

- Why .NET ?

- What is .NET → platform developed by Microsoft.  
enterprise apps, mobile, IoT  
web, desktop apps.

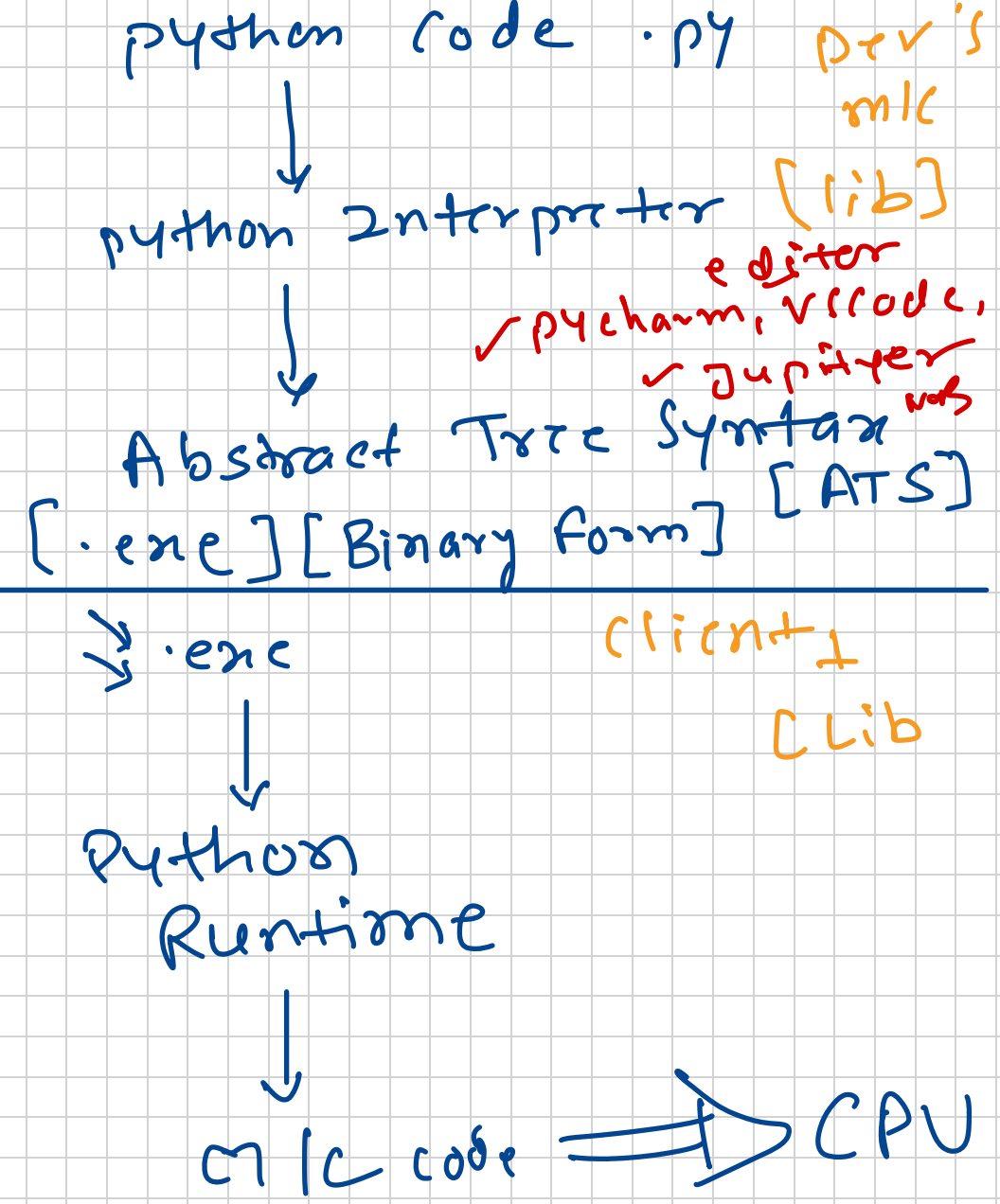
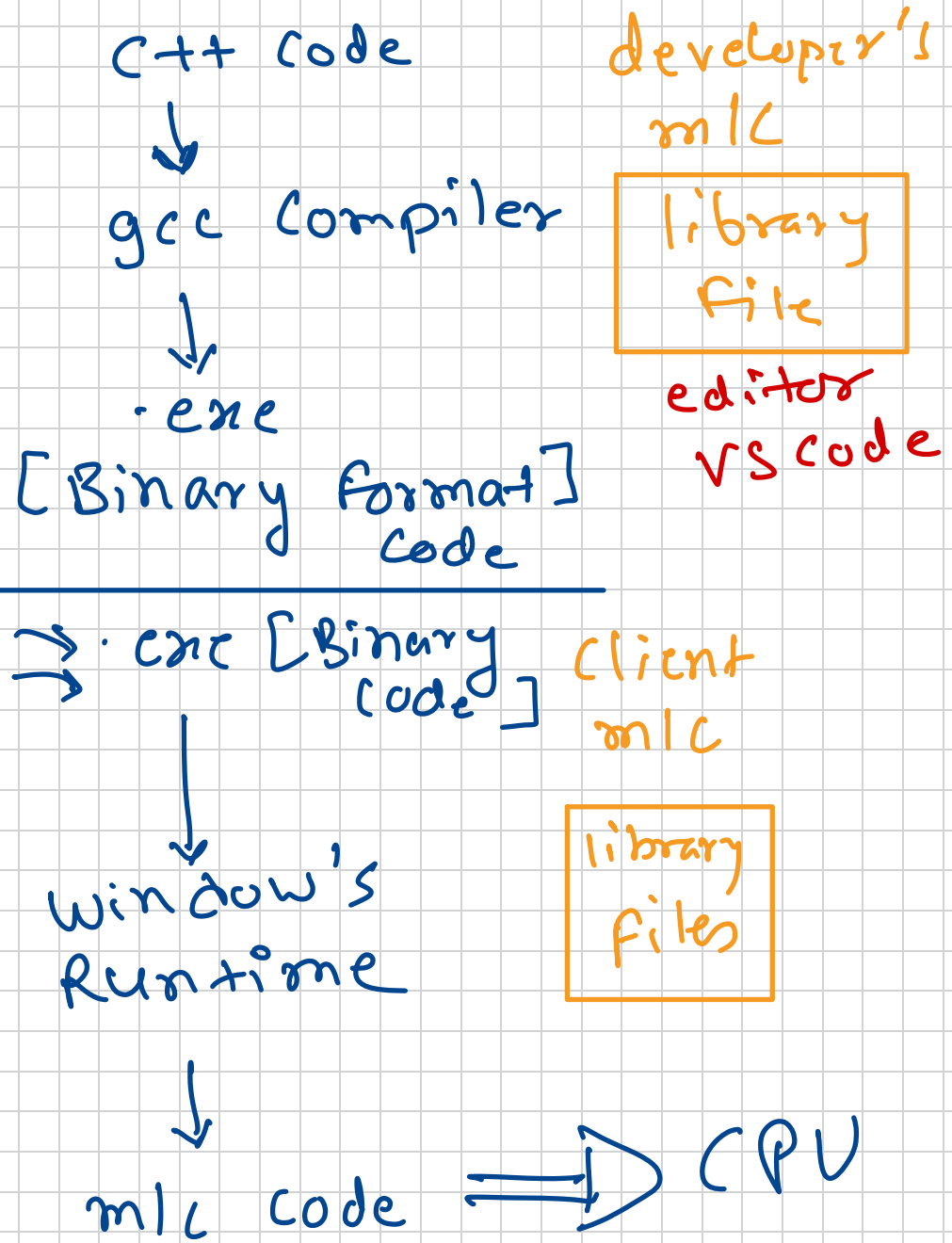
- .NET Framework :-

