=> string(6) "Banana" [2]=> string(4) “Kiwi" }

PHP Variables: Arrays

- We will examine arrays in greater depth…

PHP Variables: Objects

- Objects are data structures that contain data and methods.
- Must be explicitly declared using a class.

PHP Variables: Objects

-
```
<?php
class Fruit {
 function Fruit() {
$this->type = “Strawberry";
}
}

// create an object
$myFruit = new Fruit();

// show object properties
echo $myFruit >type;
?>
```
- Strawberry

PHP Variables: Objects

- We will examine objects in greater depth…

PHP Variables: Null

- Null is a special data type which can have only one value:
  null
- A variable of data type null is a variable that has no value assigned to it.
- Remember that a variable created without a value is automatically assigned a value of null.
- Variables can also be emptied by setting the value to null

PHP Variables: Null

```
<?php
$a = “Hello Class!";

echo $a;
$a = null;
echo "<br>";
var_dump($a);

?>
```

Hello Class!

NULL

PHP Variables: Resource

- The special resource type is not an actual data type.
- It is the storing of a reference to functions and resources external to PHP.
- A common example of using the resource data type is a database call.

PHP Operators

- Operators are used to perform operations on variables and values.
    - Arithmetic operators
    - Assignment operators
    - Comparison operators
    - Increment/Decrement operators
    - Logical operators
    - String operators
    - Array operators

PHP Operators: Arithmetic

Operator	Name	Example	Result
+	Addition	$x + $y	Sum of $x and $y
-	Subtraction	$x - $y	Difference of $x and $y
*	Multiplication	$x * $y	Product of $x and $y
/	Division	$x / $y	Quotient of $x and $y
%	Modulus	$x % $y	Remainder of $x divided by $y
**	Exponentiation	$x ** $y	Result of raising $x to the $y'th power (Introduced in PHP 5.6)

PHP Operators: Assignment

Assignment	Same as...	Description
x = y	x = y	The left operand gets set to the value of the expression on the right
x += y	x = x + y	Addition
x -= y	x = x - y	Subtraction
x *= y	x = x * y	Multiplication
x /= y	x = x / y	Division
x %= y	x = x % y	Modulus

PHP Operators: Comparison

Operator	Name	Example	Result
==	Equal	$x == $y	Returns true if $x is equal to $y
===	Identical	$x === $y	Returns true if $x is equal to $y, and they are of the same type
!=	Not equal	$x != $y	Returns true if $x is not equal to $y
<>	Not equal	$x <> $y	Returns true if $x is not equal to $y
!==	Not identical	$x !== $y	Returns true if $x is not equal to $y, or they are not of the same type
>	Greater than	$x > $y	Returns true if $x is greater than $y
<	Less than	$x < $y	Returns true if $x is less than $y
>=	Greater than or equal to	$x >= $y	Returns true if $x is greater than or equal to $y
<=	Less than or equal to	$x <= $y	Returns true if $x is less than or equal to $y

PHP Operators: Increment Decrement

•

Operator	Name	Description
++$x	Pre-increment	Increments $x by one, then returns $x
$x++	Post-increment	Returns $x, then increments $x by one
--$x	Pre-decrement	Decrements $x by one, then returns $x
$x--	Post-decrement	Returns $x, then decrements $x by one

PHP Operators: Logical

Operator	Name	Example	Result
and	And	$x and $y	True if both $x and $y are true
or	Or	$x or $y	True if either $x or $y is true
xor	Xor	$x xor $y	True if either $x or $y is true, but not both
&&	And	$x && $y	True if both $x and $y are true
||	Or	$x || $y	True if either $x or $y is true
!	Not	!$x	True if $x is not true

