

## About this Course

من اللغات القويه وتقدر تعمل بيها تطبيقات سواء ويب او desktop او موبايل وتعتبر portable language يعني تقدر تشغل التطبيقات بقاعدتك على أي سيسنتم وهيا لغه قويه ومطلوبه

لو ماشي معاه في ال 13 كورس اللي فاتوا الكورس ده هيكون سهل بالنسبة لك  
هوا مش هيعد شرح المفاهيم اللي شرحها قبل كده  
في البدايه هتعلم ال syntax وتعليمها مش هيأخذ منك اكتر من 3 أيام

بعد كده هيأخذ الحاجات الجديده اللي موجوده في C# وبعدها ال windows forms وال graphical desktop application interface ونعمل forms ونعرف عالادوات بقاعدتك ال C# في بناء ال front end و back end قوي وبعدين تقدر تشفو أي حاجه تانيه سواء ويب او موبايل

هناخدتها ضمن اطار ال .net framework. لانه بيوفر بيئه عمل لكل لغات البرمجه تحت مسمى managed code يعني بتقدر تستخدم اكتر من لغة عشان تستخدم برنامجه

The advertisement features a large image of a laptop displaying the course title 'Programming In C#' and 'Level 1'. A man in a suit stands behind the laptop. The website 'ProgrammingAdvices.com' is visible at the bottom right. On the left, there's a red tag labeled 'Price 20\$' and another tag showing '90% OFF'. The background shows a stack of books and papers.

**ProgrammingAdvices.com**

**مميزات الكورس**

في هذا الكورس سنبدأ بتعلم C# لأنها من أقوى لغات البرمجة المطلوبة في سوق العمل فهي تمكّنك من إنشاء مختلف أنواع التطبيقات، Desktop, Mobile, Web and others المستوى ستتعلم قواعد اللغة وستنتقل للتعرّف على Windows Forms و Desktop Development والتعرّف على مفاهيم واساليب برمجية جديدة ستكون اساس قوي لك للتقدم في عالم البرمجة بقوّة.

## يجب ان تكون انهيت جميع الكورسات من 1 الى 13

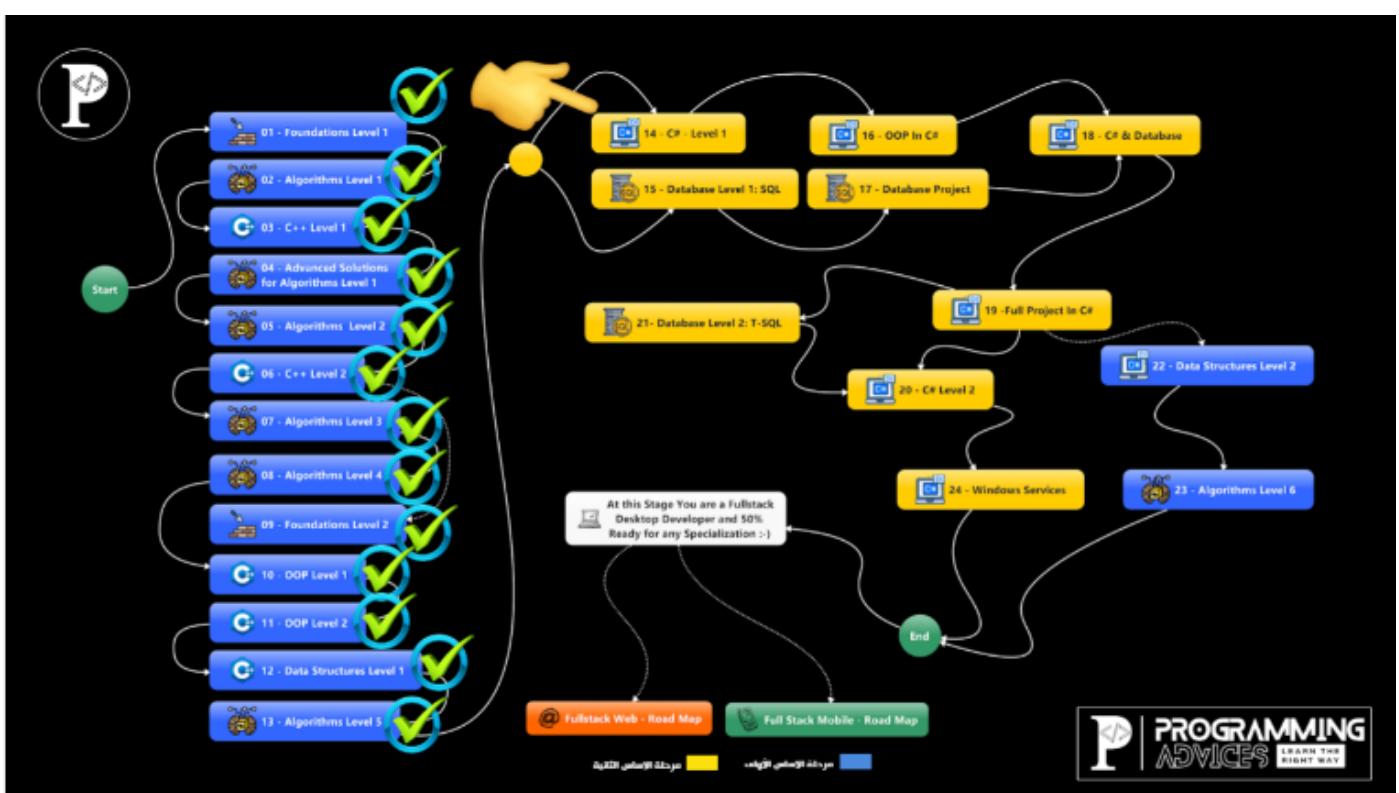
في الكورسات السابقة نحن تعلمها البرمجة وبنينا اساس جيد جدا فيها فهي تعتبر حجر اساس للمضي في هذا الكورس

كل من تخطى اي كورس من الكورسات السابقة سيواجه صعوبة في هذا الكورس

ومن انهى الكورسات السابقة فسيجد هذا الكورس قمه في السهولة والمنعنة

نحن لن نعيد شرح المفاهيم والاساسيات البرمجية التي تم شرحها في الكورسات السابقة بل سنبني عليها ونكمي المسيرة

كل التوفيق للجميع



**Telegram Group**

<https://t.me/+BkTQUwvUZcQxNDU0>

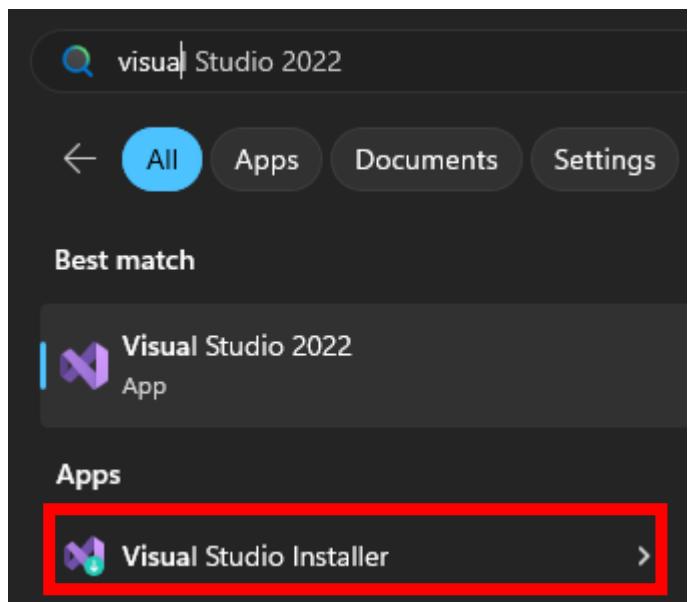
**Installing C#**

دلوقي هنعرف ازاي نحمل ال C#

هنشتغل عال visual studio community edition

بتفتح قائمة start وبعد بتبث عن visual studio

هتللاقي visual studio installer



# Visual Studio Installer

Installed Available

Checking for updates...

**Visual Studio Community 2022**  
17.7.0  
Powerful IDE, free for students, open-source contributors, and individuals  
[Release notes](#)

Modify  
Launch  
More ▾

This screenshot shows the "Available" tab of the Visual Studio Installer. It lists the "Visual Studio Community 2022" package, version 17.7.0. The package description states it's a "Powerful IDE, free for students, open-source contributors, and individuals". Below the description are links for "Release notes", "Modify", "Launch", and "More".

Modifying — Visual Studio Community 2022 — 17.7.0

Workloads Individual components Language packs Installation locations

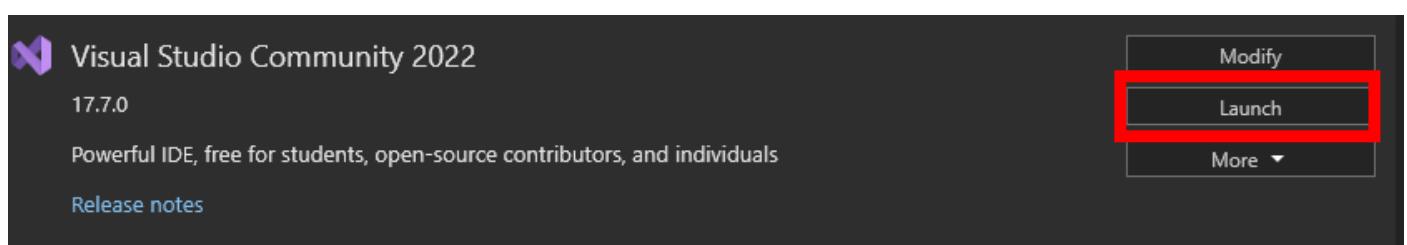
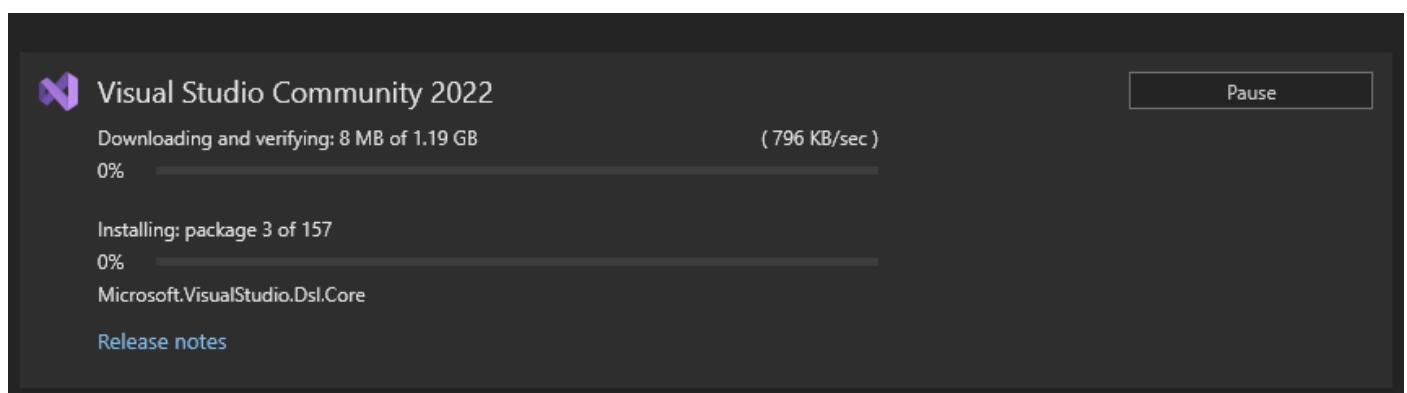
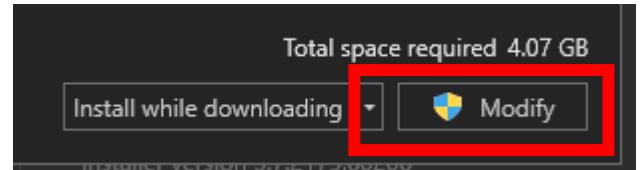
### Web & Cloud (4)

- ASP.NET and web development**  
Build web applications using ASP.NET Core, ASP.NET, HTML/JavaScript, and Containers including Docker supp...
- Azure development**  
Azure SDKs, tools, and projects for developing cloud apps and creating resources using .NET and .NET Framework...
- Python development**  
Editing, debugging, interactive development and source control for Python.
- Node.js development**  
Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.

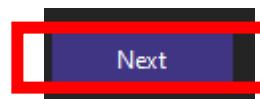
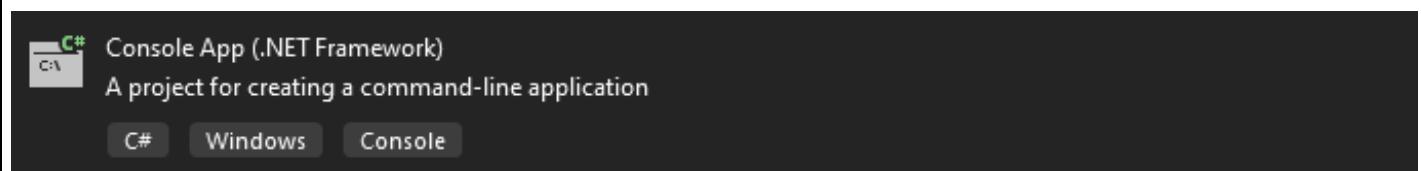
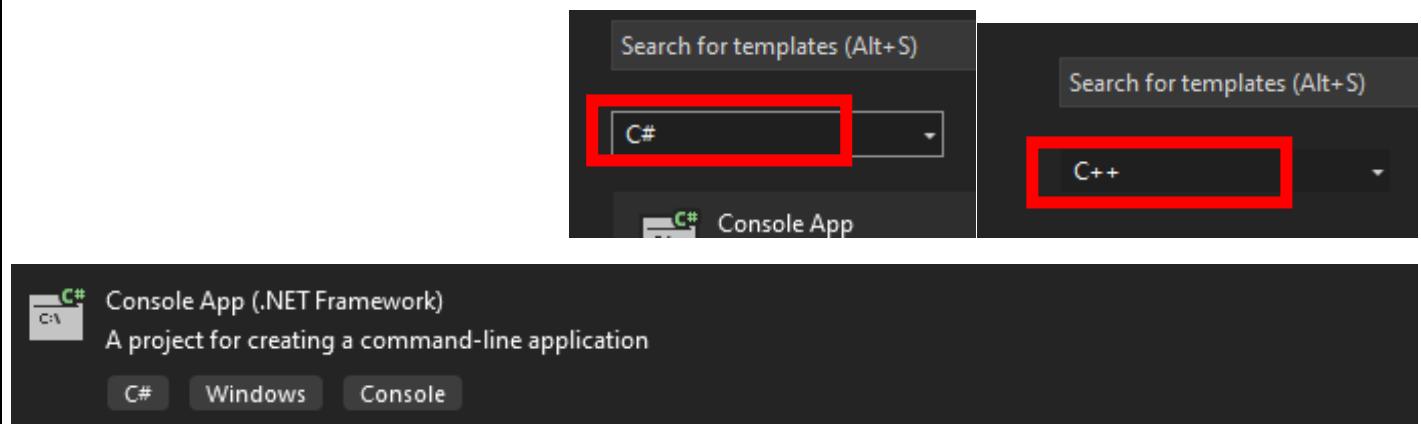
### Desktop & Mobile (5)