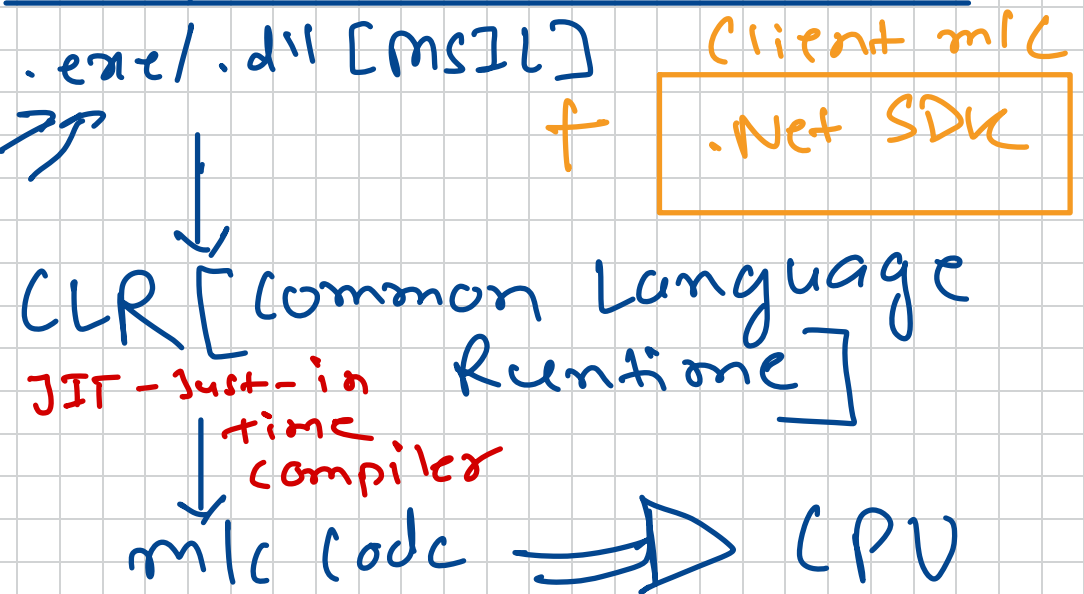
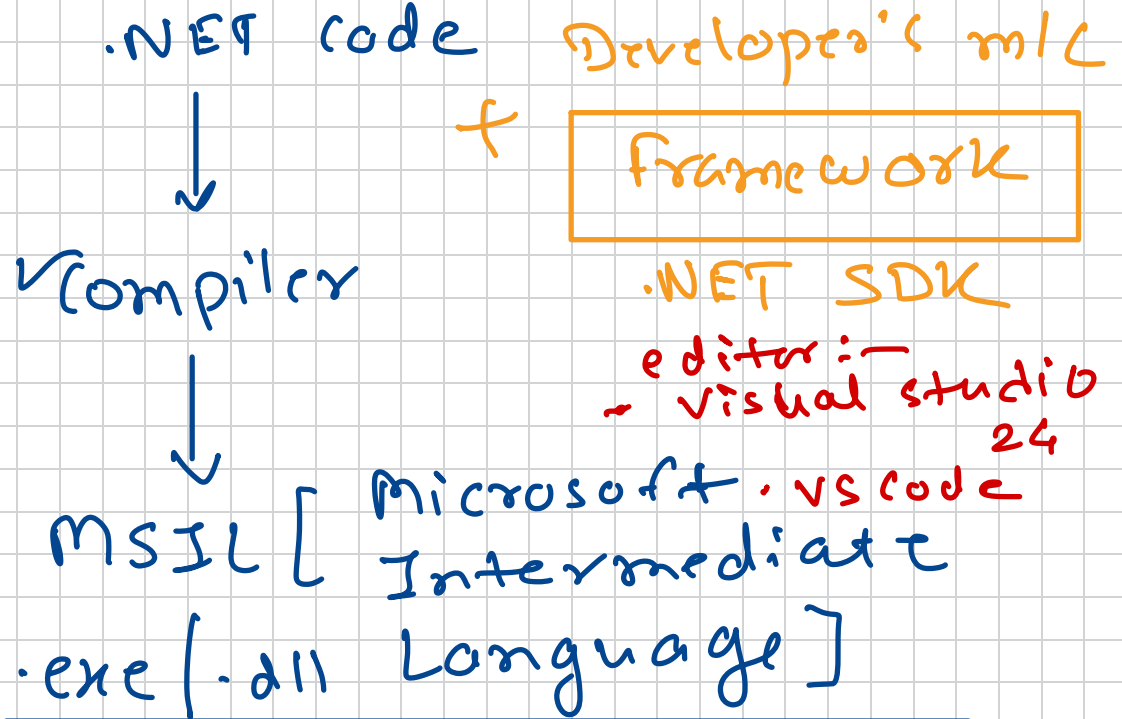
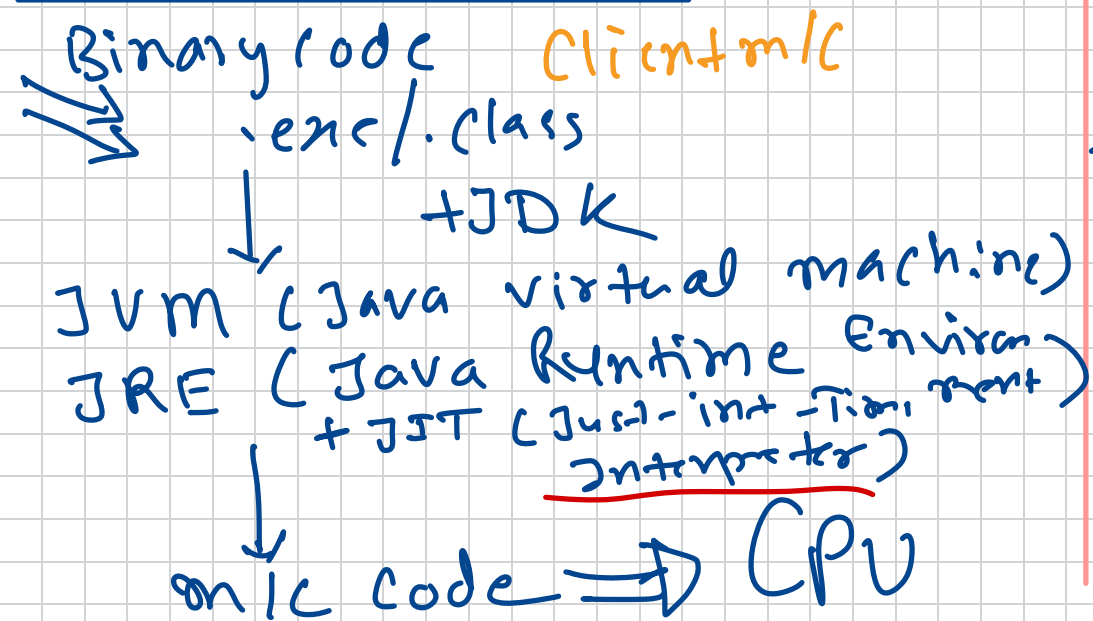
