

- The PHP concatenation character, i.e., the character used to join strings into a single string, is:  
a.) +      b.) &      c.) ^      d.) ~      e.) -      **f.) .**      g.) :      h.) >
- The statement `$var1 = 5;` declares the variable `$var1` as:  
a.) a boolean      b.) a float      **c.) an integer**      d.) a string      e.) code doesn't force a declaration
- If the variable `$var2` is declared as a float, then later, the statement `$var2 = false;` is executed, which of the following happens?  
a.) The type of `$var2` remains unchanged and the value is changed to 1.  
**b.) The type of `$var2` remains unchanged and the value is changed to 0.**  
c.) The type of `$var2` is changed to boolean and the value false is assigned to it.  
d.) The type and stored value of `$var2` remain unchanged, i.e., the statement has not effect.
- Assume a decimal value is type cast to an integer variable `$i` using the PHP expression `$intval = (int)48.95;` What does `$intval` contain after this expression executes?  
a.) null      b.) 0      **c.) 48**      d.) 49      e.) None of the above
- In the foreach loop shown below, **circle** the variable, `$a`, `$b`, or `$c` that represents the index of the array element when accessed within the loop and **underline** the variable that represents the array.

```
foreach ($a as $b => $c) { ...loop code... };
```

- In the space below, convert/rewrite the following JavaScript code to PHP.

```
if (value >= 0)
    document.write ("The value equals " + value + ".\n");

if ($value >= 0)
    print "The value equals ".$value.".\\n";
```

- The PHP keyword "static" is used to:  
a.) make the PHP code of your scripts visible to the client like JavaScript.  
b.) declare global variables visible to all other functions.  
c.) declare a special data type used for random numbers.  
**d.) declare a variable that remains declared for the duration, but is visible only within its function.**  
e.) create an array of fixed length.
- True/False:** In PHP it is legal to declare an array of mixed types, e.g., `$a = array(false, 9.8, "boy", 5);`

**Use the following array declaration for the next three problems.**

```
$links = array('etsu'=>'http://www.etsu.edu',
              3=>'http://www.yahoo.com',
              'what'=>'http://www.whatis.com');
```

- Fix the code below to ensure that it will not print "Array element 3 is Array[3]" and instead print "Array element 3 is http://www.yahoo.com"

**OR**

```
print "Array element 3 is {$links[3]}";
print "Array element 3 is ".$links[3];
```

10. Assume an additional element were added to the array \$links with the code  
`$links[]='http://www.google.com'`; After which element would it be included in the array?
- a.) 'etsu'                    b.) 3                    **c.) 'what'**                    d.) Can't be predicted
11. Assume an additional element were added to the array \$links with the code  
`$links[]='http://www.google.com'`; What would the index of this new element be?
- a.) 0                    b.) 1                    **c.) 4**                    d.) Can't be predicted

**Use the following code as a reference for problems 12 through 16. The numbers along the left side are line numbers and are included only as a reference. They are not part of the code.**

```

1:  <?php
2:  $conn = mysql_connect ("localhost", "zxyx999", "12345");
3:  mysql_select_db("zxyx999", $conn);
4:  $result = mysql_query("SELECT * FROM timetable ORDER BY COURSE", $conn);
5:  $count=0;
6:  while($record = mysql_fetch_array($result, MYSQL_ASSOC))
7:  {
8:      print "<p>Record {$count}:</p>\n<ul>\n";
9:      foreach($record as $index => $field_value)
10:         print "<li>{$index} is set to {$field_value}</li>\n";
11:         print "</ul>\n";
12:         count++;
13:     }
14:     mysql_close ($connection);
15:     ?>

```

12. What MySQL operation is equivalent to the PHP code on line 3 of the above code?

**use zxyx999; -- selecting the database to use**

13. What MySQL operation is equivalent to the PHP code on line 4 of the above code?

**Performing the query "SELECT \* FROM timetable ORDER BY COURSE;"**

14. There is a syntax error on line 10. What is it?

**There is a semi-colon missing from the end of the line of code.**

15. What is the security risk in letting clients see this source code?

**In line 2, the MySQL database username and password is visible.**

16. \$record is an array. Describe what the values of \$index are going to be, when we print them out as part of the string in line 10.

**\$index contains the field/column names from the table output from the query performed in line 4. The argument MYSQL\_ASSOC in the mysql\_fetch\_array() function call in line 6 forces the index values of the array to be the field names, *not* integer values 0, 1, 2, etc.**