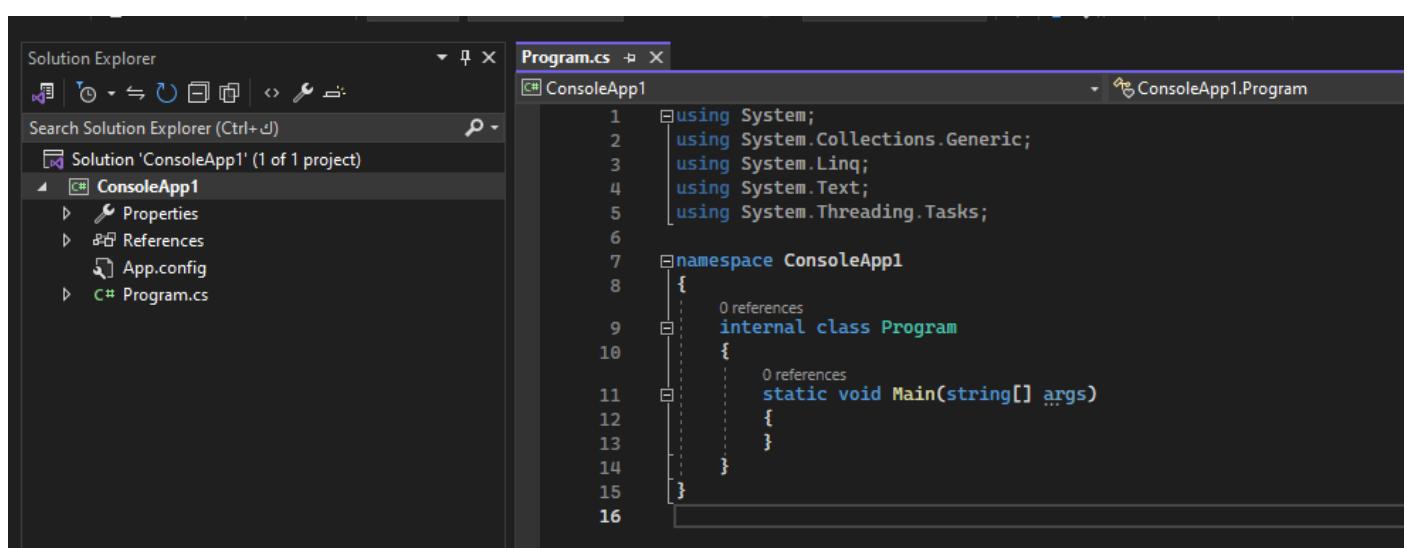
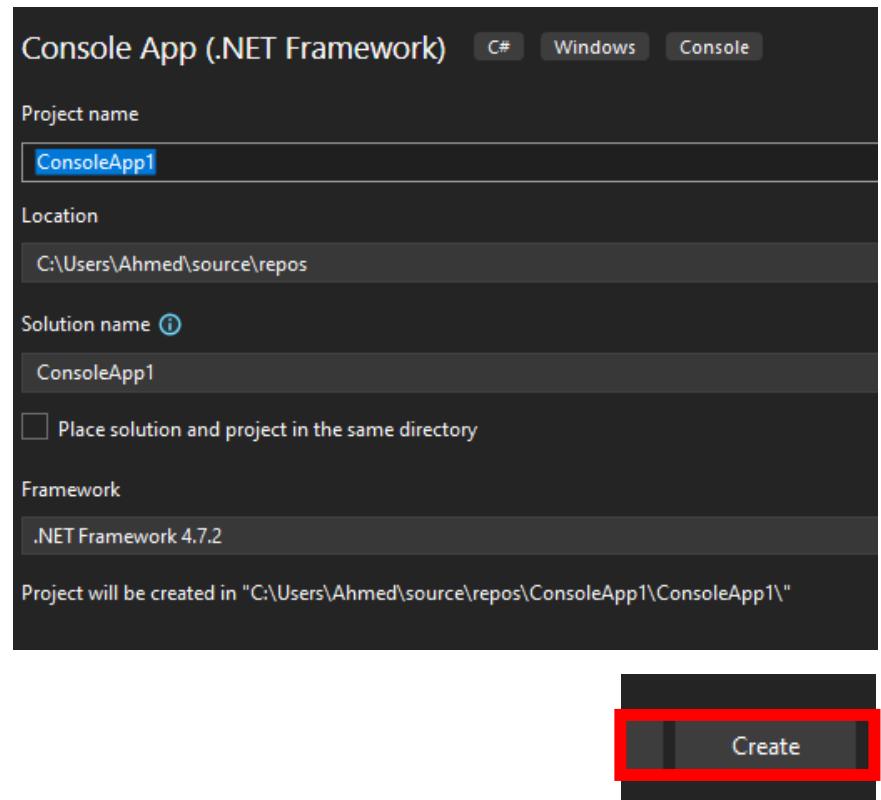
- .NET Multi-platform App UI development**  
Build Android, iOS, Windows, and Mac apps from a single codebase using C# with .NET MAUI.
- .NET desktop development**  
Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET and .NET Frame...
- Desktop development with C++**  
Build modern C++ apps for Windows using tools of your choice: CMake, MSBuild, CMake, CMake, CMake, CMake, C...
- Universal Windows Platform development**  
Create applications for the Universal Windows Platform

This screenshot shows the "Workloads" tab of the Visual Studio Installer. It lists five categories: "Web & Cloud", "Desktop & Mobile", "Individual components", "Language packs", and "Installation locations". Under "Web & Cloud", there are four items: "ASP.NET and web development", "Azure development", "Python development", and "Node.js development". Under "Desktop & Mobile", there are five items: ".NET Multi-platform App UI development", ".NET desktop development", "Desktop development with C++" (which is checked and highlighted with a red box), and "Universal Windows Platform development".



بعدين عشان تتأكد افتح مشروع جديد و بتغير الفلتر لـ C# وبعدها تختار

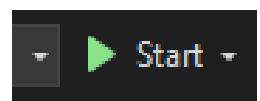




هناخد الشاشه السوداء تاني لمدة 3 أيام بعدين ننقل للعالم الملون  
وهنا بنجرب جملة طباعة

```
static void Main(string[] args)
{
    Console.WriteLine("hi this is my first C# app :-)");
}
```

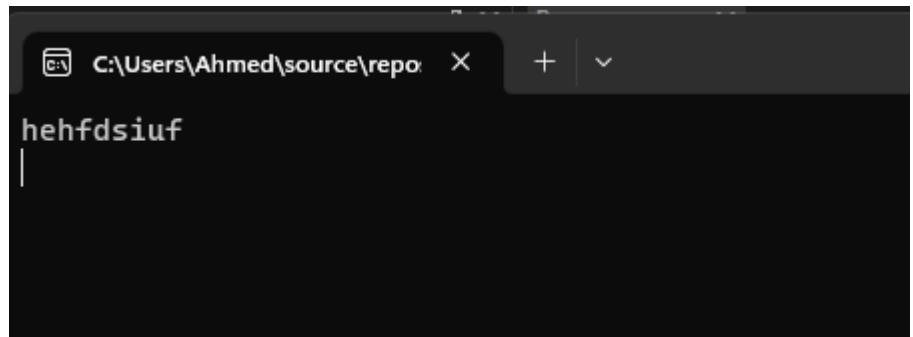
وبتشغله من هنا



هتشغله هتلaci الشاشه طلعت وقفلت تفتكـر انه ماشتغلش لا هوا بعد م اخلص قـل الشاشه  
عشان تعمل زي pause اللي كـنا بنعملها في ال C++ بتكتب كـده

```
Console.ReadLine();
```

طبعا انا كاتب اي كلام



Incase you are not upgrading your current visual studio , you can download it from here:

<https://visualstudio.microsoft.com/vs/community/>

في حال اذا لم يكن لديك فيجوال ستديو 2022 يمكنك تحميله من الرابط اعلاه

<https://visualstudio.microsoft.com/vs/community/>

## Live Meeting (Questions and Answers) Feb 17, 2023

Live Meeting (Questions & Answers)

Mon , Feb 27 , 2023

### **Library vs Framework vs Platform**

هنعرف النهارده الفرق بين ال library وال frame work وال platform

ال library مجموعه من ال functions اللي مرتبـه ببعضها او لوحـدها وبتمكـنك من إعادة استخدامها  
بدون ماتعيد كتابـتهم

ال frame work هو مجموعـه من المكتـبات والقوانين اللي بتحكمـك في طريـقة كتابـة الكـود وبيكونـ هوـا  
ال environment اللي بتشغـل ال software عـليـها وال environment دي يـاما بتكونـ  
software او hardware

ال frame work بينظم طريـقة عمل المـبرمجـين عـشـان مايـجيـش كل واحد يـكتبـ علىـ كـيفـه

ال platform هو أوسع من ال framework كده انه يسمح لك انك تشغله  
البرامـج

وال platform ده ممكن يتكون من لغة برمجه ومكتبات

## What is Library?

A library is a collection of pre-written code that developers can use to add functionality to their applications.

A library typically provides a set of functions or classes that a developer can call from their code to perform specific tasks, without having to write all the code from scratch.

Example: Math Library.

## What is Framework?

A framework is a set of libraries and tools that provide structure and support for building and running applications.

A framework defines a set of rules, protocols, and conventions that developers must follow when writing their code. This helps ensure that all parts of an application work well together and follow a consistent design pattern.

## What is Platform?

A platform refers to the hardware or software environment in which a piece of software runs.

For example, a computer running the Windows operating system could be considered a platform for running software written for Windows

Platform = Programming Language + Libraries

### What is .NET? .NET Core vs .NET Framework? What can we do with .NET?

ال platform .net هي عباره عن بتمننك انك تشغله البرامج بتاعتك اللي مكتوبه بلغات مدعومه من ال .net. اللي بيدعم اكتر من 70 لغه برمجه

كل اللغات دي بتسخدم نفس المكتبات وكل اللغات الموجوده في ال .net. هيا أوجه لعمله واحده يعني لو استخدمت مثلاً ال visual basic .net بتابعها هيكون زيه زي ال c# لأنهم بيستخدموا نفس المكتبات

ال platform فيه اكتر من component وفيه 3 أشياء :-

1 - .net core : وده اتعمل بعد ال netframe work . والفرق بينهم انه ال .net core عباره عن cross platform التطبيق بتابعه بيشتغل علي windows و linux و mac os و windows . بيشتغل عال windows بس انم ال .net fram work

2 - .netframe work : ده اول حاجه نزلت واتعملت عليه مشاريع كبيره .  
3 - Xamarin / mono : بتقدر تعمل منه تطبيق موبايل لأندرويد وال ios

## What is .NET?

Platform = Programming Language + Libraries

.Net is a Platform ☺

Languages in .Net Platform are:

- C#, Visual Basic, and F#...etc.

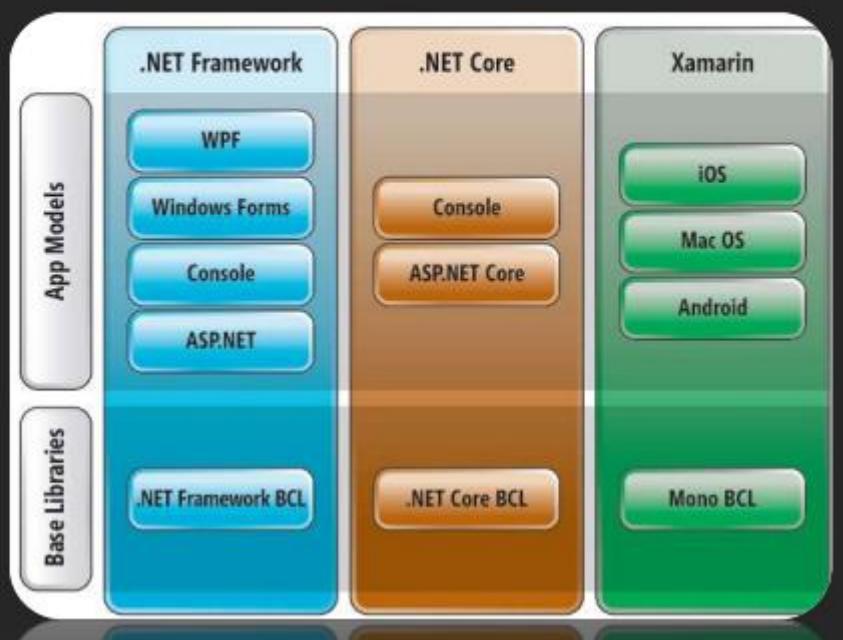
Platforms:

- .Net Core: (runs anywhere) Windows, Linux, and macOS
- .Net Framework: Websites, services, and desktop apps ..etc on windows
- Xamarin /Mono : a .NET for Mobile (Cross Platform runs on android or ios)

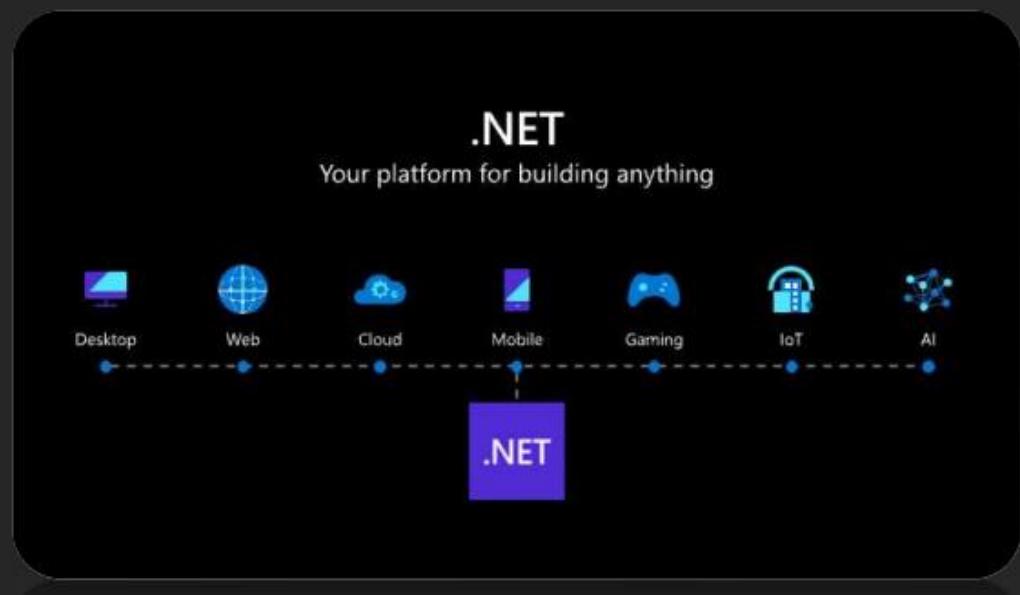
All use a standard set of libraries ☺

يعني اللي بتاخده في ال c# تقدر تعمل بيها تطبيق يشتغل علي أي حاجه  
ال .net فيها مكتبه اسمها net standard library . ودي عبار عن api بتضم الحاجات المشتركة  
بين أي تطبيق علي أي operating system يعني بتكتب التطبيق مرره واحده وال api بينظم تشغيل  
التطبيق ده علي ي سистем

# .NET Platform?



## What can we build using .NET?



<b>Web</b> Build web apps and services for Windows, Linux, macOS, and Docker.	<b>Mobile</b> Use a single codebase to build native mobile apps for iOS, Android, and more.	<b>Desktop</b> Create native apps for Windows and macOS or build apps that run anywhere with web technologies.	<b>Microservices</b> Create independently deployable microservices that run on Docker containers.
<b>Cloud</b> Consume existing cloud services, or create and deploy your own.	<b>Machine learning</b> Add vision algorithms, speech processing, predictive models, and more to your apps.	<b>Game development</b> Develop 2D and 3D games for the most popular desktops, phones, and consoles.	<b>Internet of Things</b> Make IoT apps, with native support for the Raspberry Pi and other single-board computers.

# Who Built .NET?

Microsoft + several thousands of companies 😊

Over 100,000 developers and 3700 Companies Contribute to the .NET Platform 😊

## .NET Platform

- .NET is a platform, created by Microsoft, for building and running many different types of applications including web, desktop, mobile, gaming, IoT, AI, and cloud.
- It is a comprehensive development platform that provides tools and technologies for building all types of applications, including web, desktop, mobile, gaming, IoT, AI, and cloud.
- The .NET platform includes multiple components such as the .NET runtime, the Base Class Library (BCL), and the .NET Standard Library.

## .NET Standard

- .NET Standard is a formal specification of the APIs that are common across .NET implementations. This allows the same code and libraries to run on different implementations.

## .NET Framework

- ".NET Framework" is one of the components of the .NET platform.
- It is a specific implementation of the .NET platform that provides a runtime environment for executing .NET applications and provides a set of libraries for developing and running applications on Windows.
- The .NET Framework was first released in 2002 and has since been updated multiple times to include new features and improvements.

In short, .NET is a platform, while .NET Framework is one of its implementations.

# .Net vs .Net Core

- ".NET" and ".NET Core" are both components of the .NET platform, but they differ in their design goals and use cases.
- ".NET Core" is a cross-platform, open-source, and modular implementation of the .NET platform. It was created to provide a high-performance, scalable, and flexible runtime environment for building modern, cloud-based applications that can run on multiple operating systems, including Windows, macOS, and Linux.
- .NET Core provides a subset of the features and libraries available in the full .NET platform, but it is designed to be faster, lighter-weight, and more efficient.
- In short, .NET is a comprehensive platform for building all types of applications, while .NET Core is a cross-platform, open-source, and modular implementation of the .NET platform focused on modern, cloud-based applications.

## What to Learn .NET Frame Work or .NET Core?

دلوقي طالما اال net core. اتعملت وها احدث من ال net framework. ليه ماتجهش للاحداث؟ قالك ان ال net core. معموله من 2002 وحصل فيها تعديلات وبقت احسن صحيح لكن فيه شركات كتير استثمرت في ال net frame work. وعموا فيها مشاريع وباوعوها عملوا فيها استثمارات كبيرة وعشان ينقلوا للتكنولوجيا الجديد ده بيتطلب سنين كمان تجييك شركه عامله برنامج بـ 10 مليون زرار وقادده بتبعيه وشغاله بيها تيجي انت تقوله يعيد كتابة البرنامج من الصفر لمجرد انه فيه حاجة جديدة ظهرت ؟ علي حساب مين ؟

العميل مش هيرضي يدفع ولو هتعمله علي حسابك هيطلب منك انك تعيد الاستثمار فيه وكمان العميل هيرفض انك تغير البرنامج بتاعه لانه عنده بالفعل برنامج شغال ولو مرتاح معاه عمره ماهيسمحلك انك تلغيه عشان تنزله حاجة جديدة

ممكن تعمل المشاريع الجديدة علي التكنولوجيا الجديدة اه لكن القديمه لا وعشان يحصل التغيير ده الموضوع بيأخذ من 5 ل 10 سنين  
انتعلم netframe work. ولما تلاقي شغل ابقي انلعل اللي انت عايزه

## Compilation in .NET

احنا دلوقي عارفين انه فيه لغات بتكون compiled و فيه لغات interpreted  
دلوقي احنا عارفين انه اال net frame work. فيه لغات كتير اللغات دي بتكون managed languages يعني بيتم ادارتها من قبل مدير المدير ده اسمه clr وهو اختصار ل common language runtime

ال compiler هنا مش بيحول من اللغات دي لل machine code على طول لا بيحولها لحاجه اسمها cil و هو اختصار ل common intermediate language يعني لغه وسيطه وبعدين بيجي اال clr وهو موجود فيه حاجه اسمها JIT و هو اختصار ل just in time compiler

ود تقدر تعتبره interpreter وده وظيفته انه يحول الكود اللي انت عايز لل machine code في ال runtime

يعني لو عندك برنامج وضغطت على زر معين الكود اللي في الزرار ده هوا بس اللي هيتحول لل machine code ويشتغل وده بيدعم ال flexibility

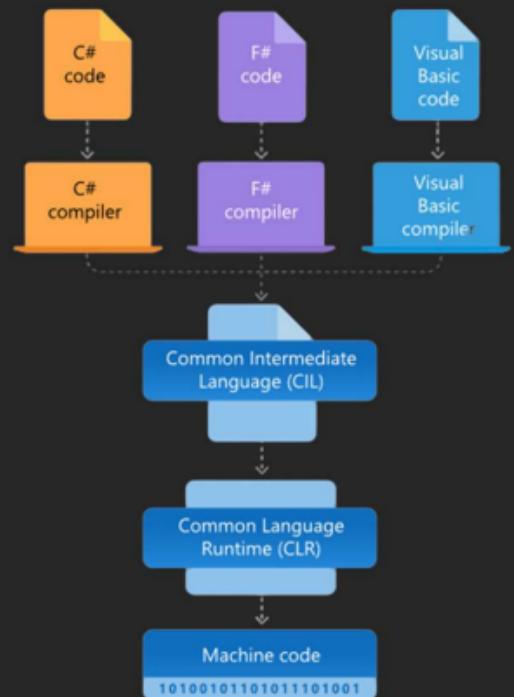
اول مره بيشتغل بيكون بطئ وبعدها بيكون سريع

ولو عايز حاجه تانيه تكون سريعه في ال start up بتروح ل mode تاني اسمه compilation واساعتها ال jit بيطفي

يعني ال net frame work بتدعم الاثنين بس ال default بتاعها هوا ال jit

## Compilation

- .NET applications are written in the C#, F#, or Visual Basic programming language. Code is compiled into a language-agnostic Common Intermediate Language (CIL). Compiled code is stored in assemblies—files with a .dll or .exe file extension.
- When an app runs, the CLR takes the assembly and uses a just-in-time compiler (JIT) to turn it into machine code that can execute on the specific architecture of the computer it is running on.



