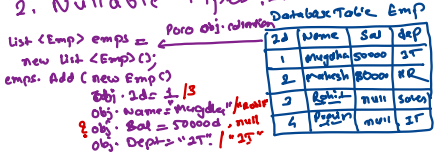
C# Features:-

1. Partial Classes:-

Writing same class (logic) in two different physical files. After compilation → compiler generates single assembly i.e single MSIL.

Using partial keyword over class, helps us to write down logic by different developers using diff. machine parallelly.

2. Nullable Types:-



For e.g:- If we are collecting database records using .NET variables, the value types are not allowed to hold null values by default.

In such scenarios POCO class obj. properties or value types can be assigned "null" by declaring its type as Nullable<value type);

3. Anonymous Methods

→ A method without Name.

→ used for short logic like, compare, data massaging

→ local scope. : you cannot use these methods outside the declared class or anywhere else in assembly.

4. Lambada Expression

→ even shorter syntax than anonymous methods.

→ it uses ⟹ operator : 'goes to' operator

5. Predicate delegate :—

public bool Delegatename<T> (T para)

Predicate<T>

Generic

bool check (int i)

{ return i > 10; }

int

Mydel<int> del = new Mydel<int> (para)

= delegate (int para)

{ return para > 10; }

= para => para > 10

6. Iterator :—

→ Iterator collection. → implementation of IEnumerable.

→ A custom class object behaves like a face of a private member of a class of type collection.

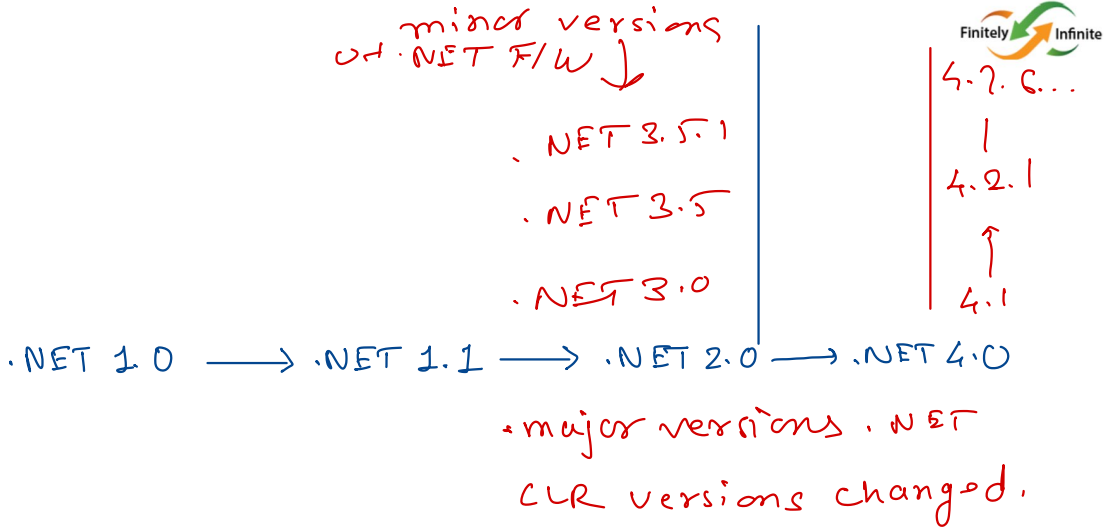
class Weeks → foreach (string day in obj)

↳ has a string [] Days.

and

Weeks obj = new Weeks()

obj → allowing foreach to iterate through private string [] in foreach loop.


minor versions of .NET F/W
.NET 3.5.1
.NET 3.5
.NET 3.0
4.7.6...
4.2.1
4.1
.NET 1.0
.NET 1.1
.NET 2.0
.NET 4.0
major versions .NET
CLR versions changed.

11. Extension method.

:- methods to extend functionalities of existed types. & in built types.

e.g:- string, int[] etc.

- Extension methods are static methods and present in static classes.
- The datatype along with which you need to associate extension method should be passed as a parameter to a method with marked as "this" keyword.

obj. myCollection.ToArray();

prop List<int> MyCollection { get; set; }