PHP Operators: String

•

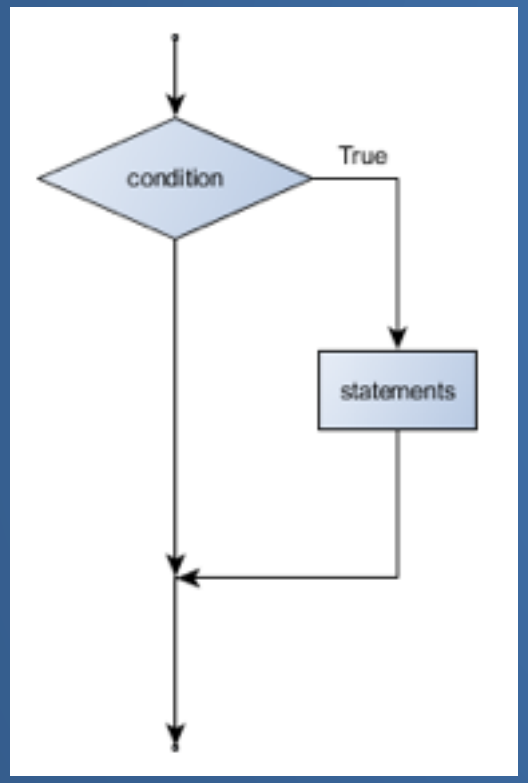
Operator	Name	Example	Result
.	Concatenation	$txt1 . $txt2	Concatenation of $txt1 and $txt2
.=	Concatenation assignment	$txt1 .= $txt2	Appends $txt2 to $txt1

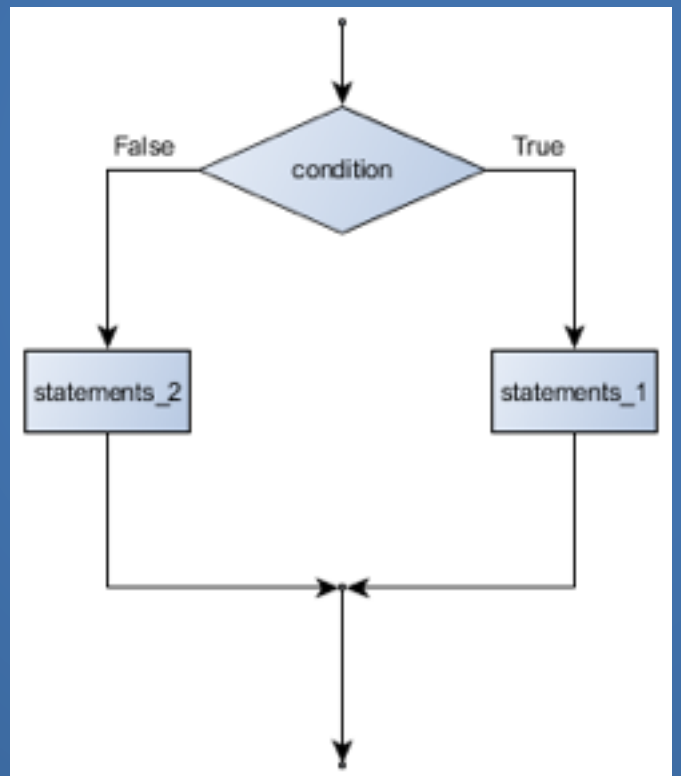
PHP Operators: Array

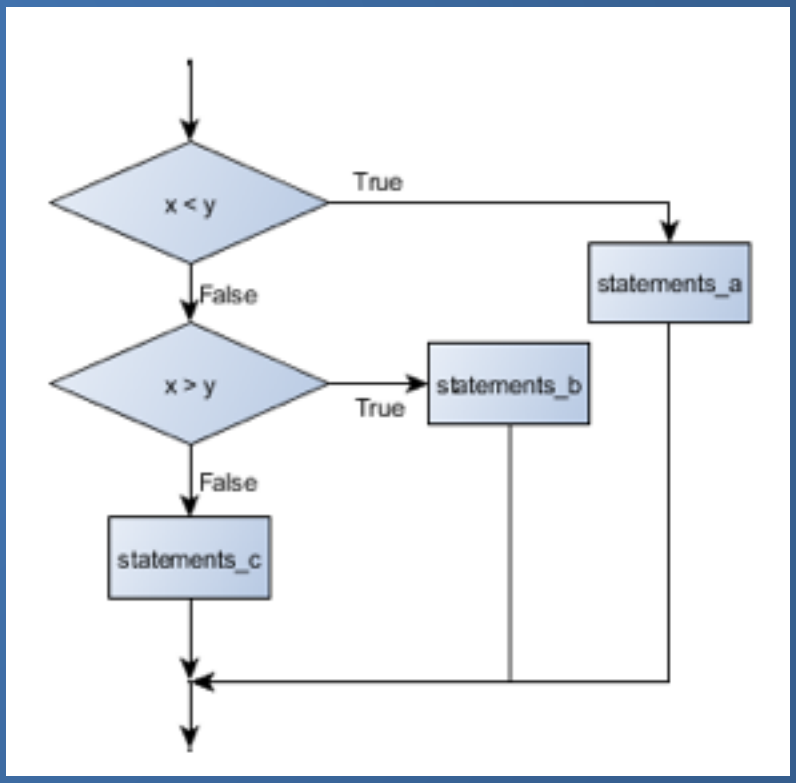
Operator	Name	Example	Result
+	Union	$x + $y	Union of $x and $y
==	Equality	$x == $y	Returns true if $x and $y have the same key/value pairs
===	Identity	$x === $y	Returns true if $x and $y have the same key/value pairs in the same order and of the same types
!=	Inequality	$x != $y	Returns true if $x is not equal to $y
<>	Inequality	$x <> $y	Returns true if $x is not equal to $y
!==	Non-identity	$x !== $y	Returns true if $x is not identical to $y

