**01 Introduction to C# and Data Types**

Candidate Name: Omprakash Narayanan Lakshmanan

**Understanding Data Types**

1. Data type for

* A person’s telephone number – string
* A person’s height – float
* A person’s age – byte
* A person’s gender – Enum
* A person’s salary – decimal
* A book’s ISBN – string
* A book’s price – float
* A book’s shipping weight – float
* A country’s population – long
* The number of stars in the universe – long
* The number of employees in small or medium businesses – ushort or int

2. Value type – Variables of value type directly contain the data. When two value typed variables contain the same data, change in one variable doesn’t affect the other variable. It is stored in stack memory.

Reference type – Variables of reference type contain references to the data. When two variables reference the same data, change in one variable affects the other variables. It is stored in heap memory.

Boxing – The process of converting a value type variable to a reference type variable(object)

Unboxing - The process of converting reference type variable(object) to a value type variable

3. Managed objects are created, managed and are under the scope of Common Language Runtime (CLR).

Unmanaged resources are out of the scope of CLR such as database connections and file streams.

4. Garbage collector manages the allocation and release of memory of an application automatically. Automatic memory management can eliminate problems that cause memory leaks or attempt to access memory for an object that has already been freed.

**Controlling Flow and Converting Types**

1. When we divide an int by 0, we get a DivideByZeroException
2. When we divide a double by 0 we don’t get any exception. Floating point values have FloatingPointType.PositiveInfinity and FloatingPointType.NegativeInfinity.
3. ArgumentOutOfRangeException is thrown when we overflow an int variable.
4. x = y++ is post increment, Let’s say y = 1 before this statement, x will be set to 1 and after the given statement is executed, y will be incremented to 2.

x = ++y is pre increment, for the same example, y will be incremented first and x will be set to 2.

1. break – It breaks out of a loop and the statements after the loop will be executed.

continue – It stops the current iteration and begins the next iteration.

return – It will exit the entire method or function that is currently executing.

1. The three parts of a for statement are initializer, condition, and iterator. All the sections are optional.
2. = is the assignment operator. It is used to assign data to a variable.

== is the equality operator. It checks if the operands are equal or not.

1. Yes, for(;true;); is valid
2. The underscore \_ character replaces the **default** keyword to signify it should match anything if reached.
3. An object must implement **IEnumerable** interface to be enumerated over by using the foreach statement.

**Practice loops and operators**

int max = 500;  
for (byte i = 0; i < max; i++)  
{  
 WriteLine(i);  
}

The code gives “The name ‘WriteLine’ does not exist in the current context”. We can add “use static System.Console” to add the namespace.