Day 17

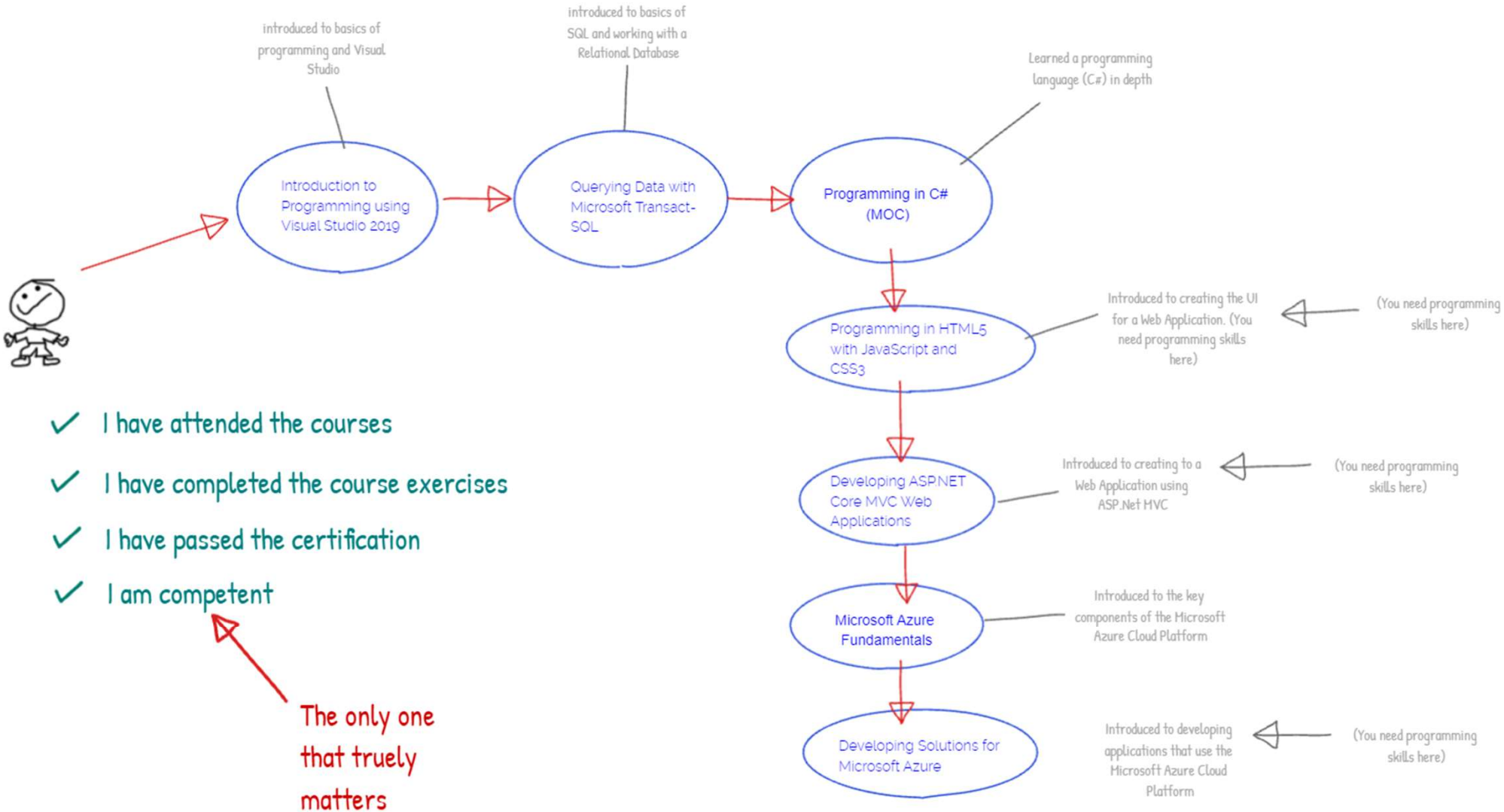
The one link you need to remember