## Compilation

In .NET, the compilation process occurs in two stages:

1. just-in-time (JIT) compilation.
2. ahead-of-time (AOT) compilation.

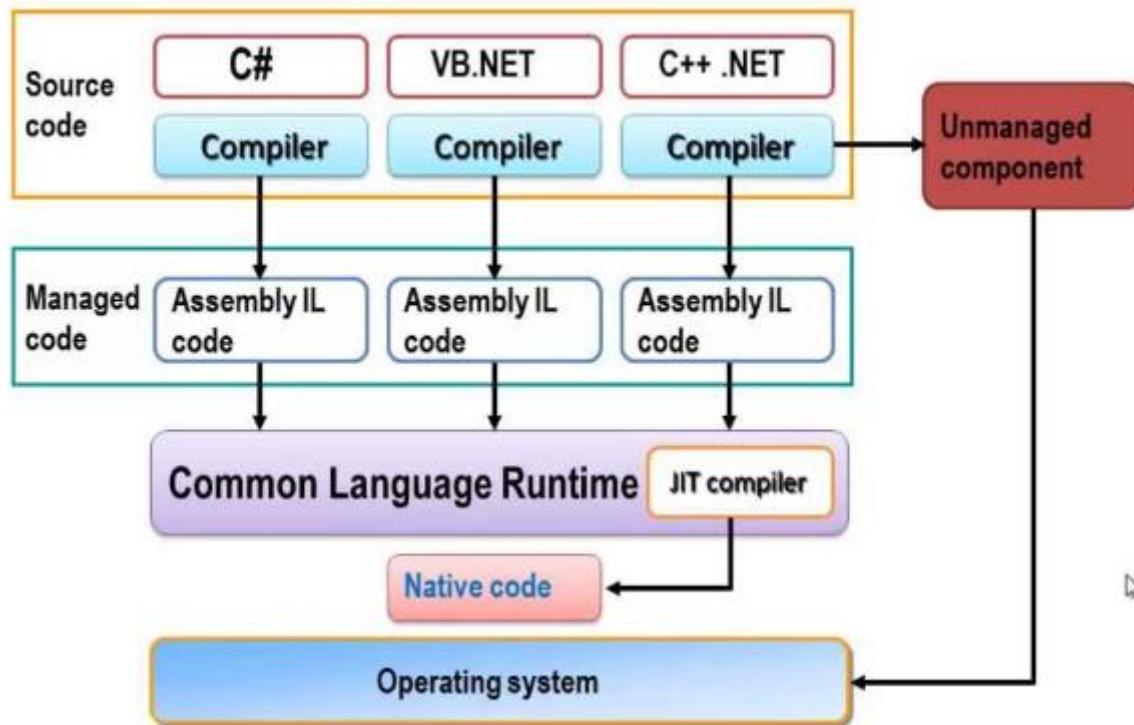
# JIT: JUST In Time Compilation.

- Just-in-Time (JIT) Compilation: During the execution of a .NET application, the CLR performs JIT compilation of the application's Intermediate Language (IL) code into machine code. The JIT compiler translates the IL code into native machine code for the target platform on which the application is running. This allows the application to take advantage of the full performance of the target platform's hardware. The JIT compiler only compiles the methods that are actually executed, so applications start up quickly, and unused code is never compiled.

# AOT: Ahead of Time Compilation.

- AOT stands for Ahead-of-Time, which is a method of compiling .NET code into machine-readable code that can be executed directly by the computer's processor, without requiring an interpreter or just-in-time (JIT) compilation. This approach can lead to faster start times and better performance compared to JIT-compiled code.
- AOT-compiled .NET code is often used in scenarios where performance and startup time are critical, such as in mobile or embedded devices, or in cloud environments where instances are frequently restarted. The AOT-compiled code is optimized for the specific architecture and can take advantage of hardware features such as instruction set extensions and hardware acceleration.
- In .NET, AOT-compilation is usually performed using the Native Image Generator (Ngen.exe) tool, which creates a native image from an existing .NET assembly. The native image can then be deployed and executed on the target machine, without requiring the .NET runtime to be installed.

# The CLR Execution Model



## .NET Framework Architecture

هنا بيقولك انه مايكروسوفت لما سمته framework work هما غلطوا في التسميه لانه ال framework work انجما ال .net frame work. مش بس بينظم الشغل لا ده كمان بيشغلك التطبيق بتاعك فالافضل كانوا يسموه platform

ال framework architecture فيه مكونين اساسيين :-

- : Clr -1
- : BCL -2

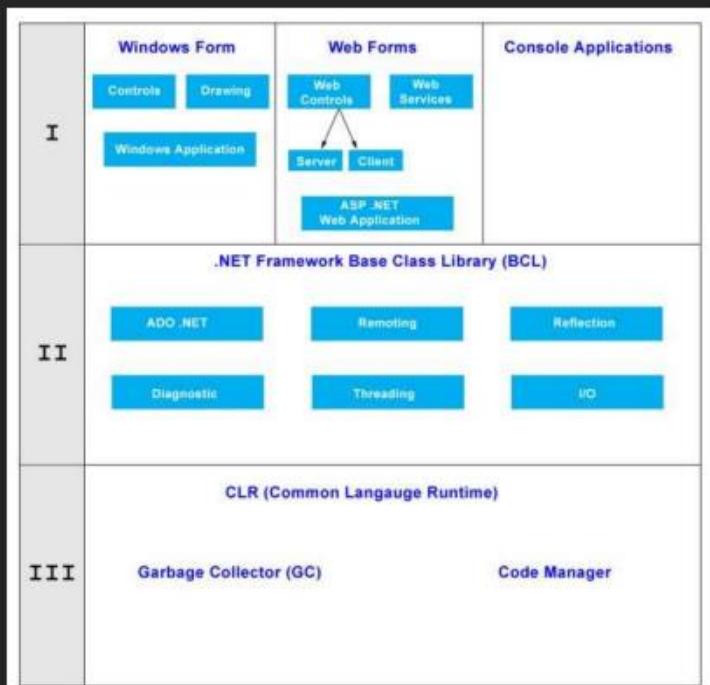
## Architecture of .NET Framework

The two major components of .NET Framework are the Common Language Runtime and the .NET Framework Class Library.

- The Common Language Runtime (CLR) is the execution engine that handles running applications. It provides services like thread management, garbage collection, type-safety, exception handling, and more.
- The Class Library provides a set of APIs and types for common functionality. It provides types for strings, dates, numbers, etc. The Class Library includes APIs for reading and writing files, connecting to databases, drawing, and more.

Another important component is : Languages.

# Architecture of .NET Framework



## Latest Versions

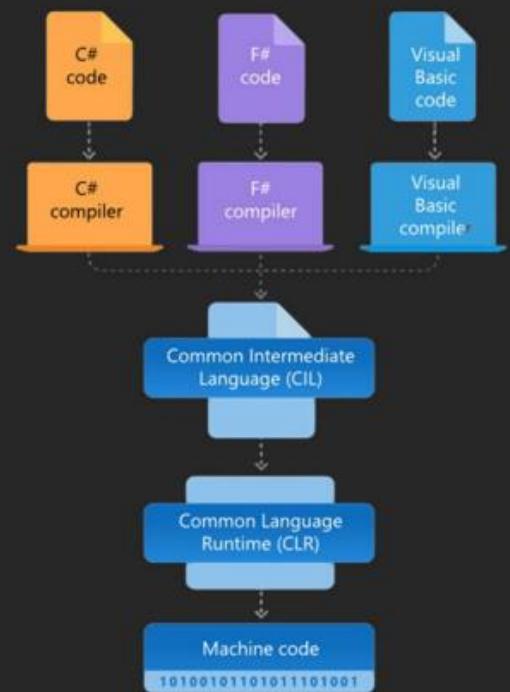
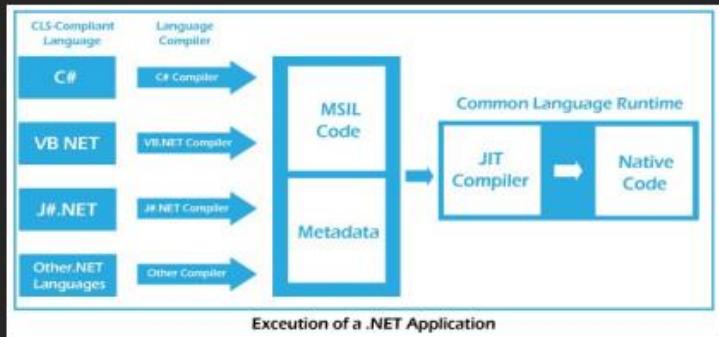
- Latest version of .NET framework is : .NET Framework 4.8.1
- Latest version of .NET is: 6.0

### What is CLR?

ال `clr` هو اختصار ل Common Language Runtime وهو عبارة عن `intermediate execution` والمسؤول عن ال `real life Translator` بتابع البرنامج بتحول من `JIT` machine code ل language `MSIL` هي نفسها ال Microsoft intermediate language

# Compilation

- CLR is a Real Life Translator.



## Common Language Run Time (CLR)

- CLR stands for Common Language Runtime. It is a key component of the .NET platform and is responsible for executing .NET applications.
- The CLR is the runtime environment in which .NET applications are executed. It provides a number of services to .NET applications, including memory management, type safety, security, and exception handling.
- The CLR also manages the execution of .NET code, handling tasks such as just-in-time (JIT) compilation, thread management, and garbage collection.
- In essence, the CLR acts as a bridge between the .NET code and the underlying operating system, allowing .NET code to run on any supported platform.
- This allows .NET developers to write code in any .NET-supported language and have it run on any supported operating system, without having to worry about the underlying hardware and software details.

### CLR Main Components

الـ components الرئيسية في الـ CLR

# CLR Main Components:

- Common type system (CTS)
- Common language specification
- Garbage Collector
- Just in Time Compiler
- Metadata and Assemblies

## 1- Common Type System (CTS)

احنا دلوقتي عندنا ال clr اللي هو بيحول اللغة الوسيطه للغه الاله واللي بيشغل البرنامج

او مكون من المكونات الرئيسيه اللي فيه هيا ال CTS وهيا اختصر ل

Common Type Sysytem

فكترته انك لما بتحول للغه وسيطه فلازم يكون عندك data types وسيطه وهو ده اللي بيعمله ال CTs وهو انه بيحول ال data type بناء اللغه اللي انت كتبتي بيها ال data type خاصه باللغه وسيطه دي زي مثلا ال int في ال c# المقابل ليه في لغه ال vb.net هو ال

هو بقى ال data type الوسيط بناء اسمه Int32

يعني من الاخر البناء ده خاص بتحويل المتغيرات بناء اللغات لأنواع المتغيرات بناء اللغه وسيطه

## Common Type System (CTS):

CTS provides guidelines for declaring, using, and managing data types at runtime.

It offers cross-language communication.

For example, VB.NET has an integer data type, and C# has an int data type for managing integers. After compilation, Int32 is used by both data types.

So, CTS provides the data types using managed code. A common type system helps in writing language-independent code.

## 2- Common Language Specification (CLS)

ال CIS هو اختصار ل Common language specification هو عامل زي القانون او الدستور الموضوع للغات البرمجة اللي تحت ال frame work او عايزه تدخل ضمن ال framework وده معمول عشان يسهل التواصل بين اللغات وبعضها وبيت Hick انك تعمل مشروع مكون من اكتر من لغه

اكيد كل لغه ليها ال syntax بتعاتها لكن فيه قوانين تانيه محظوظه عشان اللغات تتكلم مع بعضها وده اسمه interoperability ومعناه التوافقية

## Common Language Specification (CLS)

Common Language Specification (CLS) contains a set of rules to be followed by all .NET-supported languages.

The common rules make it easy to implement language integration and help in cross-language inheritance and debugging.

Each language supported by .NET Framework has its own syntax rules.

But CLS ensures interoperability among applications developed using .NET languages.

## 3- Garbage Collector (GC)

ال GC هو اختصار ل garbage collector وده موجود في اللغات الحديثه زي ال c# ال C++ كان لازم تنصف وراك بانك تعمل delete في اللغات الحديثه عملوا فيها حد ينصف وراك

## Garbage Collection:

Garbage Collector is a component of CLR that works as an automatic memory manager.

It helps manage memory by automatically allocating memory according to the requirement.

It allocates heap memory to objects.

When objects are not in use, it reclaims the memory allocated to them for future use.

It also ensures the safety of objects by not allowing one object to use the content of another object.

## 4- Just In Time Compilation (JIT)

ال JIT بيحول من لغه وسيطه للغة الاله وفيه منه 3 أنواع:-

- وده اللي هوا ال AOT وده بيقرأ الكود كله علي بعضه قبل ما تشغله التطبيق بيكون اسرع لكن بيستهلك ذاكره اكتر لانه بيخزن كل حاجه عالذاكره قبل ما يشتغل Ngen.exe وبيستخدم

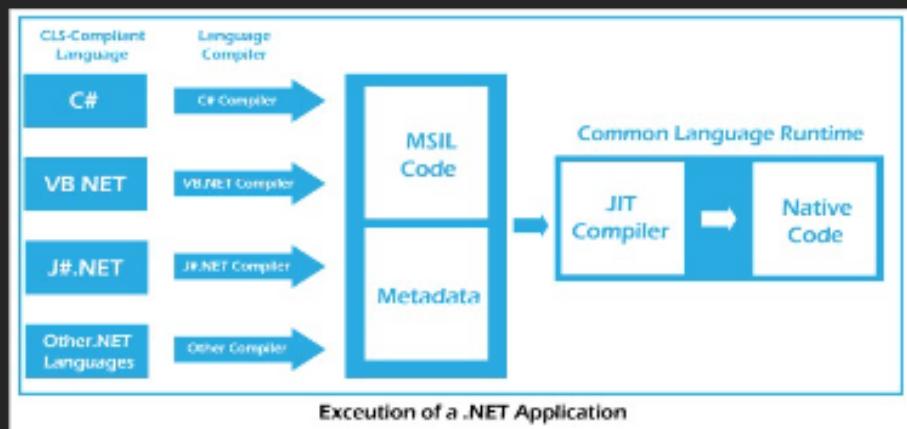
- ده نوع وسط بين النوعين وهو بيعمل compile للكود اللي انت تحتاجه بس مش بيخزنه في ال memory يعني كل مره بتشغل كود بيعمله memory

- وده ال Normal JIT بيعمل compile للكود اللي انت تحتاجه بس والكود اللي بيشتغل بيتخزن في ال memory عشان مایعملش مره تانيه والتخزين ده بيتم في ال AOT بس بيكون ابطأ واستهلاكه من الذاكرة أقل بكثير من ال cache memory

## Just in Time (JIT) Compiler:

JIT Compiler is an important component of CLR.

It converts the MSIL code into native code (i.e., machine-specific code).



## **Just in Time (JIT) Compiler:**

JIT Compiler is an important component of CLR.

It converts the MSIL code into native code (i.e., machine-specific code). The .NET program is compiled either explicitly or implicitly.

The developer or programmer calls a particular compiler to compile the program in the explicit compilation. In implicit compilation, the program is compiled twice. The source code is compiled into Microsoft Intermediate Language (MSIL) during the first compilation process.

The MSIL code is converted into native code in the second compilation process. This process is called JIT compilation.

## **Just in Time (JIT) Compiler:**

The .NET program is compiled either explicitly or implicitly.

There are three types of JIT compilers -Pre, Econo, and Normal.

1. Pre JIT Compiler (AOT Ahead Of Time): compiles entire MSIL code into native code before execution.
2. Econo JIT: Compiler compiles only those parts of MSIL code required during execution and removes those parts that are not required anymore.
3. Normal JIT (Default): Compiler also compiles only those parts of MSIL code required during execution but places them in cache for future use. It does not require recompilations of already used parts as they have been placed in cache memory.

## **Just in Time (JIT) Compiler:**

The .NET program is compiled either explicitly or implicitly.

There are three types:

1. Pre JIT Compiler (AOT Ahead Of Time): (`Ngen.exe`) , everything compiles before start, No compilation at runtime.
2. Econo JIT: Compiled and not cached (each time it will compile again).
3. Normal JIT (Default): only used code is Compiled + Cached.

