C# .net Notes

1. Assembly: C# source code is compiled into an intermediate language. These code and resources are stored in an executable file called an assembly (with an extension of .exe or .dll). An assembly contains a manifest that provides information about the assembly’s types, version, culture, and security requirements. When the C# program is executed, the assembly is loaded into common language runtime.
2. Visual studio, comment code: ctrl K+ ctrl c, uncomment code: ctrl k + ctrl u
3. byte – short – int – long(8-16-32-64 bit integer)
4. float – double -decimal (7 digits fraction-15-28), we use decimal for counted values and use float/double for measured values (https://stackoverflow.com/questions/618535/difference-between-decimal-float-and-double-in-net)
5. enum Day{Sat, Sun, Mon, Tue, Wed, Thu, Fri}
6. define a default constructor for a struct is an error, struct can be instantiated without using new operator. You must initialize all members when write a constructor with parameters. It cannot be inherited.
7. Unmanaged type
8. Abstract class cannot be instantiated; sealed class prevented it from being inherited. Non-abstract class derived from an abstract class must implemented all abstract members. Abstract and interface
9. An abstract method is implicity a virtual method?
10. By default, access modifiers is internal, which can only be accessed in same assembly. Use protected, member can be accessed in a class that is derived from other one.
11. Async: any method using await must be marked async, await means that processing cannot continue until the task finishes. (https://stackoverflow.com/questions/14455293/how-and-when-to-use-async-and-await)
12. Lambda: use lambda expressions that requires instances of delegate types or expression trees. When call enumerable.select, use Func<T,TResult>; when call queryable.select, use tree type Expression<Func<TSource, TResult>>. One parameter, parentheses are optional, otherwise are required. Tuple type as an argument to a lambda expression.
13. Delegate is a reference type that can be used to encapsulate a named or an anonymous method. The delegate must be instantiated with a method or lambda expression that has a compatible return type and input parameters. It is a reference to a method. It used as a parameter to another method.
14. The static is not allowed in a constant declaration. Readonly vs constant, const can only be initialized at the declaration.
15. Readonly can be initialized in the declaration, in the constructor. Readonly struct(struct is immutable), ref readonly(returned reference cannot be modified)
16. Events: publisher(when) and receiver(what action), e.g. button click
17. Extern: combine with DllImport, must be declared as static
18. In: parameter is contravariant.
19. New: hide an inherited member, new vs override: new causes the the original member to become hidden, override extends the implementation for an inherited member.
20. Out: parameter is covariant.
21. Override: extend or modify abstract or virtual, or override implementation of an inherited method.
22. Sealed: prevent other classes from inheriting from it.
23. Static: static member belongs to type itself, not a specific object. Static class means all members of the class must be static.
24. Unsafe:unsafe context
25. Virtual: a method with virtual can be overridden by any class that inherits it. It likes abstract, other it cannot with static.
26. Volatile: a field can be modified by multiple threads that are executing same time.
27. Switch: generic cannot use null case.
28. Generic: maximize code reuse, type safe, performance
29. Struct: is used to encapsulate small groups of related variables, such as the coordinates of a rectangle. Cannot inherit from other struct, so member cannot be protected.
30. Foreach in: ref or ref readonly
31. Out parameter: to be passed by reference
32. Null-coalescing operator ??
33. Exception: don’t initialize variable at try block, otherwise it cannot be accessed at catch block. Use resources in try block, deal with exception in catch block, and release resources in finally block.
34. Checked and unchecked: checked is used as overflow checking for integral type arithmetic operations, if an expression contains one or more non-constant values, the compiler does not detect the overflow. Unchecked means ignore the overflow.