https://ddls.to/20483

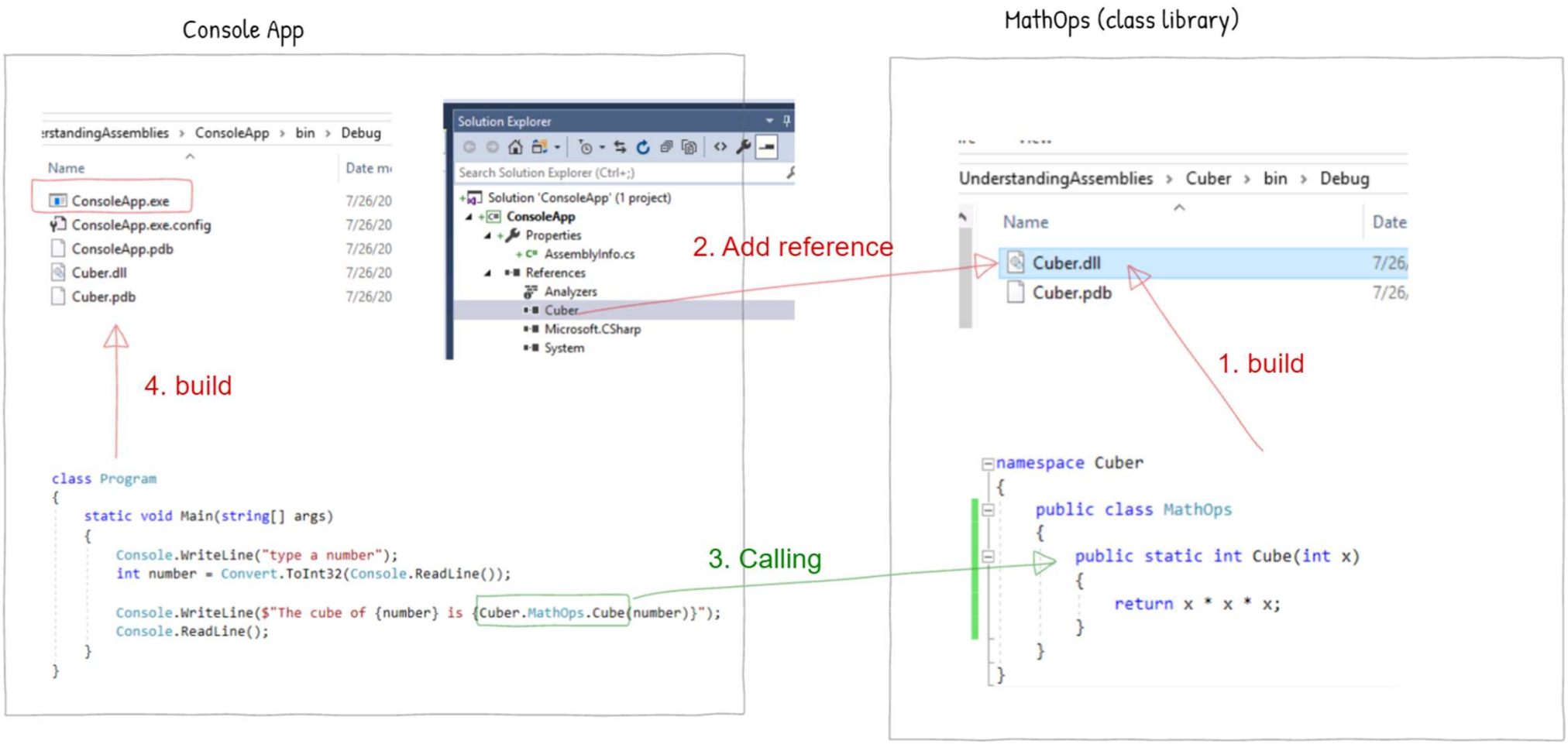
Ready?