عادة البرامج الكبير ها السرع ليها هو ال Normal JIT ولو المشروع صغير استعمل ال AOT

In brief:

Compilation are done in 2 stages:

Stage 1: from language to IL ,

Example : c# code to IL by C# compiler

Now we have new language which is IL

Stage2 : From IL to Machine Code

Done by CLR--> JIT

## 5 and 6: Assemblies & Metadata(Manifest)

الكود لما بتعمله compile بيتتحول ل intermediate language وبيسمه assembly code

بعد كده بييجي ال CLR بيحول ال native code ل assembly عن طريق ال Jit

فالكود اللي بيطلع في ال Intermediate language بيسمه assembly ويكون single unit

يعتبر بيعمل الكود بتاعك كله في ملف واحد الملف ده هو اللي بتنزله عند العميل والعميل دي اسمها

deployment

وال assembly عندك منه نوعين :-

- وده اللي هو ال Proccess assembly -1

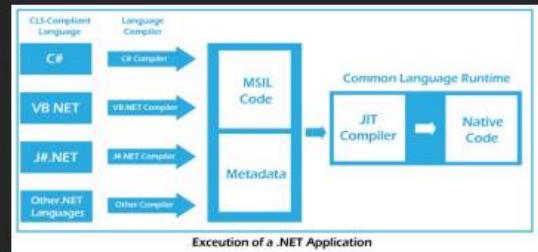
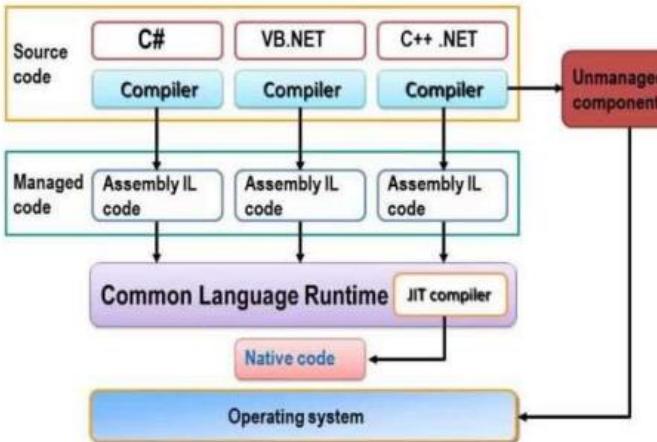
- وده اللي هو ال Library assembly -2

ل dynamic link library وهو برنامج كامل بس مش بيشتغل لوحده بيحتاج تعمله

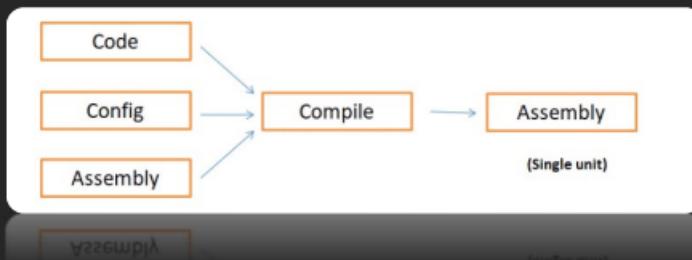
وهو اشبه بالمكتبات اللي بنستخدمها بتكتبها مره واحده وبنستخدمها في عدة مشاريع refrence

# Assemblies:

## The CLR Execution Model



# Assemblies:

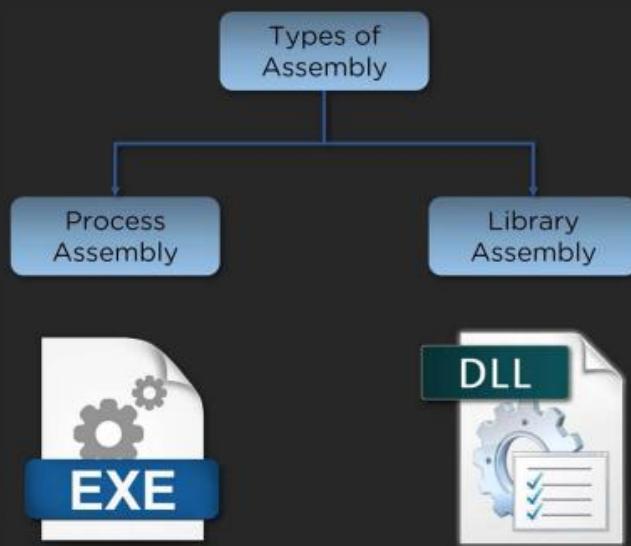


An assembly is a fundamental unit of physical code grouping.

- It is basically a compiled code that can be executed by the CLR.
- Assembly is a single unit of deployment.
- Pre-Compiled Chunk of code that can be executed by CLR.
- It is also considered a basic deployment unit.

# Assemblies:

Assembly can refer to either a dynamic link library (DLL) or an executable file (.exe), depending on its intended use.



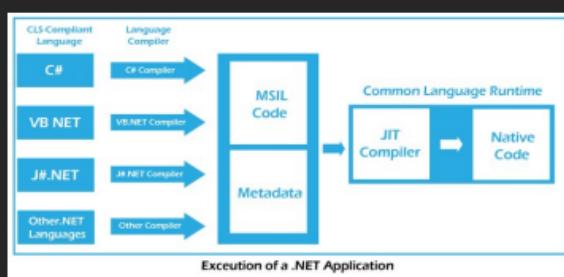
طيب لو انا محتاج اعرف معلومات عن ملف ال exe اعمل ايه؟  
قالك فيه حاجه هنا اسمها manifest او meta data وبنقدر تعرف عن ملف ال exe زي مين ال  
بتاعه والمكتبات اللي اتعمل منها وهكذا  
 يعني ال assembly زي ما بيكون عنده ال .net frame work  
 يعني ال assembly زي ما بيكون عنده ال .net frame work  
 يعني ال assembly زي ما بيكون عنده ال .net frame work  
 يعني ال assembly زي ما بيكون عنده ال .net frame work

ال assembly مهمه عشان لما تعمل compile لازم ال jit يعرف معلومات عن ال assembly  
عشان يعرف يتعامل معاه

## Metadata (Manifest):

Metadata is a Data that provides information about other data.

Metadata is information about the assemblies, modules, and types that constitute .NET programs.



# Metadata (Manifest):

Metadata Contains:

- The assembly's name
- Version number
- Digital signature that uniquely identifies an assembly's creator.
- All files which build up the assembly
- Information regarding all of the referenced assemblies.
- Information of all the exported classes, methods, properties, and other items.

## CLR Functions

ال CLR عرفا انه بيحول من لغة البرمجه للغه وسيطه دي وظيفه واحده من الوظايف اللي بيأدتها  
ودي الوظايف اللي بيعملها

## .NET CLR Functions

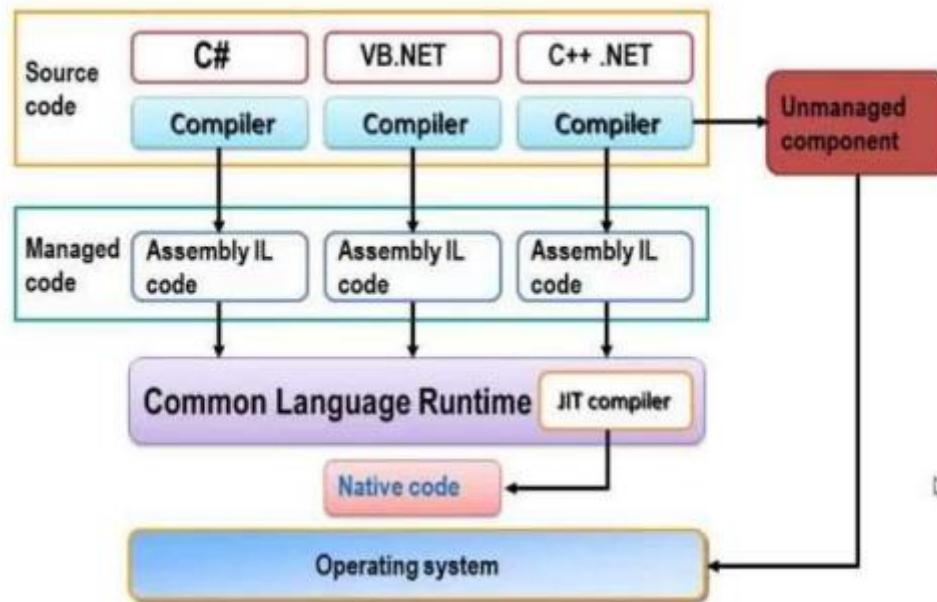
- It converts the program into native code.
- Handles Exceptions.
- Provides type-safety.
- Memory management.
- Provides security.
- Improved performance.
- Language independent.
- Platform independent.
- Garbage collection.
- Provides language features such as inheritance, interfaces, and overloading for object-oriented programs.
- Manager for all .NET supported languages.

## Managed vs Unmanaged Code

أي كود بتكتبه في ال .net. هو عباره عن managed code لأنه بيتمله clr بال compile  
بس فيه حاجه تانيه اسمها unmanaged code ودي هوa كمان بيدعمها زي ال java لنـه مش تحت  
مظلة ال .net. وفيه برضه ال c++

وال CLR بيكون ليه control managed code عال اللي بيطلع من اللغات اللي بيدعمها ال .net  
بس برضه بيوفرك أدوات تخليك تتواصل مع ال unmanaged code عن طريق layer او  
wraper class اسمه libraries

## The CLR Execution Model



## Managed Code

- The code that runs with CLR is called managed code, whereas the code outside the CLR is called unmanaged code.
- The CLR also provides an Interoperability layer, which allows both the managed and unmanaged codes to interoperate.
- Any language that is written in the .NET framework is managed code. Managed code uses CLR, which looks after your applications by managing memory, handling security, allowing cross-language debugging, etc.

## Unmanaged Code

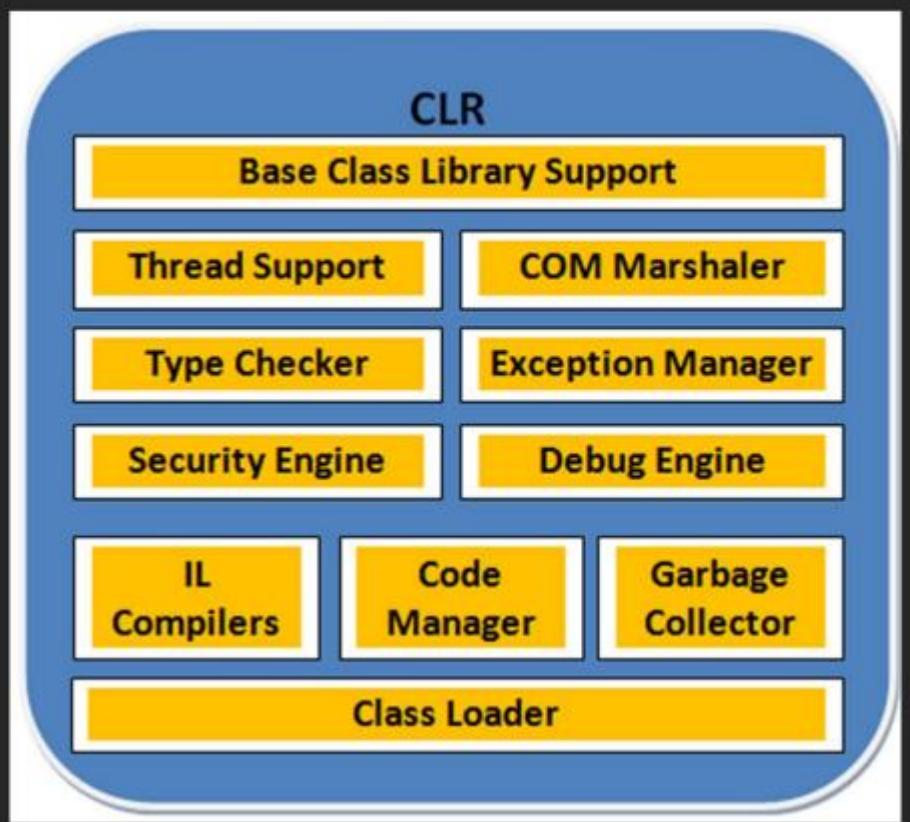
- The code developed outside the .NET framework is known as unmanaged code.
- Applications that do not run under the control of the CLR are said to be unmanaged. Certain languages such as C++ can be used to write such applications, such as low-level access functions of the operating system.
- Background compatibility with VB, ASP, and COM are examples of unmanaged code. This code is executed with the help of wrapper classes.

**COM: Component Object Model:** It's a client server communication using method in the server without the need to compile them into your application. (old technology 1993 by Microsoft).

## CLR Structure

ال thread هو انك تقدر تشغيل عمليتين في نفس الوقت ودي بيدعمها ال clr  
ال class loader مسؤول عن تحميل ال classes عال .net  
ودي باقي الحاجات اللي بيدعمها

# .NET CLR Structure



## Base Class Library

It is a class library that supports classes for the .NET application.

## Thread Support

It manages the parallel execution of the multi-threaded application.

## **COM Marshaler**

It provides communication between the COM objects and the application.

## **Security Engine**

It enforces security restrictions.

## **Debug Engine**

It allows you to debug different kinds of applications.

## **Type Checker**

It checks the types used in the application and verifies that they match the standards provided by the CLR.

## **Code Manager**

It manages code at execution runtime.

## **Garbage Collector**

It releases the unused memory and allocates it to a new application.

## **Exception Handler**

It handles the exception at runtime to avoid application failure.

## **Class Loader**

It is used to load all classes at run time.

## .NET Framework Class Library (FCL)

هنتكلم عن الـ **base class library** وهو عباره عن مجموعه من الكلاس وعددها بالالاف ودي امثله **ال حاجات اللي بتدعيمها وهيا بتدعيم كل حاجه تتخيلها**

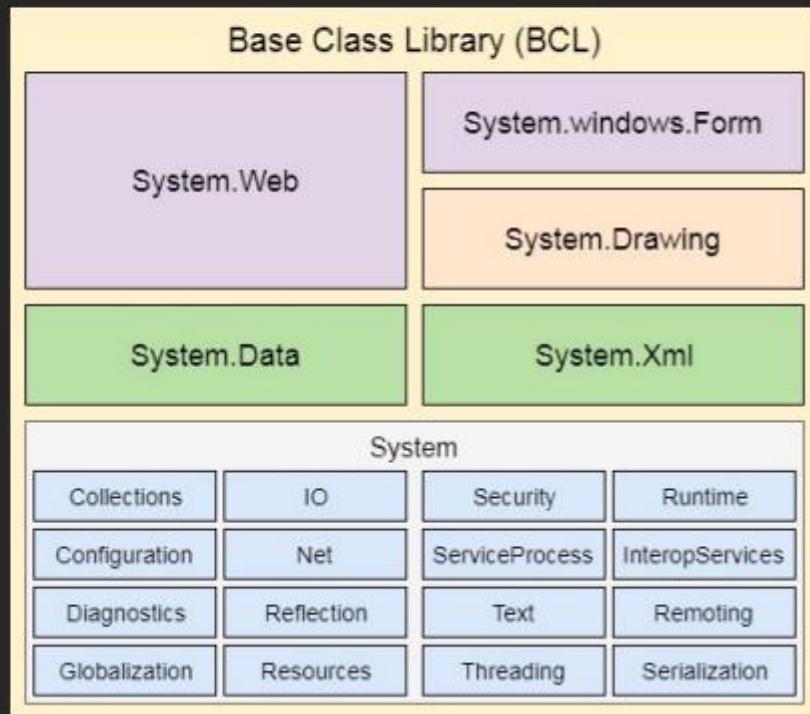
# **.NET Framework Class Library**

**.NET Framework Class Library** is the collection of classes, namespaces, interfaces and value types that are used for .NET applications.

It contains thousands of classes that supports the following functions.

- Base and user-defined data types
- Support for exceptions handling
- input/output and stream operations
- Communications with the underlying system
- Access to data
- Ability to create Windows-based GUI applications
- Ability to create web-client and server applications
- Support for creating web services

# **.NET Framework Class Library**



**What is C#? and Why to Learn It?**

ال syntax هنا مشابه لـ C++ وهذا هو يشرح الفروق بس مالوش داعي يشرح مفاهيم من أول وجديد

هيا لغة متعددة الأهداف وبتدعم ال OOP مابد عمش ال functional programming يعني ماينفعش تكتب function في الشارع (زمن الهايريباء انتهي)

هيا تم إصدارها 2002 واخر اصدار ليها 2022 وهي واحدة من اهم لغات البرمجة وشهرها في تطوير التطبيقات ومبنيه على لغه ال C ومن عائلة ال C زي ال C و ال C++ و ال Java

و فيها restrictions انه ماينفعش تخزن int في string فيها .net وفيها كل حاجه بتدعمها ال

و تم تسميتها C# على اسم درجه موسيقيه لو لعبت موسيقي غربي وعرفت السلم بتاعه هتفهم الكلام ده  
المقابل الشرقي ليه هو ديز

و تقدر تعمل بيها أي حاجه تحتاجه برمجه

## What is C#?

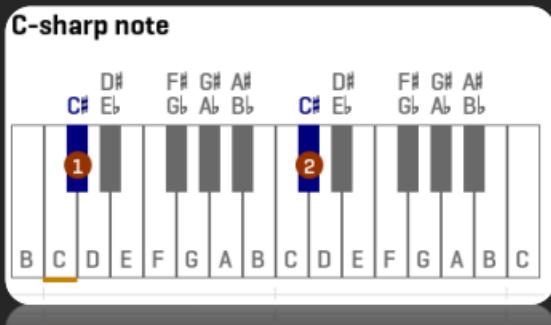
- C# (pronounced as C sharp) is a general-purpose, object-oriented programming language.
- The first version was released in year 2002. The latest version, C# 11, was released in November 2022.
- It is one of the most popular languages used for developing desktop and web applications.
- Being a C based language, C# is closer to C++ and C. Syntactically, it is similar to Java.
- Modern – C# is a modern and powerful language that allows developers to build robust applications quickly and easily. It is built based on the current trend

# What is C#?

- Simple – The code written in C# is much simpler and easier to understand. It is syntactically very similar to Java.
- Type Safe – C# ensures that each variable of a particular type does not hold values of other types.
- Object-oriented – C# supports the object-oriented paradigm such as objects, classes, inheritance, polymorphism, etc.
- Being a high-level language, the basic constructs of C# is easy to understand.
- C# has a huge community. Hence a C# related question is more likely to be answered quickly

## Why it's called C#?

The name "C sharp" was inspired by the musical notation whereby a sharp symbol indicates that the written note should be made a semitone higher in pitch.



