1. Explain the difference between for loops and foreach loops in PHP.

*Essentially, the primary difference between* for *and* foreach *loops in PHP lies in the fact that* for *loops evaluate their conditions prior to executing code as opposed to the* foreach *loop which executes before evaluation, ensuring that the code is always executed at least once.*

1. Explain the difference between do while and while loops in PHP.

*Much like the* for *and* foreach *loops, the difference between* while *and* do while *loops lies in the order in which the code executes. A* while *loop is evaluated prior to execution, while a* do while *loop executes first, ensuring at least a single iteration of the code.*

1. Find and fix the syntax errors. Once fixed, explain why this syntax would not work if $entree was an array.

* echo “my favorite entree is $entree

*The code should read:*

echo “my favorite entree is “ . $entree;

*This syntax would not work for an array, because string concatenation expects a single string object. An item an the array would need to be accessed with an index selector, such as:* $entree[1].

1. Consider the superglobal variables $\_GET, $\_POST, and $\_REQUEST. Explain what they do. How do they differ?

*The* $\_GET *variable contains an array of all the values received from an HTTP GET request, and likewise the* $\_POST *superglobal contains those values from the POST requests. The* $\_REQUEST *variable, on the other hand contains all the variables and metadata from an HTTP request. Data in* $\_POST *typically is transmitted via HTML forms or JSON whereas* $\_GET *only receives data from url parameters and the* $\_REQUEST *contains all of the data in the request, including the other two superglobals.*

1. What does the form action & method tags do in HTML?

*The action attribute in an HTML form determines which PHP page will receive the results or the form submission, while the method attribute determines which HTTP method will be used to transmit the request, which would typically be “POST.”*

1. Write a code snippet in php. How would you access every key in array, xyz, using a foreach loop, assigning a value to every key and print that to the screen?

*To print each value in array xyz using foreach use the code:*

$xyz = [0 => ‘x’, ‘y’, ‘z’];

foreach ($xyz as $key => $value) {

echo $xyz[$key];

}

1. Same as the question in #6, but use a for loop. For this loop, consider that you only have 10 keys in the array.

*To print each value in array xyz using a for loop use the code:*

$xyz = range(1, 10);

for ($i = 1; $i <= 10; $i++) {

echo $xyz[$i];

}

1. Write a code snippet in php. Write an array called flavors and add chocolate, vanilla, strawberry, and cherry to the array in the order listed.

*Flavors Array:*

$flavors = [0 => ‘chocolate’, 1 => ‘vanilla’, 2 => ‘strawberry’, 3 => ‘cherry’];

1. What does the code, isset($var), do?

*The code* isset($var) *evaluates* $var *and returns true if* $var *points to a non-null value or false if it doesn’t*.

1. Explain why the else condition should be the condition that traps errors in an if conditional statement.

*The else condition should capture errors because you only want code that meets the desired conditions to execute and let the else condition catch all unexpected behaviors and throw an error.*