1. According to the author, what are the steps to effectively debug HTML errors?

*The author suggests four step to debug HTML:*

1. *Check the source code.*
2. *Use a validation tool.*
3. *Use a browser’s debugger in the dev tools.*
4. *Test in multiple browsers.*
5. What does this do: ini\_set(‘display\_errors’, 0); ?

*Using the function ini\_set() with ‘display\_errors’ as the first parameter toggles whether or not PHP displays errors in its output depending on the value of the second parameter. By setting the second value to zero, PHP will not display these errors.*

1. What character is used to suppress errors?

*The @ character is used to suppress error reporting when used just prior to an expression.*

1. What does this do: error\_reporting(E\_ALL & ~E\_Notice); ?

*The function error\_reporting(E\_ALL & ~E\_Notice); will cause PHP to report all errors that are not notices.*

1. What are the 7 PHP errors? What is the likely cause of the error?

*According to the author, the seven most common PHP errors are:*

1. *Blank page – Likely caused by an HTML error or PHP error notifications ae disabled.*
2. *Parse Error – Likely caused by an excess or missing character such as a semicolon or parentheses.*
3. *Empty Variable Value – Either the variable is missing the $ character, is misspelled or is out of scope.*
4. *Undefined Variable – Caused when a variable is referenced before it receives a value, or when the value references nothing.*
5. *Call to Undefined Function – Likely caused by a misspelled function name or the document containing the function’s definition was not referenced where the function was called.*
6. *Cannot Redeclare Function – Caused when a function is defined in more than one location.*
7. *Headers Already Sent – Likely caused when the script contains whitespace prior to the PHP tags, printed data or included files.*
8. According to the author, how would you debug PHP errors?

*According to the author PHP can be debugged by first ensuring that ‘display\_errors’ is enabled. Second, comments can be used to block out sections of the script from running so that the error can be isolated and fixed. Finally, the author suggests the use of the print() and echo() functions to debug the steps of more complex expressions or die() and exit() for complex scripts.*

1. According to the author, how would you debug SQL queries?

*According to the author, SQL queries can be debugged by first using echo() to print the offending query to make sure PHP is supplying the desired dynamic values. Next, the author suggests running the query through an external tool, such as PhpMyAdmin and finally attempting to run the query broken down into each of its individual parts to find the part of the query that is causing the error.*