# C# Programming Reference Sheet

Working with Strings

Assignment (giving a string a value): Test = “Hello World”

Concatenation (joining strings): String Test = “Hello” + “World”

Comparison “hd” == “hd”

Construction from other types:x = X.ToString

Built In Data Types & Literals

Integers: int, unit, long

Eg: 50, 28

Floating Point Numbers: float, double

Eg: 5.5

Strings and Characters: char, string

Eg: “B” , “Ben”

Boolean: Boolean eg: True/False

Structured Programming Statements

If statement: If (correct statement) then {} else

Case statement : Switch (x) {case1:Console.WriteIn

(“Case1”);break;case 2:….;break;}

While loop: while () do {}

Repeat loop: do {} while ()

For loop: for () do ()

Simple Programming Statements

Constant declaration: Const int hour = 4;

Variable declaration: Int x = 2;

Assignment a = 7

Method call : <Access Specifier> <Return Type>

<Method Name> (Parameter List) {Method Body}

Sequence of statements – grouped: Class program {}

Declaring Methods

Declare a method with parameters: public int CombineTwoNumbers (int num1, int num2) {return num1 + num2}

Declare a method that returns data: public int CombineTwoNumbers (int num1, int num 2) {return num1 + num2}

Pass by reference: Num = 1;

squareReference(ref num); Console.WriteLine (num)

Boolean Operators and Other Statements

Comparison: equal, less, larger, not equal, less eq :

Equal (==), less (<), larger (>), not equal (! =),less equal to (<=)

Boolean: And, Or and Not: And(&&), Or (||), Not (!)

Skip an iteration of a loop: Continue;

End a loop early: Break;

End a method: Return;

Custom Types

Classes: Public class message { }

Enumerations: Enum Car {Mazda, Honda}

Structs: Struct Authours

{public string firstname; public string lastname;}

Arrays

Declaration: Int [ ] numbers;

Access: Number [ 2 ] = 5

Loop with index I : For (i = 0;i<3; i ++) {number [i] = I;}

For each loop: foreach (int… in …) {}

Programs and Modules

Creating a program

Class MainClass

{ public static void Main ()

{ }

Using a class from a library

Using GameLibrary;

Public void Draw () {

GameLibrary.DrawSquare (\_colour, \_x, \_y, \_width, \_height);}

Other Things

Reading from Terminal

Console.Readline ()

Writing to Terminal

Console.Writeline ()

Comments

// Comment on single line