German Jordanian University الجامعة الألمانية الأردنية

CS355 Web Technologies

Dr. Ismail Hababeh

German-Jordanian University

Lecture 12

PHP Object Data Type

- An object is a data type that contains data and information on the data processing.
  - A class of an object must be declared first.
  - An object must be explicitly declared.

PHP Object Data Type - Example

```
<?php
class StudyPlan {
function Course() {
$this->level = "Third";  //$this is a special variable that refers to the current object
    }
}
$web =new Course();       // create course object
echo $web->level; // show object properties
?>
```

Output: Third

PHP NULL Data Type

- NULL is a special data type that only has one value: NULL.
- A variable can be assigned to NULL:
- Example: $var = NULL; or $var = null;
- If a variable is created without a value, it is automatically assigned a value of NULL.
- A variable that has been assigned NULL has the following properties:
  - It evaluates to FALSE in a Boolean context.
  - It returns FALSE when tested with IsSet() function. //changed or not

PHP Resource

- PHP Resource is a special data type that refers to any external resource.
- A resource variable acts as a reference to external source of data such as a file, socket, stream, document, or connection, .. etc.
- PHP uses functions to create resources.
- Resources are handlers to opened files, database connections or image canvas areas.

PHP Resource

- Example 1: The following PHP code is used to create a file resource object reference.

```
$fp = fopen("index.php",'r');
```

- Example 2:

```
$conn = mysqli_connect(localhost,"root","admin","users");
```

This code shows how to get the database resource identifier by requesting the mysqli_connect() function that returns MySQL connection object as a resource identifier.

PHP Constants

- To create a constant, use the define() function.
- Syntax:

define(name, value, [case-insensitive]);

case-insensitive: Specifies whether the constant name should be case-insensitive (default is false).

Example:

```
<?php
        define("GJU", "German Jordanian University", true);
echo gju;
?>
```

Output: German Jordanian University

PHP Operators

- Operators are used to perform operations on variables and values. PHP supports the following operators:
- Arithmetic operators: +, - , * , / , % , ** (Exponentiation)
- Assignment operators: =
- Comparison operators: == , != , <> , < , > , <= , >= ,

=== (identical): Returns true if the compared variables are equal and of the same type.

PHP Operators

!== (not identical): Returns true if the compared variables are not equal or not of the same type.

<=> (Spaceship): Returns an integer less than, equal to, or greater than zero, depending on the compared variables if the first variable is less than, equal to, or greater than the second variable.

PHP Operators

- Increment/Decrement operators: ++$var , $var++ , --$var , $var--
- Logical operators: and(&&) , or(||) , not(!) , xor

  xor is true if either first operand or the second operand is true, but not both.
- String operators: . Concatenation , .= Append

assignment ($txt1 .= $txt2 Appends $txt2 to $txt1)

PHP Operators

- Array operators: used to compare arrays
- + (Union)
- == (Equality)
- != (inequality)
- <> (inequality)
- === (identity)
- !== (not-identity)

PHP Operators

- Conditional assignment operators: used to set a value depending on conditions.
- ?: Ternary

Example 1: $x = expr1 ? expr2 : expr3

The value of $x is expr2 if expr1 = TRUE. The value of $x is expr3 if expr1 = FALSE.

- ?? Null combining

Example 2: $x = expr1 ?? Expr2

If expr1 exists and not null, then the value of $x is expr1.
If expr1 does not exist, or null, then the value of $x is expr2.

PHP Operators Precedence

- Within PHP expression, higher precedence operators will be evaluated first. PHP operators' precedence:
- Unary : -- ++ !
- Multiplicative: * / %
- Additive: + -
- Relational: < <= > >=
- Equality: == !=
- Logical AND: &&
- Logical OR: ||
- Conditional: ?: ??
- Assignment: %= /= *= -= += =

PHP Conditional Statements

- PHP have the following conditional statements forms:
- if statement - executes some code if one condition is true
- if...else statement - executes some code if a condition is true and another code if that condition is false
- if...elseif...else statement - executes different codes for more than two conditions
- switch statement - selects one of many blocks of code to be executed

PHP if .. else Statement - Example

```
<?php
    $d=date("D");
    if ($d=="Friday")
        echo "Have a nice weekend!";
    else
        echo "Have a nice day!";
?>
```

PHP switch Statement - Example

•The Switch statement is used to select one of many blocks of code to be executed.

```
<?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, nor green!";
} ?>
```