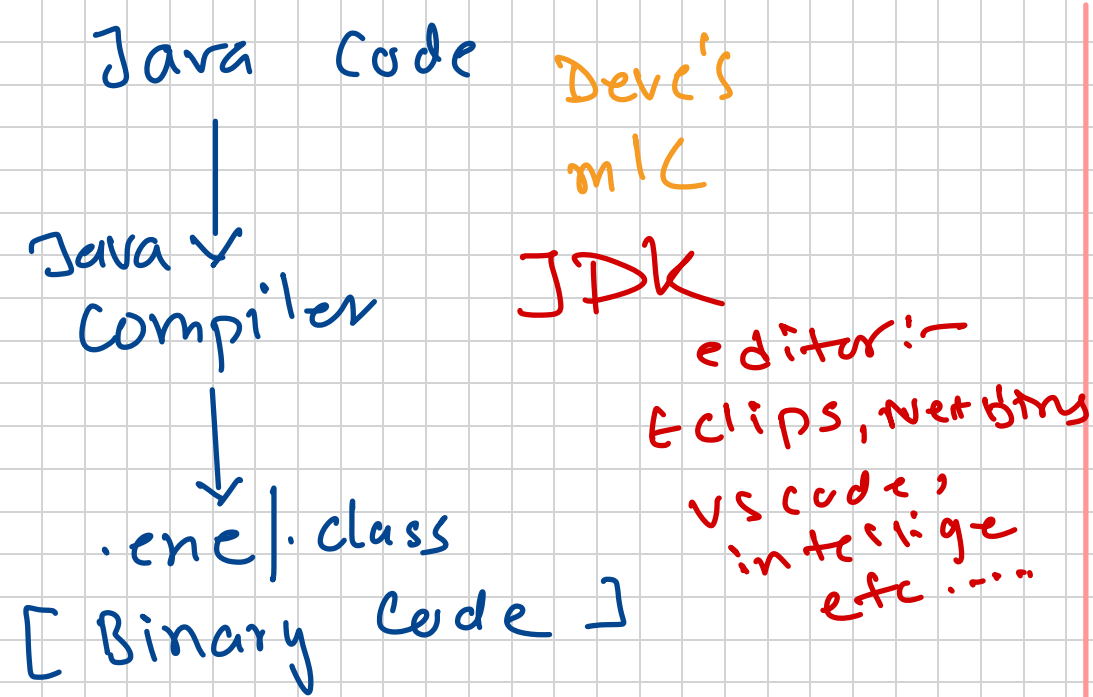
It is a pre-compiled set of library files.  
[ .dll ]