Do this every day BEFORE the class starts (takes about 15 minutes) (http://ddls.to/everyday)

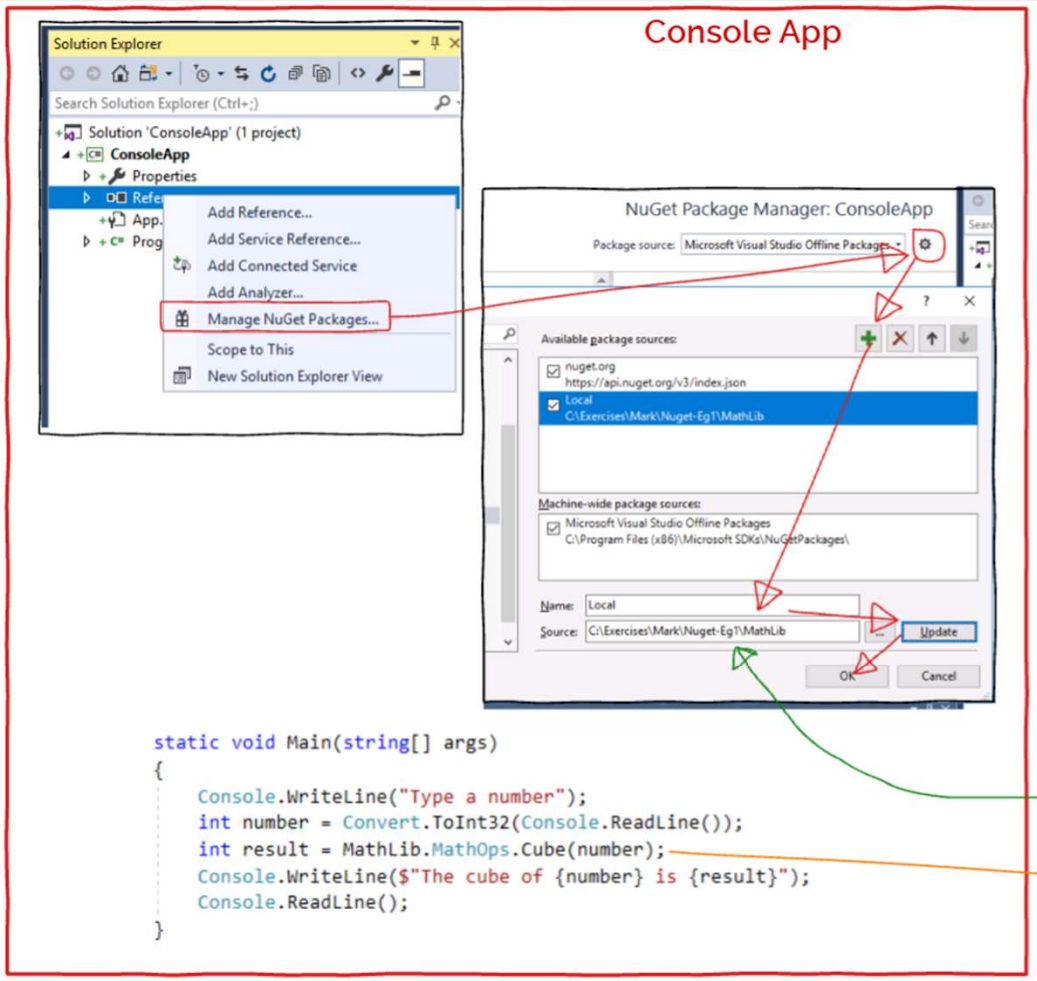
1. Launch Lab01.
2. Login to Lab01 as **Admin**.
3. While in the Lab01 environment,
   i. run **cmd.exe** from the Windows Start button.
   ii. Run the command **git clone --depth 1** https://github.com/Mark-AIICT/CAD-2.git **C:\Users\Admin\Desktop\MarksFiles**
   iii. Navigate to **C:\Users\Admin\Desktop\MarksFiles\setups**, then right-mouse click **bootstrap.cmd** and **run as administrator**
   iv. While it's running, Sign in to Visual Studio on the Lab Environment. You can use any Microsoft account.
   v. When the script end it reboots the Virtual Machine. That's necessary.
   vi. Save the lab. (the save link is at the top right of the screen in the dropdown menu)