PHP Conditional Statements

- PHP uses conditional statements to control the execution of code based on different conditions.





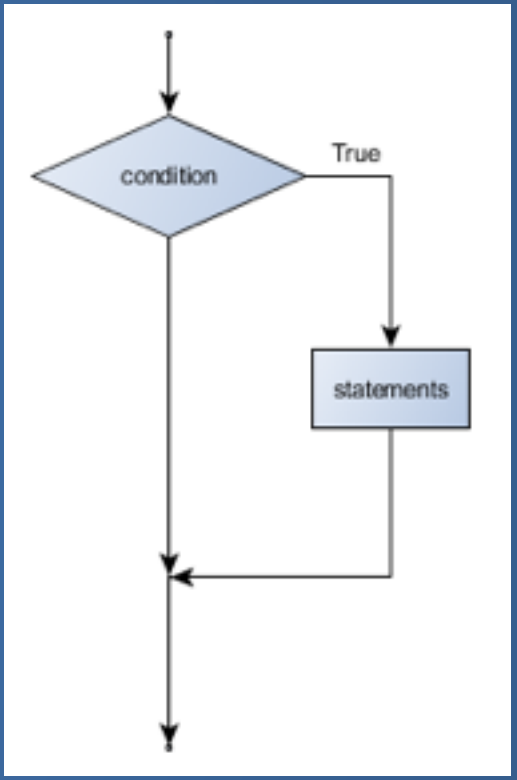


PHP If statement

- Executes code only if condition is true

-
```
<?php
$t = date("H");

if ($t < "20") {
echo "Have a good day!";
}
?>
```

