

Pseudocode

Keywords and Examples

Refer to the following definitions and examples of pseudocode:

Description	Pseudocode example	C# equivalent
Declaring a variable	<pre>DECLARE iNum DECLARE sName DECLARE bAlive DECLARE arrInts[3] DECLARE arr2DInts[2, 2]</pre>	<pre>int iNum; string sName; bool bAlive; int[] arrInts = new int[3]; int[,] arr2DInts = new int[2, 2];</pre>
Initialising a variable	<pre>INIT iNum = 0 INIT sName = "Mr. Atzeni" INIT bAlive = true</pre>	<pre>int iNum = 0; string sName = "Mr. Atzeni"; bool bAlive = true;</pre>
Initialising an array	<pre>INIT arrInts = [1, 2, 3] INIT arr2DInts = [{1, 2}, {3, 4}]</pre>	<pre>int[] arrInts = { 1, 2, 3 }; int[,] arr2DInts = { {1, 2}, {3, 4} };</pre>
Updating a variable value after it has been initialised with a value	<pre>SET iNum = iNum * 10 SET sName = "Gerard Atzeni" SET bAlive = !bAlive SET arrInts[0] = 10</pre>	<pre>iNum = iNum * 10; sName = "Gerard Atzeni"; bAlive = !bAlive; arrInts[0] = 10;</pre>
Displaying output	<pre>OUTPUT "What is your name?" PRINT "You are " + iNum + " years old." // PRINT and OUTPUT used interchangeably</pre>	<pre>Console.WriteLine("What is your name?"); Console.WriteLine("You are " + iNum + " years old.");</pre>
Receiving user input	<pre>INPUT sName = "What is your name?" INPUT iNum = "How old are you?" as integer</pre>	<pre>Console.WriteLine("What is your name?"); sName = Console.ReadLine(); Console.WriteLine("How old are you?"); iNum = Convert.ToInt16(Console.ReadLine());</pre>
Selection (if statement)	<pre>IF iNum > 0 AND bAlive == true THEN OUTPUT "You are okay!" ENDIF</pre>	<pre>if (iNum > 0 && bAlive) { Console.WriteLine("You are okay!"); }</pre>
Selection (if-else statement)	<pre>IF iNum < 10 THEN OUTPUT "Sorry, not enough!" ELSE OUTPUT "Woah, too much!" ENDIF</pre>	<pre>if (iNum < 10) { Console.WriteLine("Sorry, not enough!"); } else { Console.WriteLine("Woah, too much!"); }</pre>

		<pre> }</pre>
Selection (else-if statement)	<pre> IF dGrade >= 100 THEN OUTPUT "Perfect!" ELSE IF dGrade >= 50 THEN OUTPUT "Not too bad..." ELSE OUTPUT "Hmm. Needs work!" ENDIF</pre>	<pre> if (dGrade >= 100) { Console.WriteLine("Perfect!"); } else if (dGrade >= 50) { Console.WriteLine("Not too bad..."); } else { Console.WriteLine("Hmm. Needs work!"); } }</pre>
Selection (switch statement)	<pre> SWITCH sName CASE "Mr. Atzeni" OUTPUT "Meh." CASE "Dr. Atzeni" OUTPUT "Now we're talking." DEFAULT OUTPUT "Who?" ENDSWITCH</pre>	<pre> switch (sName) { case "Mr. Atzeni": Console.WriteLine("Meh."); break; case "Dr. Atzeni": Console.WriteLine("Now we're talking."); break; default: Console.WriteLine("Who?"); break; } }</pre>
Iteration (while loop)	<pre> INIT i = 0 WHILE i < 100 DO OUTPUT i SET i = i + 1 ENDWHILE</pre>	<pre> int i = 0; while (i < 100) { Console.WriteLine(i); i = i + 1; } }</pre>
Iteration (do-while loop)	<pre> INIT i = 0 DO OUTPUT i SET i = i + 1 WHILE i < 100</pre>	<pre> int i = 0; do { Console.WriteLine(i); i = i + 1; } while (i < 100); }</pre>
Iteration (for loop)	<pre> INIT i = 0 FOR i <= 100 STEP 1 DO OUTPUT i ENDFOR</pre>	<pre> for (int i = 0; i <= 100; i++) { Console.WriteLine(i); } }</pre>
Iteration (foreach loop)	<pre> INIT arrInts = [1, 2, 3] FOR EACH iNum IN arrInts DO</pre>	<pre> int[] arrInts = { 1, 2, 3 }; foreach (var iNum in arrInts) {</pre>

	OUTPUT "Displaying " + iNum ENDFOR	Console.WriteLine("Displaying " + iNum); }
Modularisation (procedure)	BEGIN resetValues SET iNum = 0 SET sName = "Gerard Atzeni" SET bAlive = true OUTPUT "Reset complete." END	public static void resetValues() { iNum = 0; sName = "Gerard Atzeni"; bAlive = true; Console.WriteLine("Reset complete."); }
Modularisation (function)	BEGIN getArea (iWidth, iHeight) INIT iArea = iWidth * iHeight RETURN iArea END	public static int getArea(int iWidth, int iHeight) { int iArea = iWidth * iHeight; return iArea; }
Modularisation (calling a procedure or function)	CALL resetValues INIT iArea = getArea(3, 5)	resetValues(); int iArea = getArea(3, 5);
Global variables	DECLARE sName BEGIN resetValues SET iNum = 0 SET sName = "Gerard Atzeni" SET bAlive = true OUTPUT "Reset complete." END	string sName; public static void resetValues() { iNum = 0; sName = "Gerard Atzeni"; bAlive = true; Console.WriteLine("Reset complete."); }
Object-oriented programming (class declaration)	CLASS Student { iAge = 14 sName = "Mary" arrSubjects = ["DIG", "MAT"] }	public class Student { public int iAge = 14; public string sName = "Mary"; public string[] arrSubjects = { "DIG", "MAT" }; }
Object-oriented programming (object instantiation)	INSTANTIATE Student AS oStudent	Student oStudent = new Student();
Object-oriented programming (lists)	INSTANTIATE List<Person> AS lstPeople	List<Person> lstPeople = new List<Person>();
Object-oriented programming (properties)	INSTANTIATE Student AS oStudent SET oStudent.iAge = 15	Student oStudent = new Student(); oStudent.iAge = 15;
Object-oriented programming (methods)	INSTANTIATE Random AS oRandom INIT iRandom = oRandom between 1 and 100	Random oRandom = new Random(); int iRandom = oRandom.Next(0, 101);