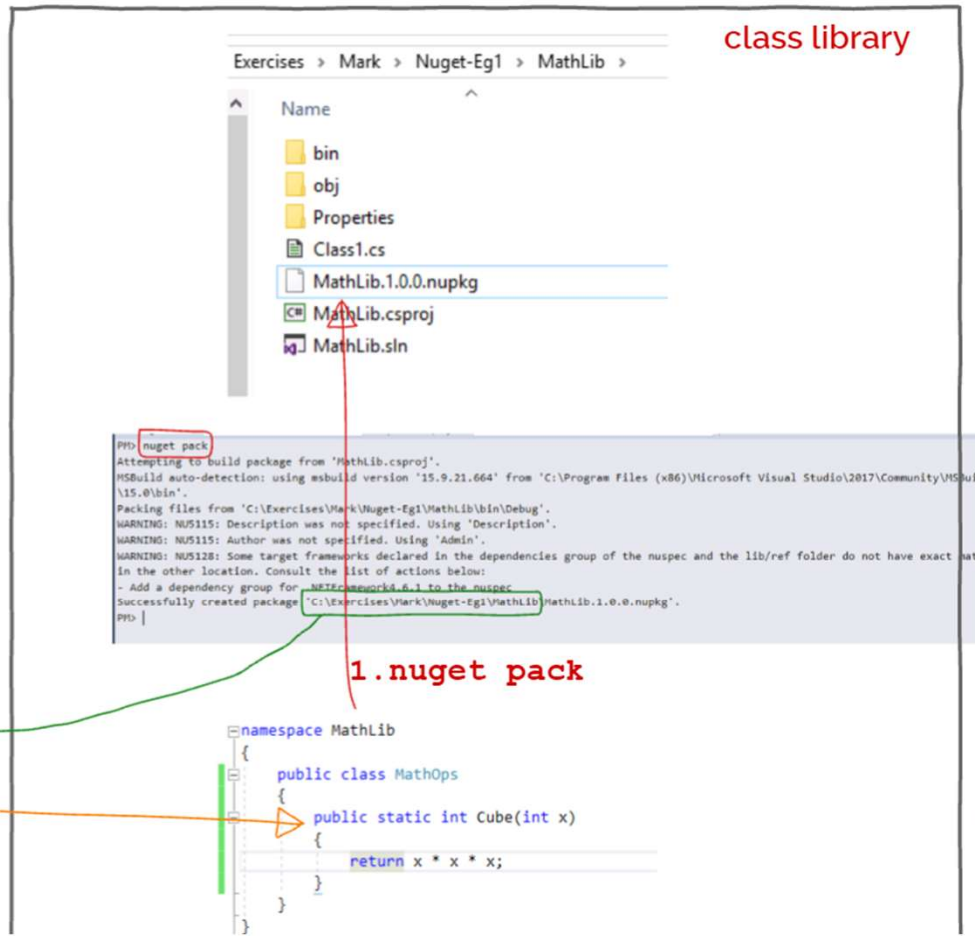

introduced to basics of programming and Visual Studio
introduced to basics of SQL and working with a Relational Database
Learned a programming language (C#) in depth
Introduction to Programming using Visual Studio 2019
Querying Data with Microsoft Transact-SQL
Programming in C# (MOC)
Programming in HTML5 with JavaScript and CSS3
Introduced to creating the UI for a Web Application. (You need programming skills here)
(You need programming skills here)
Developing ASP.NET Core MVC Web Applications
Introduced to creating to a Web Application using ASP.Net MVC
(You need programming skills here)
Microsoft Azure Fundamentals
Introduced to the key components of the Microsoft Azure Cloud Platform
Developing Solutions for Microsoft Azure
Introduced to developing applications that use the Microsoft Azure Cloud Platform
(You need programming skills here)
✓ I have attended the courses
✓ I have completed the course exercises
✓ I have passed the certification
✓ I am competent
The only one that truely matters

