Identifiers

An identifier is a name used to identify a class, variable, function, or any other user-defined item.

The basic rules for naming classes in C# are as follows:

- A name must begin with a letter that could be followed by a sequence of letters, digits (0 – 9) or underscore. The first character in an identifier cannot be a digit.
- It must not contain any embedded space or symbol such as? – + ! @ # % ^ & * ( ) [ ] { } . ; : " ' / and \. However, an underscore (_) can be used.
- It should not be a C# keyword.

Keywords

- Keywords are **reserved words** predefined to the C# compiler. These keywords cannot be used as identifiers.
- However, if you want to use these keywords as identifiers, you may prefix the keyword with the @ character.
- In C#, some identifiers have special meaning in context of code, such as get and set are called **contextual keywords**.

Reserved Keywords

abstract	as	base	bool	break	byte	case
catch	char	checked	class	const	continue	decimal
default	delegate	do	double	else	enum	event
explicit	extern	false	finally	fixed	float	for
foreach	goto	if	implicit	in	in (generic modifier)	int
interface	internal	is	lock	long	namespace	new
null	object	operator	out	out (generic modifier)	override	params
private	protected	public	readonly	ref	return	sbyte
sealed	short	sizeof	stackalloc	static	string	struct
switch	this	throw	true	try	typeof	uint
ulong	unchecked	unsafe	ushort	using	virtual	void
volatile	while					

Contextual Keywords

add	alias	ascending	descending	dynamic	from	get
global	group	into	join	let	orderby	partial (type)
partial (method)	remove	select	set			

Comments

Comments are used for explaining code. Compilers ignore the comment entries.

The **multiline comments** in C# programs start with /* and terminates with the characters */

```
/* This program demonstrates
The basic syntax of C# programming
Language */
```

Single–line comments are indicated by the '//' symbol

```
}//end class Rectangle
```