## What can we do with C#?

Everything we can do with .NET ☺



## Syntax

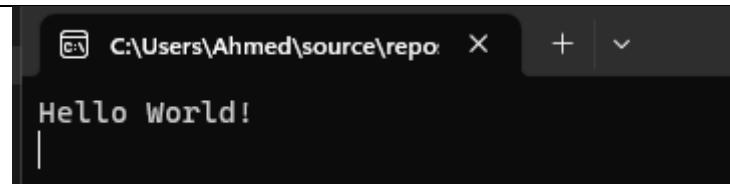
المكتبات دي كلها مش عايزين منها غير اول واحد

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
```

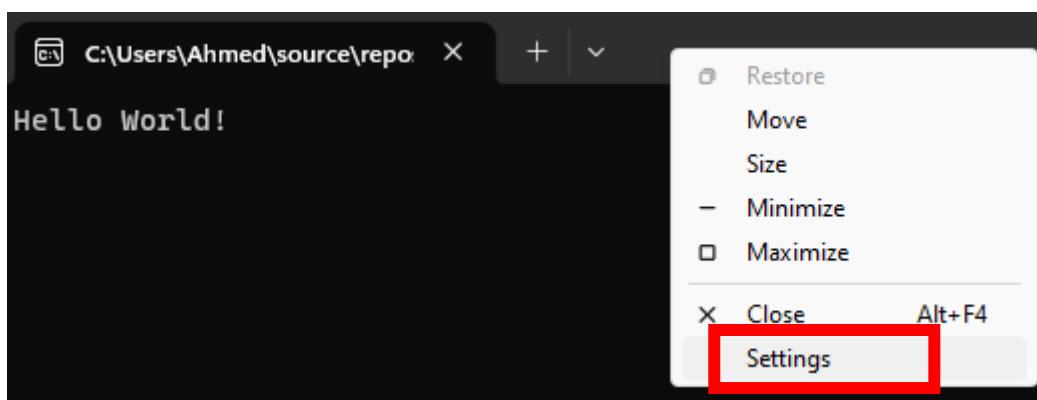
وقمنا كتبنا جمله طباعه

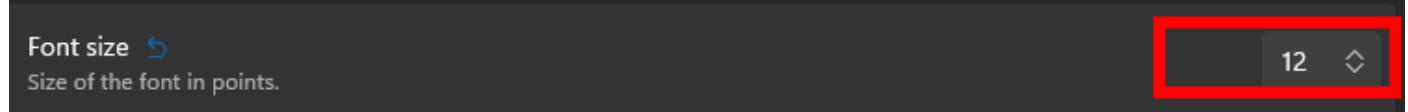
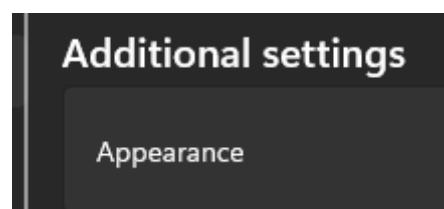
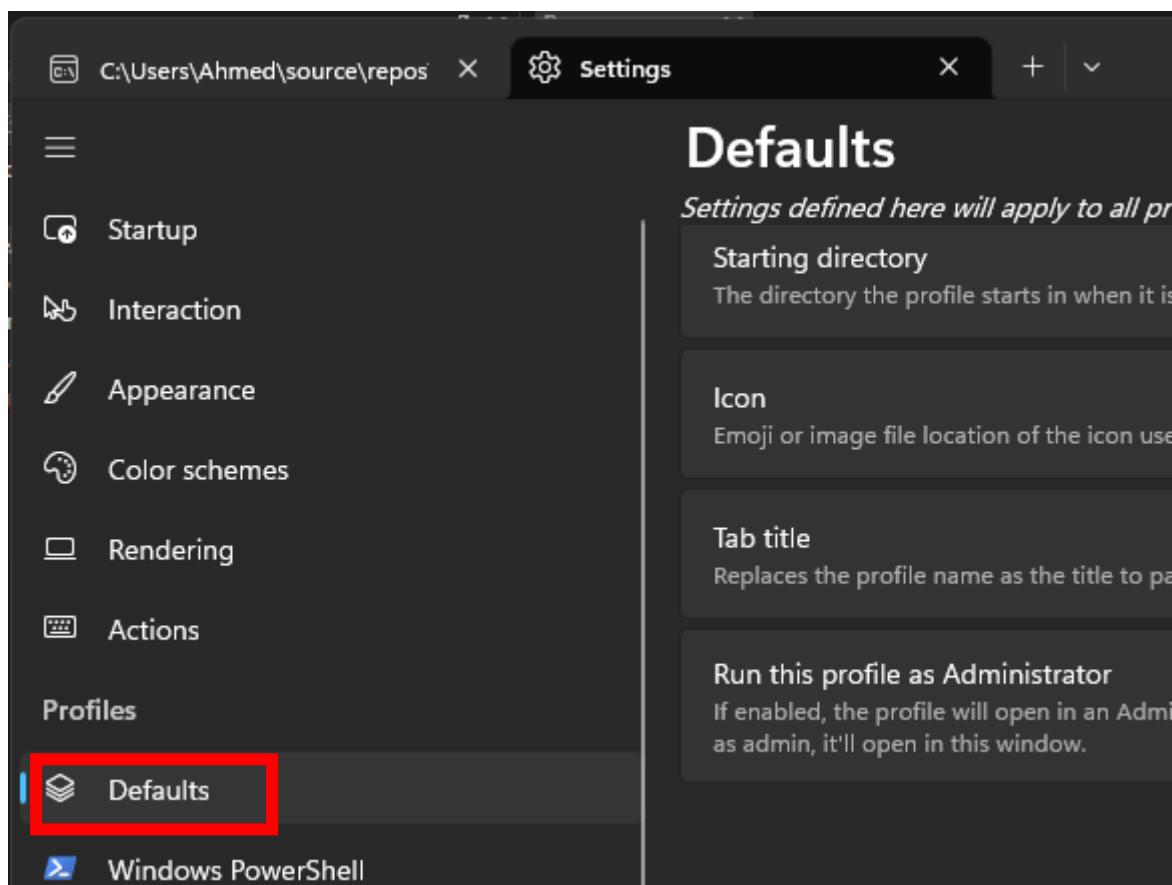
```
using System;

namespace ConsoleApp1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
            Console.ReadLine();
        }
    }
}
```



عشان تكبر الخط





ال C++ includeiostream في ال system using ليها مقابل

`using System;`

وبعدين عندك ال namespace

Screenshot of the code editor showing the 'namespace ConsoleApp1' declaration. A red box highlights the namespace declaration.

وبعدين كلاس اسمه program

internal class Program

وجوه الكلاس بتلاقى ال main اللي بتكتب جواها الكود

```

0 references
static void Main(string[] args)
{
    Console.WriteLine("Hello World!");
    Console.ReadLine();
}

```

## WirteLine

في ال C++ كنا بنستخدم ال cout

هنا هنستخدم Console.WriteLine وكل جمله بتنكتب في سطر يعني لو عايز تنزل سطر جديد  
بنستدعي ال function ثانوي

العمليات الحابيه بتحطها بين قوسين والا هيرصهم جنب بعض

```

//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
            Console.WriteLine("My Name is Mohammed Abu-Hadhoud");
            Console.WriteLine("I am Learning C#");
            Console.WriteLine("It is an easy language to learn :-)");
            Console.WriteLine("The sum of 10 + 20 is " + (10 + 20));
            Console.WriteLine("The sum of 10 + 20 is " + 10 + 20);
            Console.ReadKey();
        }
    }
}

```

## Write

لو مش عايز تنزل سطر جديد بستدعي ال write اللي اسمها و هيكتب الكلام جنب بعضه

```

//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {
        static void Main(string[] args)
        {

```

```

        Console.WriteLine("Hello World!");
        Console.WriteLine("My Name is Mohammed Abu-Hadhoud");
        Console.Write("I am Learning C#");
        Console.Write("It is an easy language to learn :-)");
        Console.Write("The sum of 10 + 20 is " + (10 + 20));
        Console.ReadKey();

    }
}
}

```

## Formatted String

لو فاكر ال print f فال موضوع هنا زيه

بتكتب الكلام اللي انت عاييز تكتبه ولو فيه parameter معينه عاييز تحشرها في النص بتكتب الرقم بين القوسين دول {} وبتعمل فاصله وبعدين تديله ال parameter وهو بيعوض عن الرقم بالقيمه اللي في ال parameter

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("{0} {1}", "Welcome to", "ProgrammingAdvices");
            Console.WriteLine("Hi MyName is: {0} I live in {1}", "Mohammed", "Jordan");

            Console.ReadKey();
        }
    }
}

```

في اول جمله بيعوض عن الرقم صفر بال welcome to والرقم 1 بيعوض عنه بال programming advices

## Escape Characters

الرموز زي انك تنزل سطر جديد او ال \t او ال \n او ال \a هنا نفس الكلام فيه بس حاجه هنا انك لو عملت b \a بيمسح الحرف اللي قبله

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

```

```

static void Main(string[] args)
{
    Console.WriteLine("Useful Escape Characters:\n");

    //Newline
    Console.WriteLine("Newline:");
    Console.WriteLine("Welcome to \n ProgrammingAvices\n");

    //Tab
    Console.WriteLine("Tab:");
    Console.WriteLine("Welcome to\tProgrammingAvices\n");

    //Backspace
    Console.WriteLine("Backspace:");
    Console.WriteLine("Welcome to \bProgrammingAvices\n");

    //Single quote
    Console.WriteLine("Single Quote:");
    Console.WriteLine("Welcome to \' ProgrammingAvices\n");

    //Double quote
    Console.WriteLine("Double Quote:");
    Console.WriteLine("Welcome to \" ProgrammingAvices\n");

    //Backslash
    Console.WriteLine("Backslash:");
    Console.WriteLine("Welcome to \\ ProgrammingAvices\n");

    //Alert
    Console.WriteLine("Alert:");
    Console.WriteLine("\a");

    Console.ReadKey();
}
}
}

```

## Single Line/Multiple Lines Comments

ال comments زیها زی ال c++

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //this is a single line comment
            Console.Write("My Name is Mohammed Abu-Hadhoud ");
            /*
            This is
        }
    }
}

```

```
a multiple line  
comment.
```

```
*/  
Console.ReadKey();  
  
}  
}  
}
```

## Variables

تعيين المتغيرات هيا هيا بنفس الاسم مع تغيير بسيط ان ال **doube** بتكتب **d** بعد الرقم

```
//ProgrammingAdvises.com  
//Mohammed Abu-Hadhoud
```

```
using System;  
  
namespace Main  
{  
    internal class Program  
    {  
        static void Main(string[] args)  
        {  
            string MyName = "Mohammed Abu-Hadhoud";  
            Console.WriteLine(MyName);  
  
            int x = 10; int y = 20;  
  
            Console.WriteLine("x=" + x);  
            Console.WriteLine("y=" + y);  
            //this line will give wrong answer :-)  
            Console.WriteLine("x+y=" + x + y);  
  
            //this line will give right answer :-)  
            Console.WriteLine("x+y=" + (x + y));  
  
            //other common data types  
            double MyDouble = 25.89D;  
            char MyLetter = 'M';  
            bool MyBool = true;  
  
            Console.ReadKey();  
        }  
    }  
}
```

## Rules for Naming Variables in C#

قواعد التسمية هيا نفسها بتاعت ال C++ بس هنا فيه حاجه زياده انك تقدر تكتب @ و بتتفع لو استخدمت مثلاً كلمة زي **break** دي ممنوعه لانه محجوزه فلما تحط قبلها ال @ بتشتغل عادي

# Rules of Naming Variables

- The variable name can contain letters (uppercase and lowercase), underscore( \_ ) and digits only.
- The variable name must start with either letter, underscore or @ symbol, Examples:

Rules for naming variables in C#	
Variable Names	Remarks
name	Valid
subject101	Valid
_age	Valid (Best practice for naming private member variables)
@break	Valid (Used if name is a reserved keyword)
101subject	Invalid (Starts with digit)
your_name	Valid
your name	Invalid (Contains whitespace)

# Rules of Naming Variables

- C# is case sensitive. It means age and Age refers to 2 different variables.
- A variable name must not be a C# keyword. For example, if, for, using can not be a variable name. keywords can found in the link below:

<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/>

<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/>

## Implicitly Typed Variables

في ال C++ لو مش عايز تسمى ال variable سواء int او string كنت بتكتب auto هنا بقى بتكتب var بس شرطها انك تعملها initial value

//ProgrammingAdvises.com  
//Mohammed Abu-Hadhoud

```
using System;

namespace Main
{
    internal class Program
    {
        static void Main(string[] args)
        {

            /* Implicitly typed variables
            Alternatively in C#, we can declare a variable without knowing
            its type using var keyword.
```

Such variables are called implicitly typed local variables.

Variables declared using var keyword must be initialized at the time of declaration.

\*/

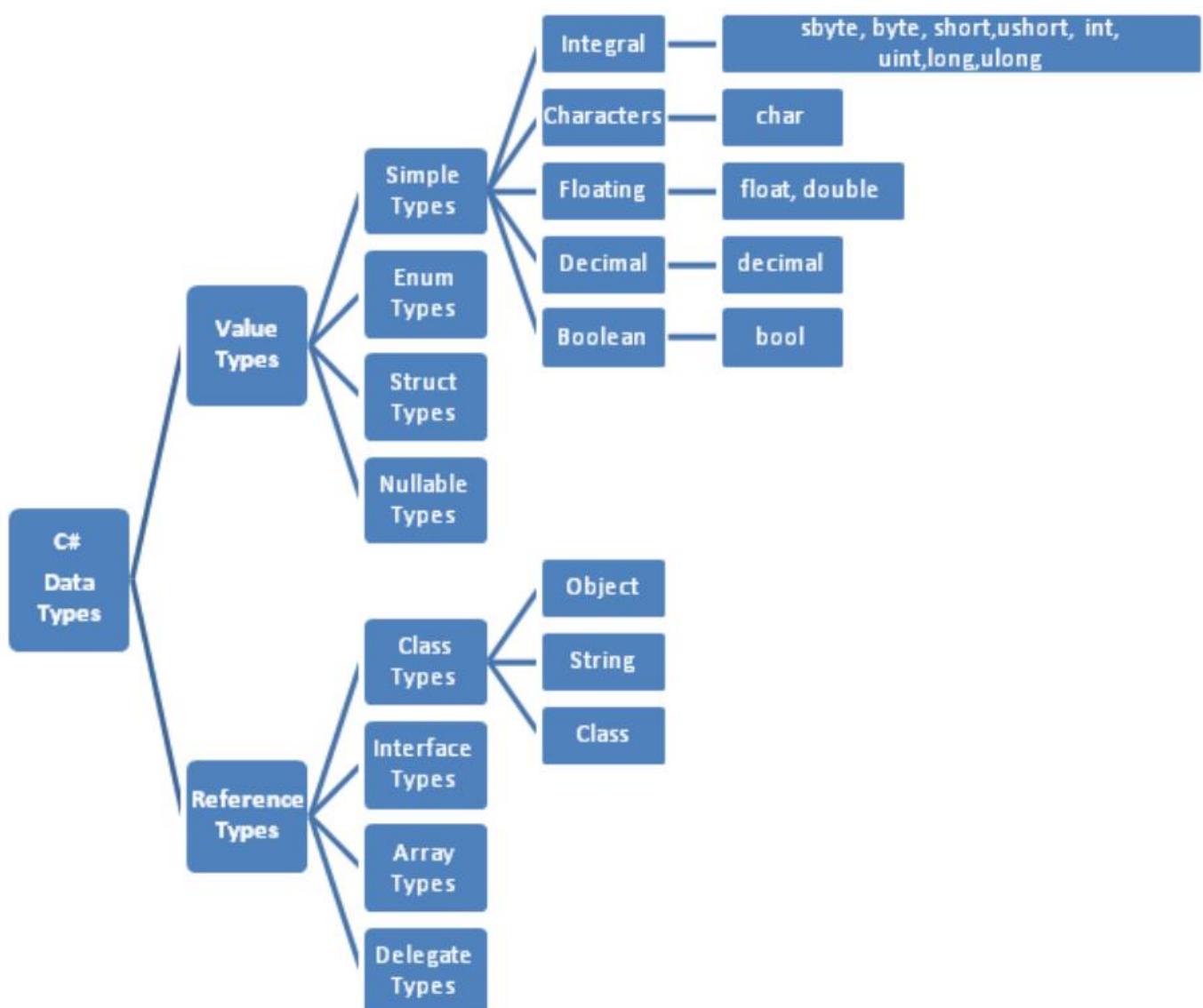
```
var x = 10;  
var y = 10.5;  
var z = "Mohammed";
```

```
Console.WriteLine("x={0}, y={1}, z={2}", x, y, z);
```

```
Console.ReadKey();
```

```
}
```