Referencing DLL's

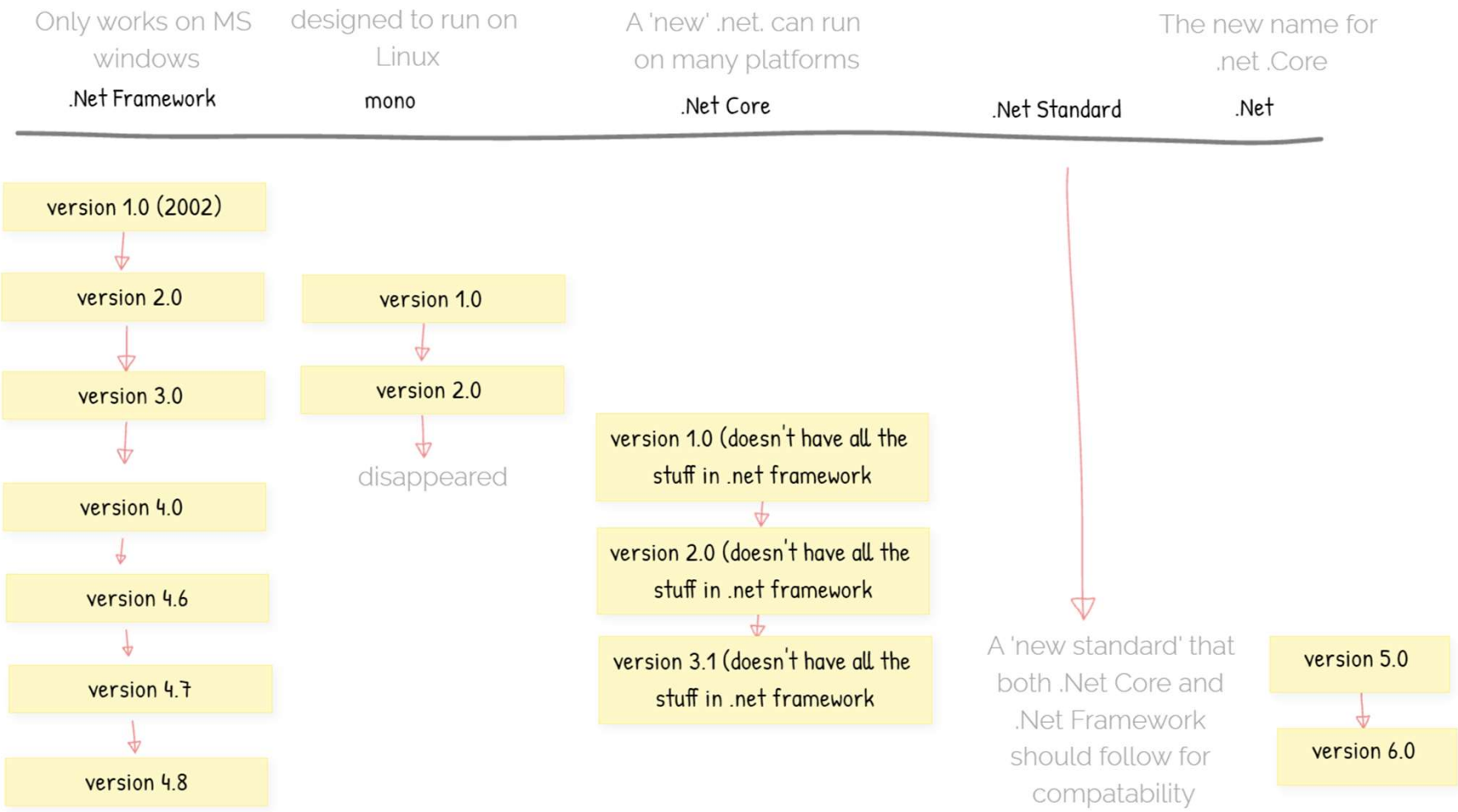


Nuget Packages





