

Data Types

bool	Boolean value
byte	8-bit unsigned integer
char	16-bit Unicode character
decimal	128-bit precise decimal values with 28-29 significant digits
double	64-bit double-precision floating point
float	32-bit single-precision floating point
int	32-bit signed integer
long	64-bit signed integer
object	Base type for all other types
sbyte	8-bit signed integer
short	16-bit signed integer
string	String value
uint	32-bit unsigned integer
ulong	64-bit unsigned integer
ushort	16-bit unsigned integer

Arrays

```
int[] array = new int[] {1,2,3}
int[] array = {1,2,3}
var array = new int[] {1,2,3}
int[] array = new int[3]
```

Classes

Class	public class Dog {...}	
Inheritance	public class Dog: Pet {...}	
Constructor (no parameters)	public Dog () {...}	Constructors can co-exist
Constructor (one parameter)	public Dog (string var) {...}	Constructors can co-exist
Field	public string name	
Static Class	public static class Dog {...}	Must only have static members
Static Member	public static int = 1	
Finalizer (destructor)	~Dog () {...}	Cannot have modifiers or parameters

Naming Conventions

Class	MyClass
Method	MyMethod
Local variable	myLocalVariable
Private variable	_myPrivateVariable
Constant	MyConstant

Logical Operators

&&	and
	or
!	not
^	xor

Arithmetic Operators

+	Add numbers
-	Subtract numbers
*	Multiply numbers
/	Divide numbers
%	Compute remainder of division of numbers

Statements

if-else	if (true) {...} else if (true) {...} else {...}
switch	switch (var) { case 1: break; default: break; }
for	for (int i = 1; i < 5; i++) {...}
foreach-in	foreach (int item in array) {...}
while	while (true) {...}
do... while	do {...} while (true);
try-catch-finally	try {...} catch (Exception e) {...} catch {...} finally {...}

Access Modifiers

public	Accessible by any other code in the same assembly or another assembly that references it
private	Only accessible by code in the same class or struct



Access Modifiers (cont)

protected	Only accessible by code in the same class or struct, or in a derived class
internal	Accessible by any code in the same assembly, but not from another assembly
protected internal	Accessible by any code in the same assembly, or by any derived class in another assembly

Other Modifiers

abstract	Indicates that a class is intended only to be a base class of other classes
async	Indicates that the modified method, lambda expression, or anonymous method is asynchronous
const	Specifies that the value of the field or the local variable cannot be modified
event	Declares an event
extern	Indicates that the method is implemented externally
new	Explicitly hides a member inherited from a base class
override	Provides a new implementation of a virtual member inherited from a base class
partial	Defines partial classes, structs and methods throughout the same assembly
readonly	Declares a field that can only be assigned values as part of the declaration or in a constructor in the same class
sealed	Specifies that a class cannot be inherited
static	Declares a member that belongs to the type itself instead of to a specific object
unsafe	Declares an unsafe context
virtual	Declares a method or an accessor whose implementation can be changed by an overriding member in a derived class
volatile	Indicates that a field can be modified in the program by something such as the operating system, the hardware, or a concurrently executing thread

Comparison Operators

<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
==	Equal to
!=	Not equal to