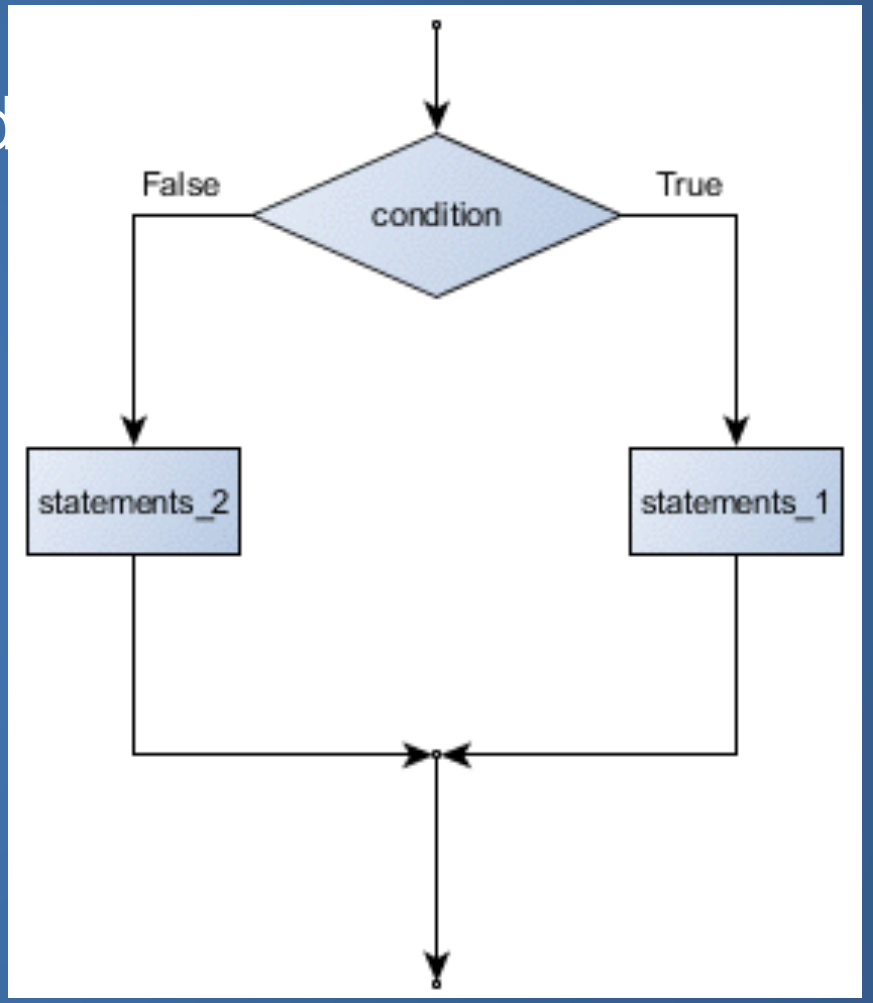


PHP If .. Else Statements

- Use if there are two logical paths for your cod
-

```
<?php
$t = date("H");

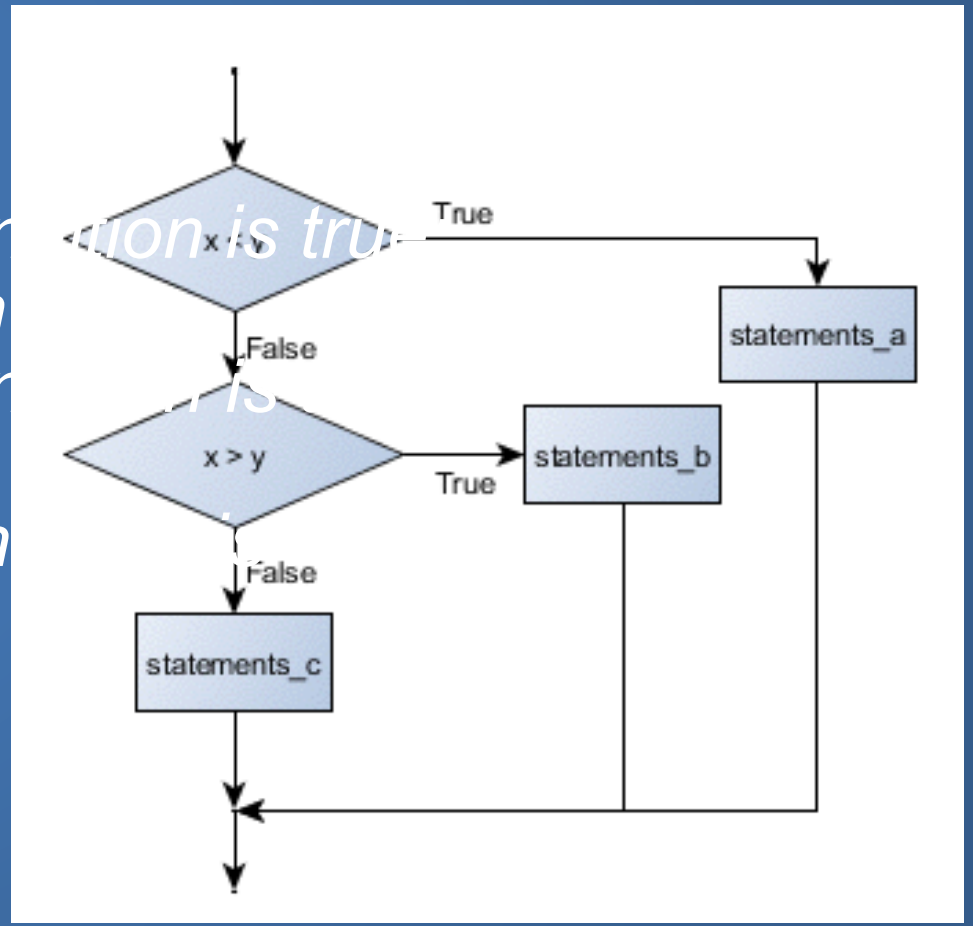
if ($t < "20") {
echo "Have a good day!"; // path 1
} else {
echo "Have a good night!"; // path 2
}
?>
```



if… elseif … else Statement

- Used to specify a new condition to test, **if the first** condition is false.

