**C# Language Features**

C# 9:

1. Record Types:  
   Immutable reference types that provide value semantics for equality.
   1. Record types are a compiler feature, So the compiler generates a class based on the record’s template.
   2. Property values cannot be changed after initialization. As the compiler converts properties to read-only fields.
   3. Offers a constructor with all properties as arguments with having to write it.
   4. We can inherit from other record as classes.
   5. Offers a Value-Based Equality as every record type gets an Equals implementation for comparing instances of record types and it compare each property.
   6. Provides an textual representation “implementation” of the ToString method as it allows to transform the property values of a record type properties into a string.
   7. Record types are a compiler feature, So the compiler generates a class based on the record’s template.
   8. Use Cases:
      1. Return data from ASP.NET WebAPI controllers (replace model / viewmodel class).
      2. Data Logging due to the default implementation for TOString method.
      3. Data Comparison.
      4. Immutability, When using the default “implicit” implementation.
2. Top Level Calls:
   1. A feature is useful for running scripts as it is a .cs file to compile and there will be no need for a project.
   2. Everything at the top-level calls is inside a main method.
   3. It is not attached to the class’s name or project’s, and only one compilation unit can have top-level statements.
   4. Offers access to the args parameter.
   5. Offers return the exit code.
   6. Offers definition for structs, classes and records.
   7. Top-Level instructions have to go before types.
3. Initial Setters:
   1. A new keyword used to initialize properties either in the constructor or the object initializer, but can’t be changed later.
4. Patter Matching Improvements:
   1. New keywords: “and or not”, they are similar to “&& || !” but the context which they are used is different.
5. Target-Typed new:
   1. Be able to tie to the statement without specifying the actual type.