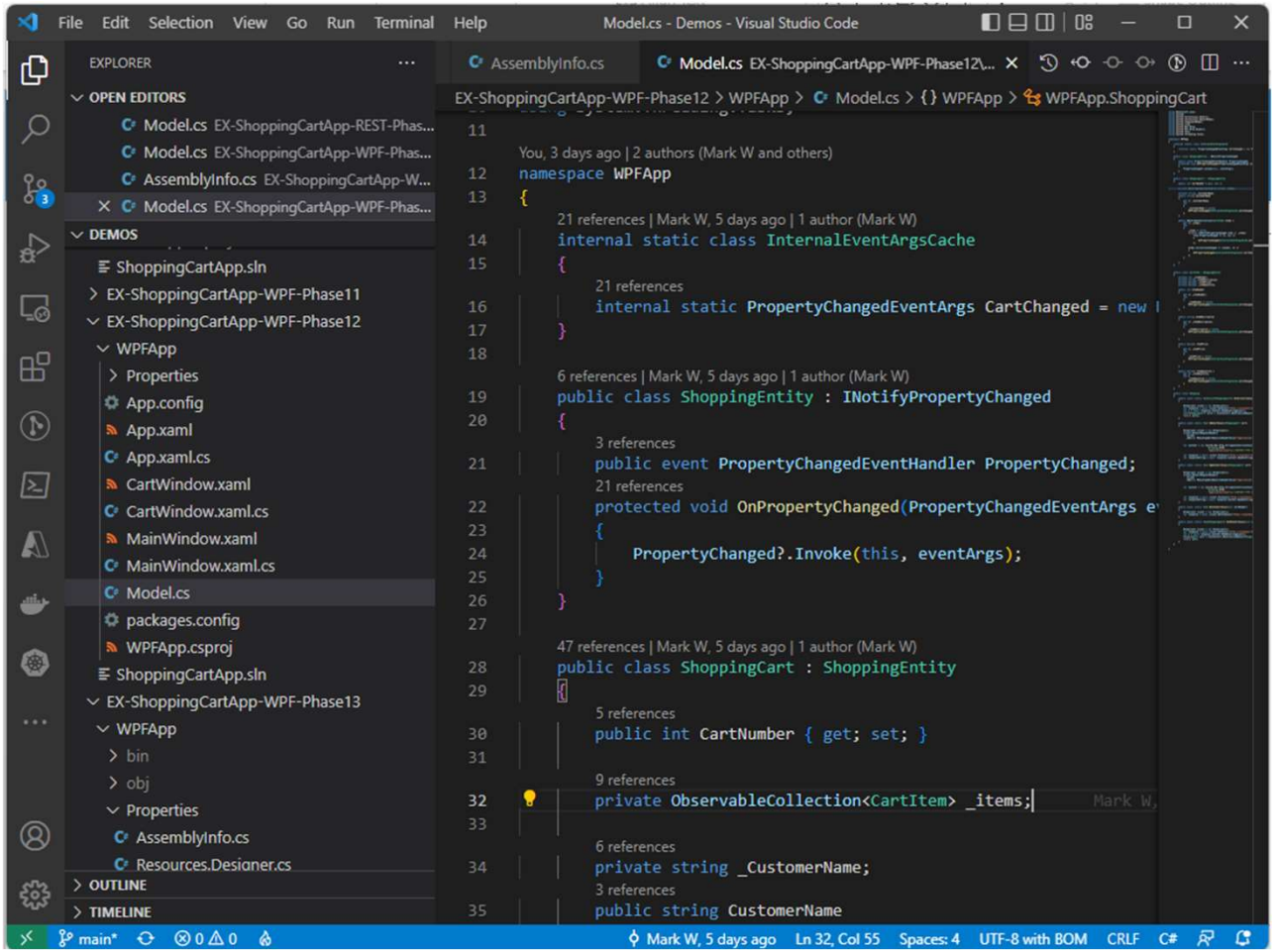
There is more than one '.net'



.Net Core (.Net)

