* The yield keyword returns the next value from an iterator or ends an iteration. See Using yield.
* The special identifier value is used in a property's write accessor to represent the value requested for assignment to the property.
* The special identifier set is used to declare the write accessor for a property.
* The special identifier partial is used to allow developers to build classes from different files and have the compiler generate one class (combining all the partial classes).
* The special identifier get is used to declare the read accessor for a property.
* It can sometimes be necessary to reference two versions of assemblies that have the same fully-qualified type names, for example when you need to use two or more versions of an assembly in the same application. By using an external assembly alias, the namespaces from each assembly can be wrapped inside root-level namespaces named by the alias, allowing them to be used in the same file.

To reference two assemblies with the same fully-qualified type names, an alias must be specified on the command line,

as follows: /r:GridV1=grid.dll /r:GridV2=grid20.dll

This creates the external aliases GridV1 and GridV2. To use these aliases from within a program, reference them using the extern keyword.

For example: extern alias GridV1; extern alias GridV2;

Each extern alias declaration introduces an additional root-level namespace that parallels (but does not lie within) the global namespace. Thus types from each assembly can be referred to without ambiguity using their fully qualified name, rooted in the appropriate namespace-alias

In the above example, GridV1::Grid would be the grid control from grid.dll, and GridV2::Grid would be the grid control from grid20.dll.

* The while keyword identifies a while loop
* The volatile keyword is used to declare a variable which may change its value over time due to modification by an outside process, the system hardware, or another concurrently running thread.
* The void keyword is used in method signatures to declare a method that does not return a value. A method declared with the void return type cannot provide any arguments to any return statements they contain.
* The unsafe keyword may be used to modify a procedure or define a block of code which uses unsafe code. Code is unsafe if it uses the "address of" operator(&) or if it uses a pointer operator (\*)
* The unchecked keyword prevents overflow-checking when doing integer arithmetic. It may be used as an operator on a single expression or as a statement on a whole block of code.