Activity 7 : NeDL Transition Academy

Programming Activity 7: Types.

.NET support many types of things. In this activity you need to document the use of each type of things. When you would use it. And how it compares to other types.

Step 1: Compare and contrast the difference between Class, Interface, and Struct. Create code samples showing the differences in how they are used.

Class

* Contains the implementation logic
* Contains data members, fields, properties, methods, constants, etc.
* Business logic
* Implements the interface (when applicable)

Interface

* Contract to the class. Only contains the methods the class wants the caller to use.

Struct

* Classes can contain default constructors, but structures will contain only constructors that have parameters.
* Unlike classes, structs can be instantiated with or without using a new operator.

Step 2: Reference types vs value types. What is the difference between reference types and value types in C#. Create a code sample showing the difference between a value type and a reference type.

* If passing a value type variable from one method to another method, the system will create a separate copy for the variable in another method. If we make changes to the variable in one method, it won’t affect the variable in another method.
* Value types: int, float, long, char, bool, byte, decimal, double, enum, sbyte, short, struct, uint, ulong, ushort
* When passing a reference type variable from one method to another method, the system won’t create a separate copy for that variable. Instead, it passes the address of the variable, so if we make any changes to the variable in one method, that also reflects in another method.
* Reference types: String, Class, Delegates, Arrays