## Datatypes



## Predefined Datatypes

هنا بيعرفك على ال datatypes في ال c# والمقابل ليهم في ال .net

هنا فيه حاجه اسمها suffix و هي حرف او حرفين بتكتبهم بعد الرقم

Type	Description	Size (bytes)	.NET type	Range
int	Whole numbers	4	System.Int32	-2, 147, 483, 648 to 2, 147, 483, 647
long	Whole numbers (bigger range)	8	System.Int64	-9, 223, 372, 036, 854, 775, 808 to 9, 223, 372, 036, 8, 54, 775, 807
float	floating-point numbers	4	System.Single	+/-3.4 x 10^38
double	Double precision (more accurate) FP numbers	8	System.Double	+/-1.7 x 10^308
decimal	Monetary values	16	System.Decimal	28 significant figures
char	Single character	2	System.Char	N/A
bool	Boolean	1	System.Boolean	True or False
DateTime	Moments in time	8	System.DateTime	0:0:00 on 01/01/0001 to 23:59:59 on 12/31/9999
string	Sequence of characters	2 per character	System.String	N/A

Type	Description	Range	Suffix
byte	8-bit unsigned integer	0 to 255	
sbyte	8-bit signed integer	-128 to 127	
short	16-bit signed integer	-32,768 to 32,767	
ushort	16-bit unsigned integer	0 to 65,535	
int	32-bit signed integer	-2,147,483,648 to 2,147,483,647	
uint	32-bit unsigned integer	0 to 4,294,967,295	u
long	64-bit signed integer	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	l
ulong	64-bit unsigned integer	0 to 18,446,744,073,709,551,615	ul
float	32-bit Single-precision floating point type	-3.402823e38 to 3.402823e38	f
double	64-bit double-precision floating point type	-1.79769313486232e308 to 1.79769313486232e308	d
decimal	128-bit decimal type for financial and monetary calculations	(+ or -)1.0 x 10e-28 to 7.9 x 10e28	m
char	16-bit single Unicode character	Any valid character, e.g. a,*, \x0058 (hex), or \u0058 (Unicode)	
bool	8-bit logical true/false value	True or False	
object	Base type of all other types.		
string	A sequence of Unicode characters		
DateTime	Represents date and time	0:00:00am 1/1/01 to 11:59:59pm 12/31/9999	

## Numbers Datatypes

الأرقام مقسمه نصين النص الأول ده اللي مافيهاش كسور وده اسمه inter types والنص الثاني أرقام فيها كسور ودي اسمها floating point types وفيه مقابل الأنواع دي بالسالب بيعبر عنها بال s ودي اختصار signed او u ودي اختصار unsigned

يعني ال byte ماقبلش بالرقم السالب وعشان تخلية يقبل بييه بتزود حرف ال s عشان يكون

وال ushort مابقفلش سالب ولو عايزها تقبل سالب بتخليةها

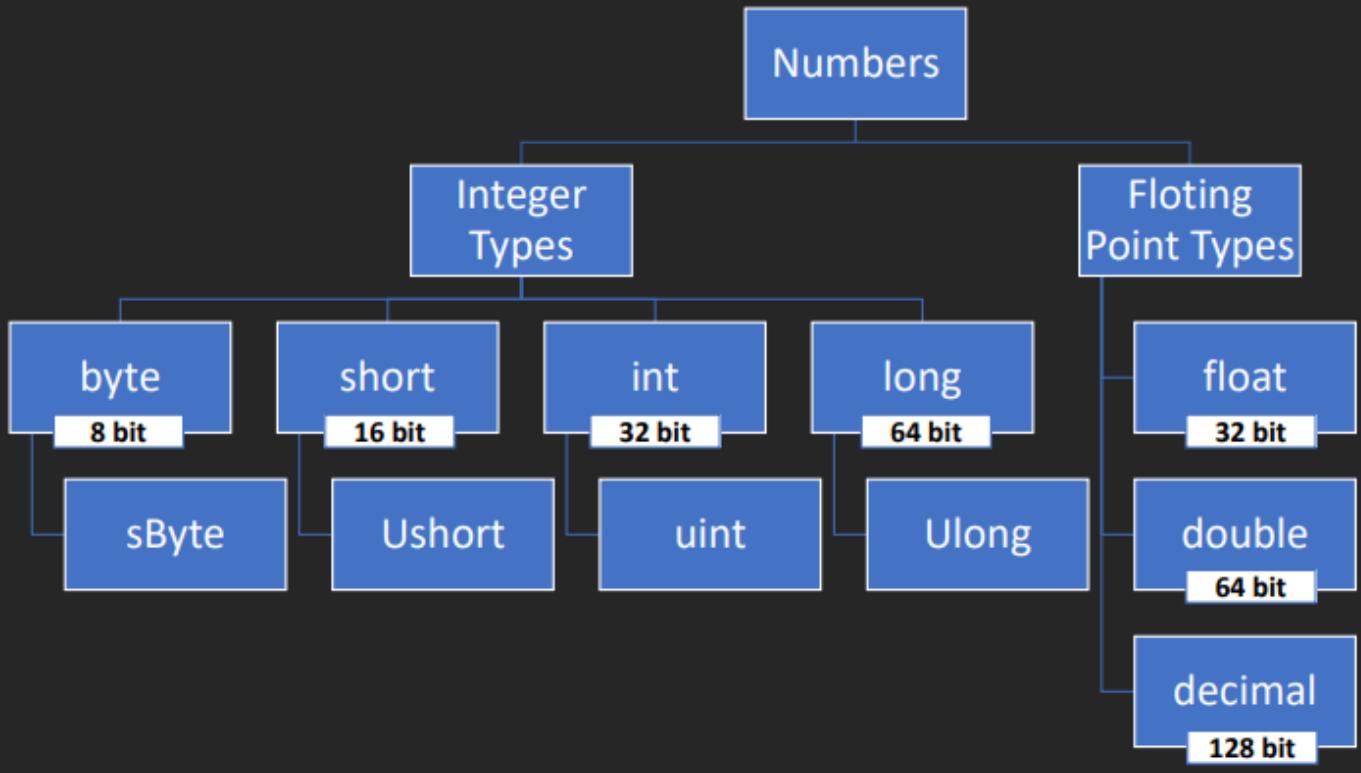
لو عايز تعرف ال min وال max اللي هوا اصغر واكبر رقم بيقدر أي متغير يخزنها بتكتب اسمه maxValue أو minValue function

بتقدر تكتب الرقم كده 0.12e2 وكده معناه انه ال 0.12 مضروب في 10 اس 2 اللي هوا 100 يعني

وبنقدر تخزن الرقم بال hexadecimal لو كتبت قبله 0x زي كده 0x2F وتخزنها في int عادي

وبنقدر تخزن ال binary بانك تكتب قبله 0b زي كده 0b\_0010\_1111

# Numbers Datatypes



//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud

```
using System;

namespace Main
{
    internal class Program
    {
        static void Main(string[] args)
        {

            //Byte

            byte b1 = 255;
            // byte b2 = -128;// compile-time error: Constant value '-128' cannot be converted to a 'byte'
            sbyte sb1 = -128;
            sbyte sb2 = 127;
            Console.WriteLine("\nByte:");
            Console.WriteLine("Min={0} , Max={1}", Byte.MinValue, Byte.MaxValue);

            Console.WriteLine("\nSByte:");
            Console.WriteLine("Min={0} , Max={1}", SByte.MinValue, SByte.MaxValue);

            //Short
            short s1 = -32768;
            short s2 = 32767;
            // short s3 = 35000;//Compile-time error: Constant value '35000' cannot be converted to a 'short'
```

```
ushort us1 = 65535;
// ushort us2 = -32000; //Compile-time error: Constant value '-32000' cannot be converted to a 'ushort'

Console.WriteLine("\nShort:");
Console.WriteLine("Min={0} , Max={1}", Int16.MinValue, Int16.MaxValue);

Console.WriteLine("\nUShort:");
Console.WriteLine("Min={0} , Max={1}", UInt16.MinValue, UInt16.MaxValue);

//int
int i = -2147483648;
int j = 2147483647;
// int k = 4294967295; //Compile-time error: Cannot implicitly convert type 'uint' to 'int'.

uint ui1 = 4294967295;
// uint ui2 = -1; //Compile-time error: Constant value '-1' cannot be converted to a 'uint'

Console.WriteLine("\nInt:");
Console.WriteLine("Min={0} , Max={1}", Int32.MinValue, Int32.MaxValue);

Console.WriteLine("\nUInt:");
Console.WriteLine("Min={0} , Max={1}", UInt32.MinValue, UInt32.MaxValue);

//Long
long l1 = -9223372036854775808;
long l2 = 9223372036854775807;

ulong ul1 = 18223372036854775808ul;
ulong ul2 = 18223372036854775808UL;

Console.WriteLine("\nLong:");
Console.WriteLine("Min={0} , Max={1}", Int64.MinValue, Int64.MaxValue);

Console.WriteLine("\nULong:");
Console.WriteLine("Min={0} , Max={1}", UInt64.MinValue, UInt64.MaxValue);

//Float
float f1 = 123456.5F;
float f2 = 1.123456f;

Console.WriteLine("\nFloat:");
Console.WriteLine("Min={0} , Max={1}", float.MinValue, float.MaxValue);

//double
double d1 = 12345678912345.5d;
double d2 = 1.123456789123456d;

Console.WriteLine("\nDouble:");
Console.WriteLine("Min={0} , Max={1}", double.MinValue, double.MaxValue);

//Decimal
//The decimal type has more precision and a smaller range
//than both float and double,
//and so it is appropriate for financial and monetary calculations.
decimal d3 = 123456789123456789123456789.5m;
decimal d4 = 1.1234567891345679123456789123m;
```

```

Console.WriteLine("\nDecimal:");
Console.WriteLine("Min={0} , Max={1}", decimal.MinValue, decimal.MaxValue);

//Scientific Notation
//Use e or E to indicate the power of 10
//as exponent part of scientific notation with float, double or decimal.

double d = 0.12e2;
Console.WriteLine(d); // 12;

float f = 123.45e-2f;
Console.WriteLine(f); // 1.2345

decimal m = 1.2e6m;
Console.WriteLine(m); // 1200000

//hex & Binary
int hex = 0x2F;
int binary = 0b_0010_1111;

Console.WriteLine(hex);
Console.WriteLine(binary);

Console.ReadKey();

}
}
}

```

## Default Values

في ال C++ لو كنت سبب التغيير من غير initialization كان بيطلك فيها rubbish data يعني قيم قديمه

هنا بيقولك ان المتغيرات اللي نوعها ارقام سواء كانت int او float او أي حاجه ال default بتاعها بصفر

وال bool ال default بتاعه ب false

وال char ال default بتاعه '0'

ولو عايز تجيب default value بتاعت ال data type بتسدعي ال function اللي اسمها بتديها ال default data type

ولو عايز تخزن ال default بتاعت ال int في متغير من النوع int ممكن تكتب default على طول

```

//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{

```

```

internal class Program
{
    static void Main(string[] args)
    {
        //get default value using default(type)
        int i = default(int); // 0
        float f = default(float); // 0
        decimal d = default(decimal); // 0
        bool b = default(bool); // false
        char c = default(char); // '\0'

        // C# 7.1 onwards
        //get default value using default
        int i2 = default; // 0
        float f2 = default; // 0
        decimal d2 = default; // 0
        bool b2 = default; // false
        char c2 = default; // '\0'

        Console.WriteLine(i);
        Console.WriteLine(f);
        Console.WriteLine(d);
        Console.WriteLine(b);
        Console.WriteLine(c);

        Console.WriteLine(i2);
        Console.WriteLine(f2);
        Console.WriteLine(d2);
        Console.WriteLine(b2);
        Console.WriteLine(c2);

        Console.ReadKey();
    }
}

```

## Enums

ال زيها زي ال **C++** و ممكن ترقم انت العناصر وممكن تسييدها و هيا تترقم لوحدها بداية من الصفر بس فيه حاجتين زياده:-

اول حاجه انك ترقم عنصر واحد بس من جوه ال **enum** لو ال **enum** بتاعتك مثلا من 3 عناصر وجيit اديت العنصر الثاني الرقم 5 مثلا

هنا هوا هيبدا في الترميم الافتوماتيك عادي من الصفر لحد مايوصل للعنصر اللي انت اديته رقم يديله الرقم اللي انت اديته وبقيت العناصر اللي بعده تبدأ تعد من بعد الرقم ده

يعني اول عنصر يكون رقمه صفر وتاني عنصر يكون رقمه 5 عشان انت رقمته والعنصر الثالث ياخد الرقم 6

تاني حاجه انك تقدر تخلی الترقيم كله يكون من نوع int او short او long او أي نوع متغير بس  
طبعا ملينفعش يكون string

### Enums in C# is the Same as C++

however you can specify a type for Enum :-)

see the code to understand them.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        enum enWeekDays
        {
            Monday,    // 0
            Tuesday,   // 1
            Wednesday, // 2
            Thursday,  // 3
            Friday,    // 4
            Saturday,  // 5
            Sunday     // 6
        }

        //if you set a value , it will auto number everything after it.
        enum enCategories
        {
            Electronics, // 0
            Food,        // 1
            Automotive = 6, // 6
            Arts,         // 7
            BeautyCare,   // 8
            Fashion       // 9
        }
    }
}

//Enum can have any numerical data type byte, sbyte, short, ushort, int, uint, long, or ulong
//but not string

enum enCategories2 : byte
{
    Electronics = 1,
    Food = 5,
    Automotive = 6,
    Arts = 10,
    BeautyCare = 11,
    Fashion = 15
```

```

    }

static void Main(string[] args)
{
    enWeekDays WeekDays;
    WeekDays = enWeekDays.Friday;

    Console.ReadKey();
}
}
}
}

```

## Nullable Types

هنا بيقولك انك تقدر تعمل متغير وتخلي قيمته ب null عن طريق انك تكتب كلمه Nullable وبعدين تعرف ال template class وتقول ان المتغير بيساوي null

## Nullable Types

As you know, a value type cannot be assigned a null value. For example, `int i = null` will give you a compile time error.

C# 2.0 introduced nullable types that allow you to assign null to value type variables. You can declare nullable types

using `Nullable<t>` where T is a type.

```
Nullable <int> i = null;
```

example, `Nullable<int>` can be assigned any value from -2147483648 to 2147483647, or a null value.

see the code.

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            // Nullable<int> can be assigned any value
            // from -2147483648 to 2147483647, or a null value.

            Nullable<int> i = null;
        }
    }
}
```

```
Console.ReadKey();  
}  
}  
}
```

## Anonymous Type

تسمع عن الزرع اللي بيطلع شيطاني؟ تلاقي نفسك ماشي في الشارع وتلاقي زرع محدث بيراعيه وبينبت؟

في البرمجه بقى عندنا الكلاس الشيطاني او ال anonymous type تكتب كلمه var وتفتح قوسين وتكلب اسم اي المتغيرات من مانقول ان كان int ولا ايه نوعه بالظبط وتدليله القيمه بتاعته وهو اصبح عندك كلاس زي كده

```
var student = new { Id = 20, FirstName = "Mohammed", LastName = "Abu-Hadhoud" };
```

مش كده وبس

انت كمان تقدر تطبعه من غير ماتستدعي اي حاجه فيه زي كده

```
Console.WriteLine(student);
```

وتقدر تعمل جوه الكلاس ده كلاس تاني شيطاني برضه

```
var student2 = new  
{  
    Id = 20,  
    FirstName = "Mohammed",  
    LastName = "Abu-Hadhoud",  
    Address = new { Id = 1, City = "Amman", Country = "Jordan" }  
};
```

بس بيقولك ان الكلاس ده بيكون read only ومانقدرش تغير القيم اللي فيه يعني تكتبه مره وتنسي انك تعرف تعدل عليه بعدين

## C# - Anonymous Type

In C#, an anonymous type is a type (class) without any name that can contain public read-only properties only. It cannot contain other members, such as fields, methods, events, etc.

see the code

```
//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud  
  
using System;  
  
namespace Main  
{  
    internal class Program  
    {
```

```

static void Main(string[] args)
{
    //you dont specify any type here , automatically will be specified
    var student = new { Id = 20, FirstName = "Mohammed", LastName = "Abu-Hadhoud" };

    Console.WriteLine("\nExample1:\n");
    Console.WriteLine(student.Id); //output: 20
    Console.WriteLine(student.FirstName); //output: Mohammed
    Console.WriteLine(student.LastName); //output: Abu-Hadhoud

    //You can print like this:
    Console.WriteLine(student);

    //anonymous types are read-only
    //you cannot change the values of properties as they are read-only.

    // student.Id = 2;//Error: cannot chage value
    // student.FirstName = "Ali";//Error: cannot chage value

    //An anonymous type's property can include another anonymous type.
    var student2 = new
    {
        Id = 20,
        FirstName = "Mohammed",
        LastName = "Abu-Hadhoud",
        Address = new { Id = 1, City = "Amman", Country = "Jordan" }
    };

    Console.WriteLine("\nExample2:\n");
    Console.WriteLine(student2.Id);
    Console.WriteLine(student2.FirstName);
    Console.WriteLine(student2.LastName);

    Console.WriteLine(student2.Address.Id);
    Console.WriteLine(student2.Address.City);
    Console.WriteLine(student2.Address.Country);
    Console.WriteLine(student2.Address);

    Console.ReadKey();
}

}
}

```

## Structures

ال **structure** هنا بتعمله زي ما كنت بتعمله في ال **C++** بس بتزود قبل المتغيرات اللي فيه كلمة **public**

فيه فرق بسيط انك وانت بتعرف متغير من النوع **struct** تقدر تعرفه عادي وتقدر تقول **=new** وبعدها اسم ال **struct**

الفرق بين الطريقتين انك لو ماعملتش new يبقى لازم تعين قيم لكل العناصر اللي جوه ال structure والا هيديك erorr

وان كلمة new دي مش معناها هنا انك بتعمل pointer لا دي معناها انك بتسدعي ال datatype عشان يحط القيم ال defaults كل constructor

```
//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        struct stStudent
        {
            public string FirstName;
            public string LastName;
        }

        static void Main(string[] args)
        {

            //A struct object can be created with or without the new operator,
            //same as primitive type variables.

            stStudent Student;
            stStudent Student2 = new stStudent();

            Student.FirstName = "Mohammed";
            Student.LastName = "Abu-Hadhoud";

            Console.WriteLine(Student.FirstName);
            Console.WriteLine(Student.LastName);

            Student2.FirstName = "Ali";
            Student2.LastName = "Ahmed";

            Console.WriteLine(Student2.FirstName);
            Console.WriteLine(Student2.LastName);

            Console.ReadKey();
        }
    }
}
```

## Dynamic Type

فكرة ال dynamic type انك مش عايز ال compiler check عال datatype وطريقة انك بدل ما تكتب int او float لا بتكتب dynamic

وبعد كده بقى مشكلة تحديد ال data type دي بتتم في ال runtime

## C# - Dynamic Types

C# 4.0 (.NET 4.5) introduced a new type called `dynamic` that avoids compile-time type checking. A `dynamic` type escapes type checking at compile-time; instead, it resolves type at run time.