- FCL - Framework Class Library files

- ↳ BCL - Base Class Libraries

e.g:- System.dll    System.Data.dll  
→ set of base required dll files.





• .NET supports multiple languages like:—  
C#, F#, vb.Net, ~~J#~~ 35+ lang.  
supported.

C++

C#



CSC



MSIL

.exe/.dll

Visual Basic

vb.Net



VBC



MSIL

.exe/.dll

F#



F# compiler



MSIL

.exe/.dll

C#

```
int i = 100;
```

```
Person obj = new  
    Person();
```

CSC

int  
double  
string

System.Int32  
System.Double  
System.String

Vb. Net

```
Dim i As Integer
```

```
Dim obj As new  
    Person()
```

CTS  
Common  
Type  
System.

MSIL

# CTS

## Common Type System

C# Code Datatype

short

int

long

double

bool

string

DateTime

→ set of Rules  
how a datatype  
should be presented  
in MSIL

System.Int16

System.Int32

System.Int64

System.Double

System.Boolean

System.String

System.DateTime

CLS :- Common Language Specifications.

- OOP based lang.
- 'new' keyword supported.
- lang. should target Common language Runtime [CLR] for its execution.
- Support JIT-compiler & Garbage collector
- Exception Handling
  - . - . etc . .

JIT - compiler [Just-In-Time compiler]  
 msIL  $\longrightarrow$  native code  
                                 mlc code.

1. Pre-JIT :- All at once.  
Standard F/w library files are pre-jitted.
2. ~~Econo~~-JIT :- when RAM is less  
on demand every time <sup>part of the code</sup> we compile  
the o/p. (native code)  
e.g :- mobile apps.
3. Standard / Normal - JIT :-  
on demand entire code gets converted  
into native code.  
e.g :- Visual Studio 24



Steps about installation:—

- 1) O.S :- Windows
- 2) SDK (.NET FW) ✓
- 3) MS SQL Server ✓ [SQL Express]
- 4) IIS Web Server ✓ [IIS Express / Kestrel Server]
- 5) Visual Studio Editor 2024. ★  
(Community Edition)

# F/W Version

# CLR Version

# C# Version

1.0	1.0	1.0
1.1	1.1	1.2
2.0	2.0	2.0
3.0	2.0	2.0
3.5	2.0	3.0
4.0	4.0	4.0
4.5	4.0	5.0
4.5.1	4.0	6.0
4.6 / 4.6.1 / 4.6.2	4.0	7.0
4.7 / 4.7.1 ...	4.0	7.x ...
4.8		