- dotnet new
- dotnet build
- dotnet run

VS Code


File Edit Selection View Go Run Terminal Help
Model.cs - Demos - Visual Studio Code
EXPLORER
OPEN EDITORS
Model.cs EX-ShoppingCartApp-REST-Phas...
Model.cs EX-ShoppingCartApp-WPF-Phas...
AssemblyInfo.cs EX-ShoppingCartApp-W...
Model.cs EX-ShoppingCartApp-WPF-Phas...
DEMOS
ShoppingCartApp.sln
EX-ShoppingCartApp-WPF-Phase11
EX-ShoppingCartApp-WPF-Phase12
WPFApp
Properties
App.config
App.xaml
App.xaml.cs
CartWindow.xaml
CartWindow.xaml.cs
MainWindow.xaml
MainWindow.xaml.cs
Model.cs
packages.config
WPFApp.csproj
ShoppingCartApp.sln
EX-ShoppingCartApp-WPF-Phase13
WPFApp
bin
obj
Properties
AssemblyInfo.cs
Resources.Designer.cs
OUTLINE
TIMELINE
AssemblyInfo.cs
Model.cs EX-ShoppingCartApp-WPF-Phase12\...
EX-ShoppingCartApp-WPF-Phase12 > WPFApp > Model.cs > { } WPFApp > WPFApp.ShoppingCart
You, 3 days ago | 2 authors (Mark W and others)
namespace WPFApp
{
21 references | Mark W, 5 days ago | 1 author (Mark W)
internal static class InternalEventArgsCache
{
21 references
internal static PropertyChangedEventArgs CartChanged = new
}
6 references | Mark W, 5 days ago | 1 author (Mark W)
public class ShoppingEntity : INotifyPropertyChanged
{
3 references
public event PropertyChangedEventHandler PropertyChanged;
21 references
protected void OnPropertyChanged(PropertyChangedEventArgs e
{
PropertyChanged?.Invoke(this, eventArgs);
}
}
47 references | Mark W, 5 days ago | 1 author (Mark W)
public class ShoppingCart : ShoppingEntity
{
5 references
public int CartNumber { get; set; }
9 references
private ObservableCollection<CartItem> _items;
6 references
private string _CustomerName;
3 references
public string CustomerName
main*
0 0
Mark W, 5 days ago
Ln 32, Col 55
Spaces: 4
UTF-8 with BOM
CRLF
C#

Course Outline

- Module 1: Review of Visual C# Syntax
- Module 2: Creating Methods, Handling Exceptions, and Monitoring Applications
- Module 3: Basic Types and Constructs of Visual C#
- Module 4: Creating Classes and Implementing Type-Safe Collections
- Module 5: Creating a Class Hierarchy by Using Inheritance
- Module 6: Reading and Writing Local Data
- Module 7: Accessing a Database
- Module 8: Accessing Remote Data (I'm replacing this with a better module)
- Module 9: Designing the User Interface for a Graphical Application
- Module 10: Improving Application Performance and Responsiveness
- GIT
- REST + WPF + Sharpen C# Skills
- Config Files
- .Net Framework Vs .Net Core Vs .net standard Vs .Net
- VSCode Vs Visual Studio
- Nuget
- Module 11: Integrating with Unmanaged Code?
- Module 12: Creating Reusable Types and Assemblies?
- Module 13: Encrypting and Decrypting Data?

```
- App.Settings Config

Assemblies
    - Assembly metadata
    - dll Vs exe
    - referencing


.Net History


Dotnet core with VS2017
    - Simple Console Application (doesn't work with VS2017 because VS2017 only supports Version <3)...Could update VS2017 -> takes X minutes.

VSCode
    - dotnet new
    - dotnet build
    - dotnet run
    - using VSCode
    - extensions
    - creating a solution with multiple projects

nuget (.net Framework)
    - nuget pack
    - Package contents
    - nuget spec
    - Accessing packages
    - NuGet.Org


nuget (.net (core))
    - nuget pack
    - Package contents
    - nuget spec
    - Accessing packages
    - NuGet.Org
```

- To do on your lab01 VM
  - Set git global config

    ```
    git config --global user.name “fred”
    git config --global user.email fred@work.com
    ```

  - Clone the exercises repo