www.learntocreategames.com

Common Coding Errors

Please Keep Close by While Coding ☺

- **";" expected**: This error could mean that you have forgotten to add a semi-colon at the end of a statement. To fix this error, just go to the line mentioned in the error message and ensure that you add a semi-colon. It can also be due to the fact that you could have forgotten the keyword **function** as you defined a function.

- **Unknown identifier**: This error could mean that Unity does not know the variable that you are mentioning. It can be due to at least three reasons: (1) the variable has not been declared yet, (2) the variable has been declared but outside the scope of the function (e.g., declared locally in a different function), or (3) the name of the variable that you are using is incorrect (i.e., spelling or case). Remember, the names of all variables and functions are case-sensitive; so by just using an incorrect case, Unity will assume that you refer to another variable, which, in this case, has not been declared yet.

- **The best method overload for function … is not compatible**: This error is probably due to the fact that you are trying to call a function and to pass a parameter with a type that is not what Unity is expecting. For example, the function **mySecondFunction**, described in the next code snippet, is expecting a String value for its parameter; so, if you pass an integer value instead, an error will be generated.

- **Expecting } found …:** This error is due to the fact that you may have forgotten to either close or open curly brackets. This can be the case for conditional statements or functions. To avoid this issue, there is a trick (or best practice) that you can use: you can ensure that you indent your code so that corresponding opening and closing brackets are at the same level. In the next example, you can see that the brackets corresponding to the start and end of the function **testBrackets** are indented at the same level, and so are the brackets for each of the conditional statements within this function. By indenting your code (using several spaces or tabulation), you can make sure that your code is clear and that missing curly brackets are easier to spot.