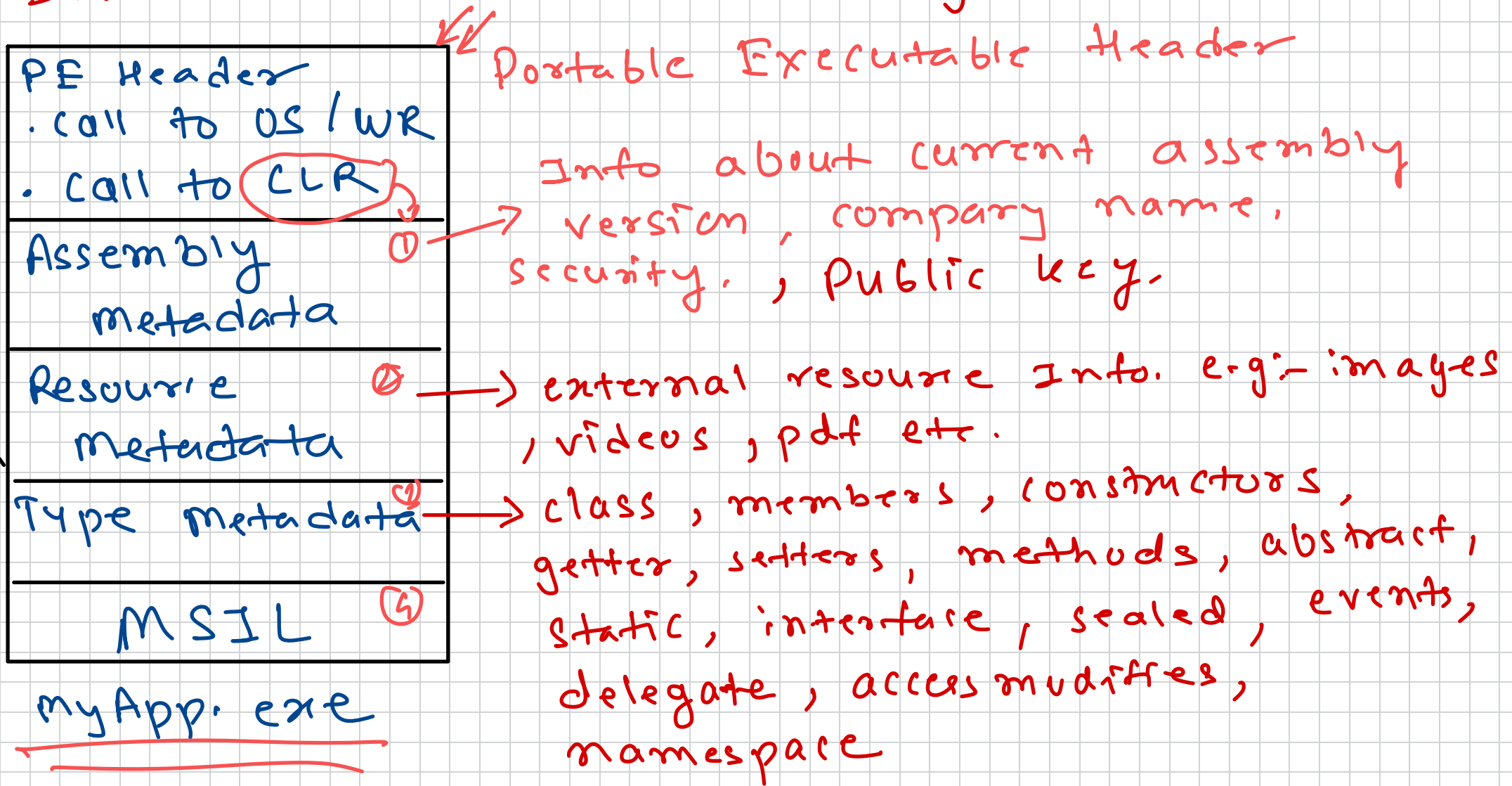
# Common Language Runtime (CLR)

Features:-

mscorlib.dll  $\rightarrow$  C++ code.

1. Assembly Execution
2. Memory allocation for 'managed code'
3. for memory de-allocation, CLR invokes its helper module - GC [Garbage collector]
4. Exception Handling.
5. Interoperability / Com Interop
6. Security.

