E – Expressions and Math

# Overview

This topic introduces concepts surrounding the use of arithmetic expressions and common Math library routines.

This topic will introduce the following grammars, syntax and language constructs. (Note that additional concepts from previous topics may also be present.)

* Arithmetic expressions
* Assignment statements
* Automatic type conversion and casting
* Integer division
* String concatenation
* Math library routines and constants (such as Random, Square Root, and **π**)

This topic will also take a deeper look at the distinctive aspects of variables, values and data types as well as demonstrate the use of overloaded constructors. This is also the first foray into creating classes that actually “do something” – these classes not only maintain state but also provide methods to generate other information besides that which is stored in a class’ fields.

**Daily LOGs**

The following daily LOGs are covered in this package.

**OOP Basics**

* Explain how method overloading is applied to constructors

**General Programming Concepts and Terms**

* Name at least three kinds of program statements commonly seen in all high-level programming languages
* Distinguish between Variables, Values and Data Types
* List the five basic rules governing variables, values and data types
* List the intrinsic (built-in) data types in C#
* Explain what is meant by the phrase “type extensibility”
* Define the terms Reference Type and Value Type as they apply to data types in C#
* Define the term Identifier
* Create and initialize variables in C#
* Explain what is meant by the terms “scope” and “lifetime” as they apply to variables
* Describe what is meant by the term "expression"
* Explain the difference between program statements and expressions as well as how the two relate to each other
* List the three general sets of operators in C#
* Describe the two basic rules of using arithmetic operators that every compiler must follow.
* Use the arithmetic operators to create arithmetic expressions
* Explain what is meant by “integer division” and how that can be useful in solving particular problems regarding numbers.
* Explain the purpose of the modulus (%) operator.
* List and describe how to use the various assignment operators in C#
* Explain the difference between binary and unary operators
* Demonstrate understanding of operator precedence and how to override the default precedence of the arithmetic operators
* Summarize and distinguish the rules for automatic type conversion involving arithmetic operators and the assignment operator
* Determine the final data type for expressions involving mixed data types and type conversion
* Describe “type casting” and explain how to use it appropriately.
* Compare and contrast the prefix and postfix behaviour of the unary arithmetic operators
* Identify and explain how to use the common Math library routines (for Power, Square Root, Absolute, random numbers, pi, trigonometry, and the various types of rounding)
* Use Math rounding methods with arithmetic expressions to round real numbers to a specific precision
* Create random whole number values within a specific range
* Use type casting appropriately in C#
* Create constants in C#
* Explain how arithmetic operations affect character values in C#
* List the most common math functions of the Math class
* Demonstrate how to use the various Math functions in simple programs
* List and describe the commonly used fields and methods of the String class
* Demonstrate how to use "String Arithmetic" (concatenation)
* List the equivalent “wrapper” classes for each of the primitive types.