- if (*condition*) {
code to be executed if con
} elseif (*condition*
code to be executed if con
} else {
code to be executed if con
}



if… elseif … else Statement

-

```
<?php
$t = date("H");

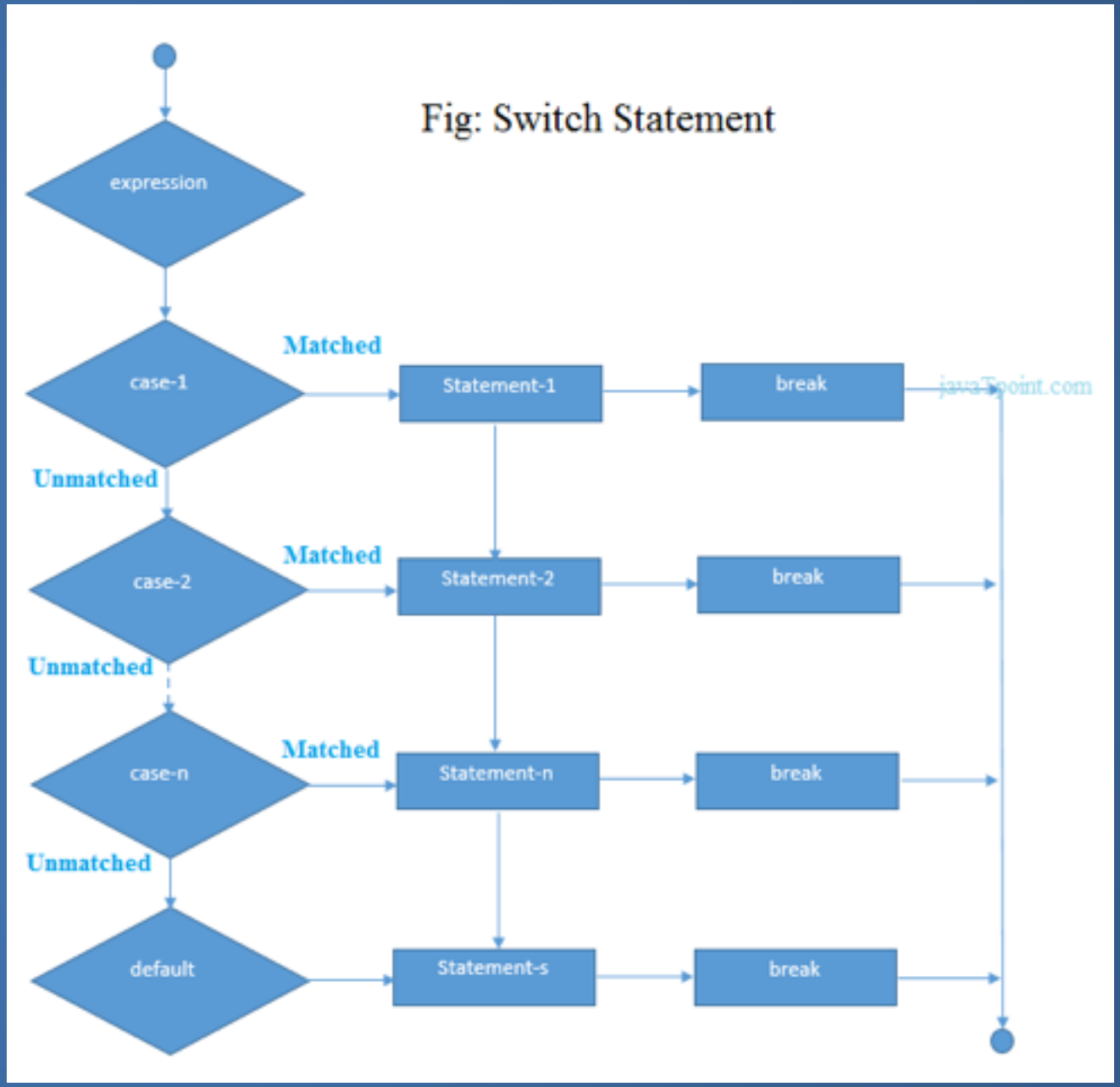
if ($t < "10") {
echo "Have a good morning!";
} elseif ($t < "20") {
echo "Have a good day!";
} else {
echo "Have a good night!";
}
?>
```