A dynamic type variables are defined using the `dynamic` keyword.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            dynamic MyDynamicVar = 100;
            Console.WriteLine("Value: {0}, Type: {1}", MyDynamicVar, MyDynamicVar.GetType());

            MyDynamicVar = "Hello World!";
            Console.WriteLine("Value: {0}, Type: {1}", MyDynamicVar, MyDynamicVar.GetType());

            MyDynamicVar = true;
            Console.WriteLine("Value: {0}, Type: {1}", MyDynamicVar, MyDynamicVar.GetType());

            MyDynamicVar = DateTime.Now;
            Console.WriteLine("Value: {0}, Type: {1}", MyDynamicVar, MyDynamicVar.GetType());

            Console.ReadKey();
        }
    }
}
```

### Set Date & Time

هنا بيقولك انه فيه `struct` جاهز اسمه `dateTime` وده عشان تقدر تتعامل مع التاريخ والوقت بسهولة وبتعرفه عن طريق انك تستدعي ال `constructor` بكلمه `new` وبعدين بيعرفك باقل واعلي قيمه ممكن يخزنها

# Working with Date and Time in C#

C# includes `DateTime` struct to work with dates and times.

To work with date and time in C#, create an object of the `DateTime` struct using the new keyword. The following creates a `DateTime` object with the default value.

```
DateTime dt = new DateTime(); // assigns default value 01/01/0001 00:00:00
```

The default and the lowest value of a `DateTime` object is January 1, 0001 00:00:00 (midnight). The maximum value can be December 31, 9999 11:59:59 P.M.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            //assigns default value 01/01/0001 00:00:00
            DateTime dt1 = new DateTime();

            //assigns year, month, day
            DateTime dt2 = new DateTime(2023, 12, 31);

            //assigns year, month, day, hour, min, seconds
            DateTime dt3 = new DateTime(2023, 12, 31, 5, 10, 20);

            //assigns year, month, day, hour, min, seconds, UTC timezone
            DateTime dt4 = new DateTime(2023, 12, 31, 5, 10, 20, DateTimeKind.Utc);

            Console.WriteLine(dt1);
            Console.WriteLine(dt2);
            Console.WriteLine(dt3);
            Console.WriteLine(dt4);

            Console.ReadKey();
        }
    }
}
```

**Get Current Datetime**

عشان تجيب التاريخ والوقت بتوع دلوقتي عملوك `object` من النوع `now` اسمه موجود في ال `datetime`

```
//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            //assigns default value 01/01/0001 00:00:00
            DateTime dt1 = new DateTime();

            //assigns year, month, day
            DateTime dt2 = new DateTime(2023, 12, 31);

            //assigns year, month, day, hour, min, seconds
            DateTime dt3 = new DateTime(2023, 12, 31, 5, 10, 20);

            //assigns year, month, day, hour, min, seconds, UTC timezone
            DateTime dt4 = new DateTime(2023, 12, 31, 5, 10, 20, DateTimeKind.Utc);

            Console.WriteLine(dt1);
            Console.WriteLine(dt2);
            Console.WriteLine(dt3);
            Console.WriteLine(dt4);

            Console.ReadKey();
        }
    }
}
```

## Ticks

تعرف المل ثانية او ال `millisecond` ؟

دي عباره انك تجيب الثانية تقسمها 1000 جزء تطلعاك مل ثانية

هنا بيقولك ان النانو ثانية او ال `Nano second` هيا عباره عن مليار جزء من الثانية يعني هات الثانية وقسمها مليار مره تطلعاك النانو ثانية

فيه بقى عندك حاجه اسمها `tick` ودي بتساوي 100 نانو ثانية

وان المل ثانية فيها 10,000 tick

وان الثانية الواحدة فيها 10 مليون tick

ودي بيتم استخدامها في تمثيل التاريخ والوقت من اول 1 يناير سنة 1 ميلاديا الساعه 12 منتصف الليل في التقويم الميلادي

## Ticks

Ticks is a date and time expressed in the number of 100-nanosecond intervals that have elapsed since January 1, 0001, at 00:00:00.000 in the Gregorian calendar.

A single tick represents one hundred nanoseconds or one ten-millionth of a second.

There are 10,000 ticks in a millisecond and **10 million ticks in a second**.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            //number of 100-nanosecond intervals that have elapsed
            //since January 1, 0001, at 00:00:00.000 in the Gregorian calendar.

            DateTime dt = new DateTime();
            Console.WriteLine(DateTime.MinValue.Ticks); //min value of ticks
            Console.WriteLine(DateTime.MaxValue.Ticks); // max value of ticks

            Console.ReadKey();

        }
    }
}
```

## Datetime Static Fields

بيوريك بعض الحاجات اللي ممكن تستدعيها من ال `dateTime` ومنها انك تجيب التوقيت العالمي **UTC** واسمه

## DateTime Static Fields

The `DateTime` struct includes static fields, properties, and methods. The following example demonstrates important static fields and properties.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;
```

```

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            DateTime currentDateTime = DateTime.Now; //returns current date and time
            DateTime todaysDate = DateTime.Today; // returns today's date
            DateTime currentDateTimeUTC = DateTime.UtcNow; // returns current UTC date and time
            DateTime maxDateTimeValue = DateTime.MaxValue; // returns max value of DateTime
            DateTime minDateTimeValue = DateTime.MinValue; // returns min value of DateTime

            Console.WriteLine("currentDateTime: " + currentDateTime);
            Console.WriteLine("Today: " + todaysDate);
            Console.WriteLine("currentDateTimeUTC: " + currentDateTimeUTC);
            Console.WriteLine("minDateTimeValue: " + minDateTimeValue);
            Console.WriteLine("maxDateTimeValue: " + maxDateTimeValue);

            Console.ReadKey();
        }
    }
}

```

## Time Span

هنا بيقولك انه فيه حاجه اسمها time span دي بتمثل الوقت وال constructor بيأخذ عدد الأيام وال ساعات وال دقائق وال ثواني وال مل ثانية كمان

يبينفعك في انك مثلًا عندك تاريخ و وقت و عايز تزود عليه مده معينه شبه حوار انك عاوز تزود التاريخ بب يومين مثلًا

## TimeSpan

TimeSpan is a struct that is used to represent time in days, hour, minutes, seconds, and milliseconds.

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            DateTime dt = new DateTime(2023, 2, 21);

            // Hours, Minutes, Seconds
            TimeSpan ts = new TimeSpan(49, 25, 34);
        }
    }
}

```

```

Console.WriteLine(ts);
Console.WriteLine(ts.Days);
Console.WriteLine(ts.Hours);
Console.WriteLine(ts.Minutes);
Console.WriteLine(ts.Seconds);

//this will add time span to the date.
DateTime newDate = dt.Add(ts);

Console.WriteLine(newDate);

Console.ReadKey();
}

}
}

```

## Subtraction of two dates results in TimeSpan

هذا بيقولك انه من استخدامات ال time span انك تخزن فيه الفرق بين تاريخين

### Subtraction of two dates results in TimeSpan

```

DateTime dt1 = new DateTime(2023, 2, 21);
DateTime dt2 = new DateTime(2023, 2, 25);
TimeSpan result = dt2.Subtract(dt1);

```

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

```

```

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            DateTime dt1 = new DateTime(2023, 2, 21);
            DateTime dt2 = new DateTime(2023, 2, 25);
            TimeSpan result = dt2.Subtract(dt1);

            Console.WriteLine(result.Days);

            Console.ReadKey();
        }
    }
}

```

## Operators

هنا من مميزات ال struct بتابع التاريخ والوقت انك تقدر مثلا تقول date1-date2 على طول من غير ما تستخدم اي function وطبعا باقي ال operators زي ال + و - و \* و < و > وهكذا

## Operators

The `DateTime` struct overloads `+, -, ==, !=, >, <, <=, >=` operators to ease out addition, subtraction, and comparison of dates. These make it easy to work with dates.

```
//ProgrammingAdvises.com  
//Mohammed Abu-Hadhoud
```

```
using System;  
  
namespace Main  
{  
    internal class Program  
    {  
  
        static void Main(string[] args)  
        {  
  
            DateTime dt1 = new DateTime(2015, 12, 20);  
            DateTime dt2 = new DateTime(2016, 12, 31, 5, 10, 20);  
            TimeSpan time = new TimeSpan(10, 5, 25, 50);  
  
            Console.WriteLine(dt2 + time); // 1/10/2017 10:36:10 AM  
            Console.WriteLine(dt2 - dt1); //377.05:10:20  
            Console.WriteLine(dt1 == dt2); //False  
            Console.WriteLine(dt1 != dt2); //True  
            Console.WriteLine(dt1 > dt2); //False  
            Console.WriteLine(dt1 < dt2); //True  
            Console.WriteLine(dt1 >= dt2); //False  
            Console.WriteLine(dt1 <= dt2); //True  
  
            Console.ReadKey();  
        }  
    }  
}
```

## Convert String to DateTime

هنا بيقولك لو عندك string ومخزن فيه تاريخ تقدر تحوله ل datetime عن طريق اكتر من زي method

`Parse()`, `ParseExact()`, `TryParse()` , `TryParseExact()`

بس ال `parse exact` وال `parse` exception بيرموا مش مكتوب فيه اتاريج بشكل صحيح عشان كده بيقولك من الأفضل انك تستخد `try parse exact` او `try parse` عشان دول بيرجعوك `bool`

## Convert String to DateTime

A valid date and time string can be converted to a `DateTime` object using `Parse()`, `ParseExact()`, `TryParse()` and `TryParseExact()` methods.

The `Parse()` and `ParseExact()` methods will **throw an exception** if the specified string is not a valid representation of a date and time. So, it's recommended to use `TryParse()` or `TryParseExact()` method because they return `false` if a string is not valid.

```
//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            string str = "6/12/2023";
            DateTime dt;

            bool isValidDate = DateTime.TryParse(str, out dt);

            if (isValidDate)
                Console.WriteLine(dt);
            else
                Console.WriteLine($"{str} is not a valid date string");

            //invalid string date
            var str2 = "6/65/2023";
            DateTime dt2;

            var isValidDate2 = DateTime.TryParse(str2, out dt2);

            if (isValidDate2)
                Console.WriteLine(dt2);
            else
                Console.WriteLine($"{str2} is not a valid date string");


            Console.ReadKey();
        }
    }
}
```

في جمله الطباعه هتلaci الحته دي  `${str}` هوa هنا بدل ماي عمل `format` للنص حط علامه الدولار عشان يحط المتغير علي طول ولو شيلت الدولار هيطبع اسم المتغير مش القيمه اللي فيه

### Quick Overview

فاكر المكتبه اللي عملناها علي ال `string` وحطينا فيها `functions` كتير؟

## C# Strings

In C#, a string is a series of characters that is used to represent text. It can be a character, a word or a long passage surrounded with the double quotes " .

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            string S1 = "Mohammed Abu-Hadhoud";

            Console.WriteLine(S1.Length);

            //this will take 5 characters staring position 2
            Console.WriteLine(S1.Substring(2, 5));
            Console.WriteLine(S1.ToLower());
            Console.WriteLine(S1.ToUpper());
            Console.WriteLine(S1[2]);
            Console.WriteLine(S1.Insert(3, "KKKK"));
            Console.WriteLine(S1.Replace("m", "*"));
            Console.WriteLine(S1.IndexOf("m"));
            Console.WriteLine(S1.Contains("m"));
            Console.WriteLine(S1.Contains("x"));
            Console.WriteLine(S1.LastIndexOf("m"));

            string S2 = "Ali,Ahmed,Khalid";

            string[] NamesList = S2.Split(',');

            Console.WriteLine(NamesList[0]);
            Console.WriteLine(NamesList[1]);
            Console.WriteLine(NamesList[2]);

            string S3 = " Abu-Hadhoud ";
            Console.WriteLine(S3.Trim());
            Console.WriteLine(S3.TrimStart());
            Console.WriteLine(S3.TrimEnd());

            Console.ReadKey();

        }
    }
}
```

### String Interpolation

زي ماقولتك قبل كده انك ممكن تدخل قيمة متغير في وسط النص شبه ماكنا بنعمل في ال بس بدل كنا بنكتب ارقام وبعدين اسمي المتغيرات لا ممكن تحط علامة الدولار وتفتح قوسين تحط فيه المتغير

## String Interpolation

String interpolation is a better way of concatenating strings. We use + sign to concatenate string variables with static strings.

C# 6 includes a special character \$ to identify an interpolated string. An interpolated string is a mixture of static string and string variable where string variables should be in {} brackets.

```
//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            // String Interpolation

            string firstName = "Mohammed";
            string lastName = "Abu-Hadhoud";
            string code = "107";

            //You shold use $ to $ to identify an interpolated string
            string fullName = $"Mr. {firstName} {lastName}, Code: {code}";

            Console.WriteLine(fullName);

            Console.ReadKey();

        }
    }
}
```

## Casting Types

أنواع ال casting زيها زي ال c++ نوعين

- **Implicit casting** : ودي بتسخدمها لما تحول من متغير السعه التخزينيه ليه قليله لمتغير السعه التخزينيه ليه اعلي

- **Explicit casting** : ودي لو هتحول من متغير السعه بتاعته اكبر لمتغير السعه بتاعته اقل وفيها بتصرح بالنوع اللي عايز تحوله

## C# Type Casting

Type casting is when you assign a value of one data type to another type.

In C#, there are two types of casting:

- **Implicit Casting** (automatically) - converting a smaller type to a larger type size

`char -> int -> long -> float -> double`

- **Explicit Casting** (manually) - converting a larger type to a smaller size type

`double -> float -> long -> int -> char`

### Implicit Casting

ال implicit casting زي ماقولنا بيشتغل توماتيكي

## Implicit Casting

Implicit casting is done automatically when passing a smaller size type to a larger size type

```
//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            int myInt = 17;
            double myDouble = myInt;      // Automatic casting: int to double

            Console.WriteLine(myInt);     // Outputs 17
            Console.WriteLine(myDouble);  // Outputs 17

            Console.ReadKey();
        }
    }
}
```

### Explicit Casting

في ال explicit casting زي ماقولنا لازم تصرح بالنوع اللي عايز تحوله

## Explicit Casting

Explicit casting must be done manually by placing the type in parentheses in front of the value.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {


            double myDouble = 17.58;
            int myInt = (int)myDouble; // Manual casting: double to int

            Console.WriteLine(myDouble); // Outputs 17.58
            Console.WriteLine(myInt); // Outputs 17


            Console.ReadKey();
        }
    }
}
```

## Type Conversion Methods

تقدر تستخدم methods عشان تحول من نوع لنوع

## Type Conversion Methods

It is also possible to convert data types explicitly by using built-in methods, such

as `Convert.ToBoolean` , `Convert.ToDouble` , `Convert.ToString` , `Convert.ToInt32 (int)`  
and `Convert.ToInt64 (long)`

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {
```

```

int myInt = 20;
double myDouble = 7.25;
bool myBool = true;

Console.WriteLine(Convert.ToString(myInt)); // convert int to string
Console.WriteLine(Convert.ToDouble(myInt)); // convert int to double
Console.WriteLine(Convert.ToInt32(myDouble)); // convert double to int
Console.WriteLine(Convert.ToString(myBool)); // convert bool to string

Console.ReadKey();
}

}
}

```

## Casting Enums

هذا بيقولك تقدر تحول من ال enum لرقم والعكس

### Enums Conversion

Explicit casting is required to convert from an enum type to its underlying integral type.

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        enum WeekDays
        {
            Monday,
            Tuesday,
            Wednesday,
            Thursday,
            Friday,
            Saturday,
            Sunday
        }

        static void Main(string[] args)
        {

            Console.WriteLine(WeekDays.Friday); //output: Friday
            int day = (int)WeekDays.Friday; // enum to int conversion
            Console.WriteLine(day); //output: 4

            var wd = (WeekDays)5; // int to enum conversion
            Console.WriteLine(wd); //output: Saturday

            Console.ReadKey();
        }
    }
}

```

```
    }  
}  
}
```

## .ReadLine();

هنا بيقولك ان ال `readline` دي مش زي ال `pause` لا أصلا معموله عشان تقرأ بيهها داتا زي ال `cin` كده ودائما بتقرأ `string` يعني لازم تحول لرقم لو هتستخدمه في ادخال رقم

