C# Programming Reference Sheet

Built In Data Types & Literals

Integers
```
short, ushort, int, uint, long, ulong
(eg: -3, 6, 27)
```
Floating Point Numbers
```
float, double, decimal
(eg: 3.14, 1.41, 1.61)
```
Strings and Characters
```
string, char (eg: "Hello", 'A')
```
Boolean
```
bool (eg: true, false)
```

Working with Strings

Assignment (giving a string a value)
```
name = "Fred";
```
Concatenation (joining strings)
```
name += " Smith";
```
Comparison
```
if name == "Fred Smith" { ... }
```
Construction from other types:
```
name = "R" + 2.ToString() + "D" +
   2.ToString();
```

Simple Programming Statements

Constant declaration
```
const float PI = 3.1415;
```
Variable declaration
```
string name; int age;
```
Assignment
```
name = "Fred"; age = 10;
```
Method call
```
WriteLine("Hello"); names.Add(name);
```
Sequence of statements - grouped
```
{ ... }
```

Structured Programming Statements

If statement
```
if done { ... } else { ... }
```
Case statement
```
switch (age) { case 1: ... default: ... }
```
While loop
```
while (!done) { ... }
```
Repeat loop
```
do { ... } while (!done);
```
For loop
```
for (int i = 0; i < 10; i++) { ... }
```

Declaring Methods

Declare a method with parameters:
```
void SayHello(string toName) { ... }
```
Declare a method that returns data:
```
Contact ReadContact() {
   Contact result = ...;
   return result;
}
```
Pass by reference:
```
void Swap(ref int v1, ref int v2);
void Init(out int n);
```

Boolean Operators and Other Statements

Comparison: equal, less, larger, not equal, less eq
```
==, <, >, !=, >=
```
Boolean: And, Or and Not
```
&&, ||, !
```
Skip an iteration of a loop
```
continue;
```
End a loop early
```
break;
```
End a method:
```
return;
```

Custom Types

Classes
```
class Contact {
   public string name;
}
```
Enumerations
```
enum Grade { PASS, CREDIT, ... }
```
Structs
```
struct Position {
   public float x; public float y;
}
```

Arrays

Declaration
```
int[5] scores;
List<Contact> friends;
```
Access
```
scores[0] = 10;
friends[0] = ReadContact();
```
Loop with index i
```
for (int i = 0; i < scores.Length; i++);
```
For each loop
```
foreach (Contact friend in friends);
```

Programs and Modules

Creating a program
```
namespace HelloWorld {
   class Program { ... }
}
```

Using a class from a library
```
using MyLibrary;

MyClass myClass = new MyClass();
myClass.Print();
```

Other Things

Reading from Terminal
```
Console.ReadLine();
```
Writing to Terminal
```
Console.WriteLine(string.Format("Hello
   {0} aged {1}", name, age);
```
Comments
```
// single line
/* multi-line
   comment
*/
```