Following are the notable differences between Class (static) and instance variables.

|  |  |
| --- | --- |
| **Instance variables** | **Static (class) variables** |
| Instance variables are declared in a class, but outside a method, constructor or any block. | Class variables also known as static variables are declared with the static keyword in a class, but outside a method, constructor or a block. |
| Instance variables are created when an object is created with the use of the keyword 'new' and destroyed when the object is destroyed. | Static variables are created when the program starts and destroyed when the program stops. |
| Instance variables can be accessed directly by calling the variable name inside the class. However, within static methods (when instance variables are given accessibility), they should be called using the fully qualified name. *ObjectReference.VariableName*. | Static variables can be accessed by calling with the class name *ClassName.VariableName*. |
| Instance variables hold values that must be referenced by more than one method, constructor or block, or essential parts of an object's state that must be present throughout the class. | There would only be one copy of each class variable per class, regardless of how many objects are created from it. |