# Structure of .NET Assembly :-



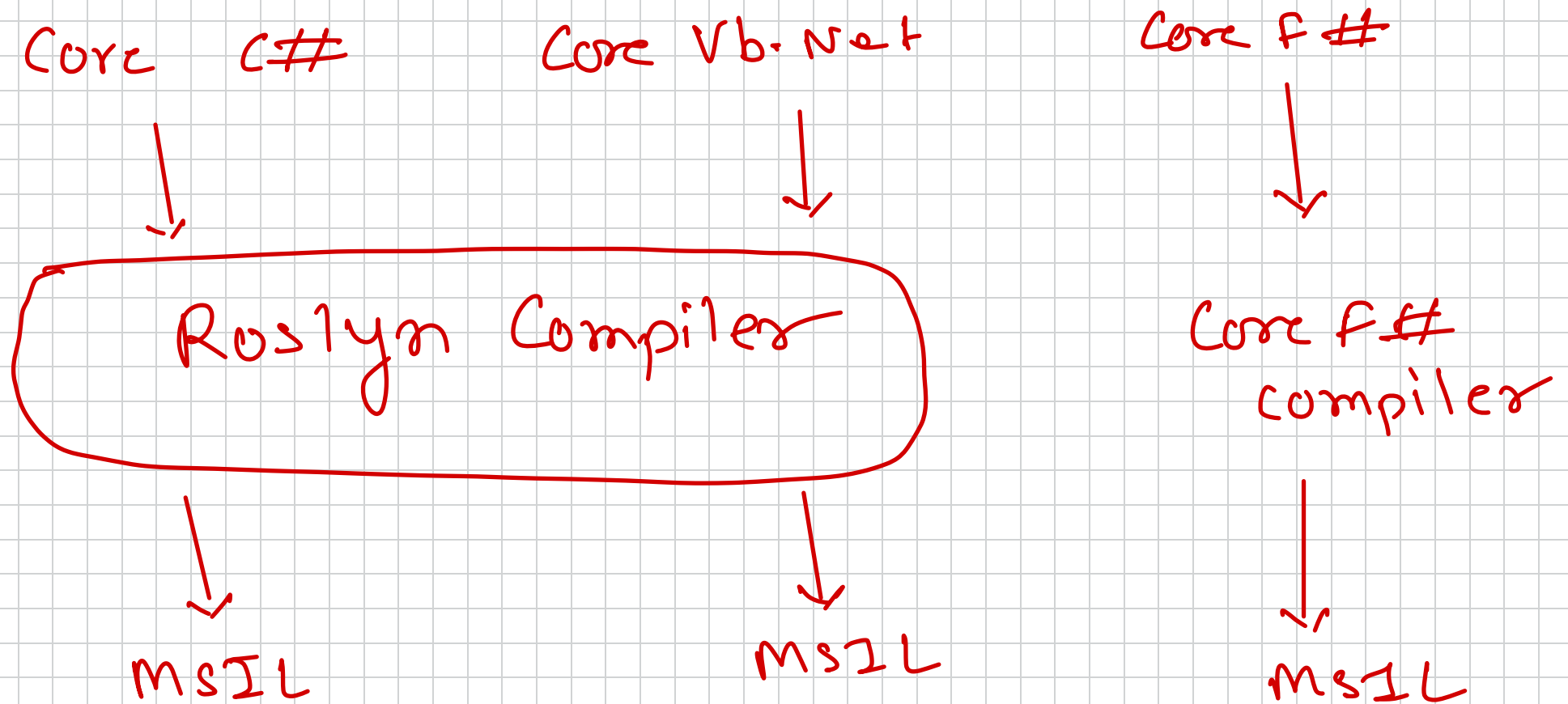
- dll Assemblies does not contain PE header section.

• .NET Core :-

- open-source, cross-platform framework
- works on windows, Linux, macOS
- web application, mobile, web-services, cloud based application, console based, Command line Interface (CLI) application

• Core CLR :- System.Private.CoreCLR.dll

• Output :- .exe + .dll



### ✓ Roslyn Compiler!:-

- fully managed, open-source compiler for C# & Vb.Net,
- provides not just compilation; also rich code analysis APIs, generates code, inspect

.Net core lang. (C# & vb.Net)