Multiple if… elseif … else Statement

- One can ‘chain’ multiple if… else… elseif
- statements together. But consider this image.
- There is a better way…

PHP Switch Stmt

- The switch statement is used to select one of many blocks of code to be executed.
- switch (*n*) {
case *label1:*
code to be executed if n=label1;
break;
case *label2:*
code to be executed if n=label2;
break;
case *label3:*
code to be executed if n=label3;
break;
...
default:
code to be executed if n is different from all labels;
}



PHP Switch Statement

```
<?php
$favcolor = "red";

switch ($favcolor) {
case "red":
echo "Your favorite color is red!";
break;
case "blue":
echo "Your favorite color is blue!";
break;
case "green":
echo "Your favorite color is green!";
break;
default:
echo "Your favorite color is neither red, blue, or green!";
}
?>
```

PHP Loops

- PHP provides the following constructs for loop

- **while** - loops through a block of code as long as the specified condition is true
- **do...while** - loops through a block of code once, and then repeats the loop as long as the specified condition is true
- **for** - loops through a block of code a specified number of times
- **foreach** - loops through a block of code for each element in an array

PHP While Loop

- Syntax

- while (*condition is true*) {
  code to be executed;
  }

```
<?php
$x = 1;

while($x <= 5) {
    echo "The number is: $x <br>";
    $x++;
}
?>
The number is: 1
The number is: 2
The number is: 3
The number is: 4
The number is: 5
```

PHP do … while Loop

- Syntax

- ```
  do {
  code to be executed;
  }
  ```

```
<?php
$x = 1;

do {
    echo "The number is: $x <br>";
    $x++;
} while ($x <= 5);
?>
The number is: 1
The number is: 2
The number is: 3
The number is: 4
The number is: 5
```
```

PHP for Loop

- Syntax
- for (*init counter; test counter; increment counter*) {
  code to be executed;
  }

```
<?php
for ($x = 0; $x <= 10; $x++) {
    echo "The number is: $x <br>";
}
?>
```

The number is: 0
The number is: 1
The number is: 2
The number is: 3
The number is: 4
The number is: 5
The number is: 6
The number is: 7
The number is: 8
The number is: 9
The number is: 10

PHP foreach Loop

- The foreach loop works only on arrays, and is used to loop through each key/value pair in an array.
- Syntax
- foreach ($*array* as $*value*)
{
code to be executed;
}

```
<?php
$colors = array("red", "green", "blue", "yellow");

foreach ($colors as $value) {
    echo "$value <br>";
}
?>
```

```
red
green
blue
yellow
```

PHP

End of lecture on PHP Language

…

Start talking about PHP IDEs

PHP IDEs

An IDE is an

Integrated

Development

Environment

PHP IDEs

Choosing an IDE is a mater of personal choice.

PHP IDEs

- Site point survey on the internet…**Best PHP IDE in 2014 – Survey Results**

Comes up with three popular choices that I can recommend

PhpStorm

Sublime Text 3

and NetBeans

PHP IDEs Homework

- Over the weekend, visit the following pages and evaluate the top three IDE for PHP for yourself. Choose one, and start to use it.

PHP IDEs

PhpStorm
from JetBrains

PHP IDEs

Sublime Text 3

From sublimetext.com

And you may want to read this article too

PHP IDEs

NetBeans
from Netbeans.org

And see this guide on setting up Netbeans with PHP.

XAMPP Live Demo

Switch to XAMPP Control Panel for live demonstration of MySQL and PHPMyAdmin interface