Roslyn compiler



MSIL

---

MSIL



Core CLR

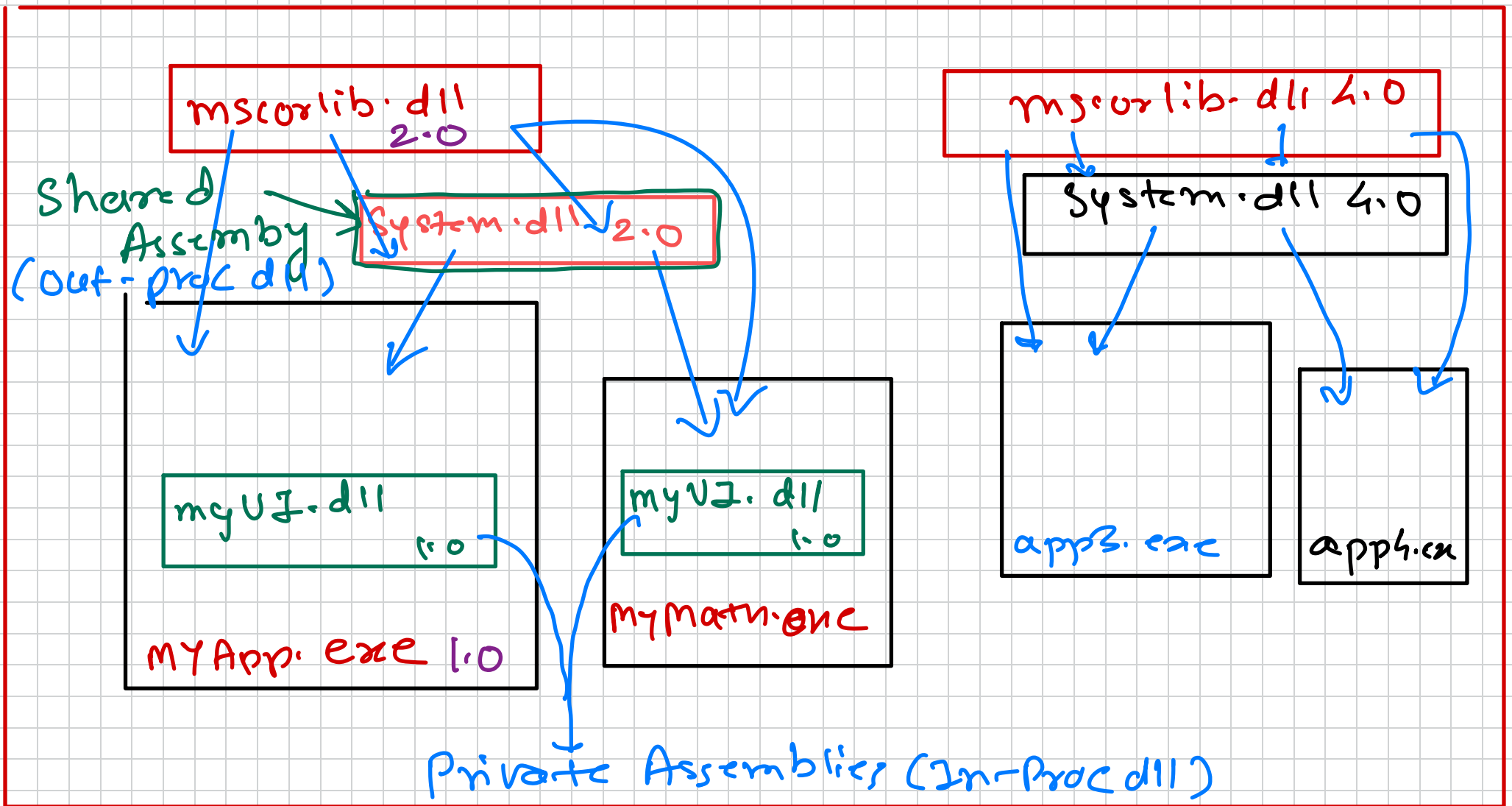
+

RyuJIT



native code  $\Rightarrow$  CPU

OS





• .Net Assemblies Types:-

1. Shared assembly:-

GAC :- Global Assembly Cache.  
[.NET Registry]

\* all framework library files gets register into GAC.