## Get User Input

You have already learned that `Console.WriteLine()` is used to output (print) values, for input we use `Console.ReadLine()`

Equivalent to `cin>>` in C++

**Important:** `Console.ReadLine()` always reads string.

```
//ProgrammingAdvises.com  
//Mohammed Abu-Hadhoud  
  
using System;  
  
namespace Main  
{  
    internal class Program  
    {  
  
        static void Main(string[] args)  
        {  
  
            // Type your username and press enter  
            Console.WriteLine("Enter username?");  
  
            string userName = Console.ReadLine();  
            Console.WriteLine("Username is: " + userName);  
  
            Console.ReadKey();  
  
        }  
    }  
}
```

## User Input and Numbers

زي ماقولتك انك لو هتاخد `input` من اليوزر وكان ال `input` ده رقم لازم تعمله تحويل

## User Input and Numbers

The `Console.ReadLine()` method returns a `string`. Therefore, you cannot get information from another data type, such as `int`.

therefore you should use casting when you read.

```

//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            Console.WriteLine("Enter your age?");
            //if you dont convert you will get error, and if you enter string you will get error
            int age = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Your age is: " + age);

            Console.ReadKey();
        }
    }
}

```

## Assignment Operators

هنا بيقولك مش بس تقدر تعمل **asignmet** للجمع والطرح بس لا فيه حاجات تانية  
زي دول:-

- ودي بتخزن باقي القسمه **Modulo assignment -1**  
- سواء **OR** او **BITWISE -2**

- الاس :- اللي هو الرقم مضروب في نفسه عدد من المرات زي 2 اس 3 يساوي 8

- ال **left SHIFT -4** :- وده بيمسك كل bit وبيرحله للشمال بعدد الخانات اللي انت كاتبها  
مثال :-

```

int a = 123;

Console.WriteLine(a);

a <<= 3;
Console.WriteLine(a);

```

هنا انت كانك بتأخذ القيمه اللي هيا 123 وبتضربها في 2 اس الرقم اللي هو 3  
ف 2 اس 3 ب 8 وبكده يبقى  $984 = 8 * 123$

- ودي بتمسك كل bit من الرقم ويتراحله لليمين بعدد الخانات اللي انت عاوزها  
مثال :-

```

int a = 123;

Console.WriteLine(a);

a >>= 2;
1- Console.WriteLine(a);

```

هنا انت كانك بتاخذ القيمه اللي هيا 123 و بتقسمها على 2 اس الرقم اللي هو 2 ف 2 اس 2 يساوي 4 وبكده بيبقى  $4/123 = 30.75$  وطبعا عشان المتغير من النوع int مش هياخد الكسور

-6 Lambda Operator : ودي بيستخدموها عشان يختصرروا الاكواد او العمليات اللي محتاجه ده اللي فهمته شوف الفيديو ده  
<https://youtu.be/GSr7BpoMKBg>

### C# Compound Assignment Operators

Operator	Operator Name	Example	Equivalent To
<code>+=</code>	Addition Assignment	<code>x += 5</code>	<code>x = x + 5</code>
<code>-=</code>	Subtraction Assignment	<code>x -= 5</code>	<code>x = x - 5</code>
<code>*=</code>	Multiplication Assignment	<code>x *= 5</code>	<code>x = x * 5</code>
<code>/=</code>	Division Assignment	<code>x /= 5</code>	<code>x = x / 5</code>
<code>%=</code>	Modulo Assignment	<code>x %= 5</code>	<code>x = x % 5</code>
<code>&amp;=</code>	Bitwise AND Assignment	<code>x &amp;= 5</code>	<code>x = x &amp; 5</code>
<code> =</code>	Bitwise OR Assignment	<code>x  = 5</code>	<code>x = x   5</code>
<code>^=</code>	Bitwise XOR Assignment	<code>x ^= 5</code>	<code>x = x ^ 5</code>
<code>&lt;&lt;=</code>	Left Shift Assignment	<code>x &lt;&lt;= 5</code>	<code>x = x &lt;&lt; 5</code>
<code>&gt;&gt;=</code>	Right Shift Assignment	<code>x &gt;&gt;= 5</code>	<code>x = x &gt;&gt; 5</code>
<code>=&gt;</code>	Lambda Operator	<code>x =&gt; x*x</code>	Returns <code>x*x</code>

## Arithmetic Operators

### Arithmetic Operators

Arithmetic operators are used to perform arithmetic operations such as addition, subtraction, multiplication, division, etc.

```
int x = 5;
int y = 10;
int z = x + y; // z = 15
```

## C# Arithmetic Operators

Operator	Operator Name	Example
+	Addition Operator	$6 + 3$ evaluates to 9
-	Subtraction Operator	$10 - 6$ evaluates to 4
*	Multiplication Operator	$4 * 2$ evaluates to 8
/	Division Operator	$10 / 5$ evaluates to 2
%	Modulo Operator (Remainder)	$16 \% 3$ evaluates to 1

## Relational Operators

برضه مش محتاج شرح

### Relational Operators

Relational operators are used to check the relationship between two operands. If the relationship is true the result will be `true`, otherwise it will result in `false`.

Relational operators are used in decision making and loops.

## C# Relational Operators

Operator	Operator Name	Example
<code>==</code>	Equal to	$6 == 4$ evaluates to false
<code>&gt;</code>	Greater than	$3 > -1$ evaluates to true
<code>&lt;</code>	Less than	$5 < 3$ evaluates to false
<code>&gt;=</code>	Greater than or equal to	$4 >= 4$ evaluates to true
<code>&lt;=</code>	Less than or equal to	$5 <= 3$ evaluates to false
<code>!=</code>	Not equal to	$10 != 2$ evaluates to true

## Logical Operators

## Logical Operators

Logical operators are used to perform logical operation such as `and`, `or`. Logical operators operates on boolean expressions (`true` and `false`) and returns boolean values. Logical operators are used in decision making and loops.

C# Logical operators			
Operand 1	Operand 2	OR (  )	AND (&&)
true	true	true	true
true	false	true	false
false	true	true	false
false	false	false	false

`&&` : for And

`||` : for Or

`!` : for NOT (reverse the results)

## Unary Operators

فاکر لما کنا بنقول `+x++` و `-x-` فيه غير هم بقى

+ لوحظته جنب الرقم مش بيعمله حاجه

- كانك بتضرب في سالب واحد

! ودي تغير ال `true` تخليةا `false` والعكس

## Unary Operators

Unlike other operators, the unary operators operates on a single operand.

## C# unary operators

Operator	Operator Name	Description
+	Unary Plus	Leaves the sign of operand as it is
-	Unary Minus	Inverts the sign of operand
++	Increment	Increment value by 1
--	Decrement	Decrement value by 1
!	Logical Negation (Not)	Inverts the value of a boolean

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud
```

```
using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            int number = 10, result;
            bool flag = true;

            result = +number;
            Console.WriteLine("+number = " + result);

            result = -number;
            Console.WriteLine("-number = " + result);

            result = ++number;
            Console.WriteLine("++number = " + result);

            result = --number;
            Console.WriteLine("--number = " + result);

            Console.WriteLine("!flag = " + (!flag));

            Console.WriteLine((number++));
            Console.WriteLine((number));

            Console.WriteLine((++number));
            Console.WriteLine((number));

            Console.ReadKey();
        }
    }
}
```

```
}
```

## Ternary Operator

هنا بيعرفك على ال short hand if

### Ternary Operator

The ternary operator `? :` operates on three operands. It is a shorthand for `if-then-else` statement. Ternary operator can be used as follows:

```
variable = Condition? Expression1 : Expression2;
```

The ternary operator works as follows:

- If the expression stated by Condition is `true`, the result of Expression1 is assigned to variable.
- If it is `false`, the result of Expression2 is assigned to variable.

```
//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud
```

```
using System;  
  
namespace Main  
{  
    internal class Program  
    {  
  
        static void Main(string[] args)  
        {  
  
            int number = 12;  
            string result;  
  
            result = (number % 2 == 0) ? "Even Number" : "Odd Number";  
            Console.WriteLine("{0} is {1}", number, result);  
  
            Console.ReadKey();  
        }  
    }  
}
```

## Bitwise and Bit Shift Operators

هذا الكلام ده عارفينه بس فيه حاجة زياده وهيا ال xor او ال exclusive bitwise or ودي بترجع true في حالة واحدة وهيا انه يكون bit واحد من الاثنين ب true يعني لو الاثنين ب true او الاثنين ب false هايديك يعني لو الاثنين ب true او الاثنين ب false

## Bitwise and Bit Shift Operators

Bitwise and bit shift operators are used to perform bit manipulation operations.

C# Bitwise and Bit Shift operators	
Operator	Operator Name
-	Bitwise Complement
&	Bitwise AND
	Bitwise OR
^	Bitwise Exclusive OR
<<	Bitwise Left Shift
>>	Bitwise Right Shift

## **if, if...else, if...else if and Nested if Statement**

هنا ماتقعدش تقرأ كتير ال C# في ال if هيا نفسها بتعات ال C++

## C# if (if-then) Statement

C# if-then statement will execute a block of code if the given condition is true.

The syntax of if-then statement in C# is:

```
if (boolean-expression)
{
    // statements executed if boolean-expression is true
}
```

## C# if...else (if-then-else) Statement

The if statement in C# may have an optional else statement. The block of code inside the else statement will be executed if the expression is evaluated to **false**.

The syntax of if...else statement in C# is:

```
if (boolean-expression)
{
    // statements executed if boolean-expression is true
}
else
{
    // statements executed if boolean-expression is false
}
```

## C# if...else if (if-then-else if) Statement

When we have only one condition to test, if-then and if-then-else statement works fine. But what if we have a multiple condition to test and execute one of the many block of code.

For such case, we can use if..else if statement in C#. The syntax for if...else if statement is:

```
if (boolean-expression-1)
{
    // statements executed if boolean-expression-1 is true
}
else if (boolean-expression-2)
{
    // statements executed if boolean-expression-2 is true
}
else if (boolean-expression-3)
{
    // statements executed if boolean-expression-3 is true
}
.
.
.
else
{
    // statements executed if all above expressions are false
}
```

//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud

```
using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            int x = 10; int y = 20;

            //if then statement
            if (x == 10 && y <= 20)
            {
                Console.WriteLine("yes x = 10 and y<=20");
            }

            //if then else statement
            if (x == 11)
            {
                Console.WriteLine("yes x = 11 ");
            }
        }
    }
}
```

```

else
{
    Console.WriteLine("yes x != 11 ");
}

//if else if statement
int number = 12;

if (number < 5)
{
    Console.WriteLine("{0} is less than 5", number);
}
else if (number > 5)
{
    Console.WriteLine("{0} is greater than 5", number);
}
else
{
    Console.WriteLine("{0} is equal to 5");
}

Console.ReadKey();

}
}
}

```

## **switch Statement**

هيا نفسها بتاعت ال C++ بس هنا بتدعم انك تعملها على string

# C# switch Statement

Switch statement can be used to replace the `if...else if statement` in C#. The advantage of using switch over if...else if statement is the codes will look much cleaner and readable with switch.

Same as C++ :-) but it also support strings

The syntax of switch statement is:

```
switch (variable/expression)
{
    case value1:
        // Statements executed if expression(or variable) = value1
        break;
    case value2:
        // Statements executed if expression(or variable) = value1
        break;
    ...
    ...
    default:
        // Statements executed if no case matches
}
```

A problem with the switch statement is, when the matching value is found, it executes all statements after it until the end of switch block.

To avoid this, we use `break` statement at the end of each case. The break statement stops the program from executing non-matching statements by terminating the execution of switch statement.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            //switch statement with grouped cases

            char ch;
            Console.WriteLine("Enter a letter");
            ch = Convert.ToChar(Console.ReadLine());

            switch (Char.ToLower(ch))
            {
                case 'a':
```

```
        case 'e':
        case 'i':
        case 'o':
        case 'u':
            Console.WriteLine("Vowel");
            break;
        default:
            Console.WriteLine("Not a vowel");
            break;
    }

    Console.ReadKey();
}

}
}

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud
```

```
using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            char ch;
            Console.WriteLine("Enter a letter?");
            ch = Convert.ToChar(Console.ReadLine());

            switch (Char.ToLower(ch))
            {
                case 'a':
                    Console.WriteLine("Vowel");
                    break;
                case 'e':
                    Console.WriteLine("Vowel");
                    break;
                case 'i':
                    Console.WriteLine("Vowel");
                    break;
                case 'o':
                    Console.WriteLine("Vowel");
                    break;
                case 'u':
                    Console.WriteLine("Vowel");
                    break;
                default:
                    Console.WriteLine("Not a vowel");
                    break;
            }

            Console.ReadKey();
        }
    }
}
```

```
        }
    }
}

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            //Simple Calculator

            char op;
            double first, second, result;

            Console.WriteLine("Enter first number: ");
            first = Convert.ToDouble(Console.ReadLine());
            Console.WriteLine("Enter second number: ");
            second = Convert.ToDouble(Console.ReadLine());
            Console.WriteLine("Enter operator (+, -, *, /): ");
            op = (char)Console.Read();

            switch (op)
            {
                case '+':
                    result = first + second;
                    Console.WriteLine("{0} + {1} = {2}", first, second, result);
                    break;

                case '-':
                    result = first - second;
                    Console.WriteLine("{0} - {1} = {2}", first, second, result);
                    break;

                case '*':
                    result = first * second;
                    Console.WriteLine("{0} * {1} = {2}", first, second, result);
                    break;

                case '/':
                    result = first / second;
                    Console.WriteLine("{0} / {1} = {2}", first, second, result);
                    break;

                default:
                    Console.WriteLine("Invalid Operator");
                    break;
            }

            Console.ReadKey();
        }
    }
}
```

```
        }
    }
}
```

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud
```

```
using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {
```

```
        //Simple Calculator
```

```
        string Name = "Mohammed";
```

```
        //You can compare string as well using switch
```

```
        switch (Name.ToLower())
        {

```

```
            case "mohammed":
```

```
                Console.WriteLine("Yes Mohammed");
```

```
                break;
```

```
            case "ali":
```

```
                Console.WriteLine("Yes ALi");
```

```
                break;
```

```
            default:
```

```
                Console.WriteLine("No Name Matched!");
```

```
                break;
```

```
        }
```

```
        Console.ReadKey();
    }
}
```

## ternary (? :) Operator

مش محتاجه شرح

# C# ternary (? :) Operator

We will learn about C# ternary operator and how to use it to control the flow of program.

Ternary **operator** are a substitute for if...else statement.

## Why is it called ternary operator?

This operator takes 3 **operand**, hence called ternary operator.

The syntax of ternary operator is:

```
Condition ? Expression1 : Expression2;
```

The ternary operator works as follows:

- If the expression stated by **Condition** is **true** , the result of **Expression1** is returned by the ternary operator.
- If it is **false** , the result of **Expression2** is returned.

For example, we can replace the following code:

```
if (number % 2 == 0)
{
    isEven = true;
}
else
{
    isEven = false;
}
```

with:

```
isEven = (number % 2 == 0) ? true : false ;
```

**for loop**

هيا بقى اسال ++C

## C# for loop

Same as C++

The **for** keyword is used to create for loop in C#. The syntax for **for loop** is:

```
for (initialization; condition; iterator)
{
    // body of for loop
}
```

## C# nested for loops

The syntax for nested **for loops** is:

```
for (initialization; condition; iterator)
{
    // body of for loop
    for (initialization; condition; iterator)
    {
        // body of inner for loop
    }
}
```

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            Console.WriteLine("\nForward Loop:");
            //forward loop
            for (int i = 1; i <= 10; i++)
            {

                Console.WriteLine(i);
            }

            Console.WriteLine("\nBackword Loop:");
            //backword loop
            for (int i = 10; i >= 1; i--)
            {


            }
        }
    }
}
```

```

        Console.WriteLine(i);

    }

    Console.WriteLine("\nNested Loops:");
    //forward loop
    for (int i = 1; i <= 10; i++)
    {
        for (int j = 0; j < 10; j++)
        {
            Console.WriteLine("i={0} and j={1}", i, j);
        }
    }

    Console.ReadKey();
}
}

```

## while loop

هيا بنا اول C++

### C# while loop

The **while** keyword is used to create while loop in C#. The syntax for while loop is:

Same as C++

```

while (test-expression)
{
    // body of while
}

```

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud
```

```

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

```

```

int i = 1;
while (i <= 5)
{
    Console.WriteLine("C# while Loop: Iteration {0}", i);
    i++;
}

Console.ReadKey();
}
}
}

```

## do while loop

هيا هيا بقى

### C# do...while loop

The **do** and **while** keyword is used to create a do...while loop. It is similar to a while loop, however there is a major difference between them.

In while loop, the condition is checked before the body is executed. It is the exact opposite in do...while loop, i.e. condition is checked after the body is executed.

This is why, the body of do...while loop will execute at least once irrespective to the test-expression.

The syntax for do...while loop is:

```

do
{
    // body of do while loop
} while (test-expression);

```

//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud

```

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            int i = 1;
            do
            {
                Console.WriteLine("C# while Loop: Iteration {0}", i);
                i++;
            }
        }
    }
}

```

```
    } while (i <= 5);  
  
    Console.ReadKey();  
  
}  
}  
}
```

## **break Statement**

## هیا هیا بتاعت ال C++

## C# break Statement

In C#, we use the break statement to terminate the loop.

As we know, loops iterate over a block of code until the test expression is false. However, sometimes we may need to terminate the loop immediately without checking the test expression.

In such cases, the break statement is used. The syntax of break statement is,

**break;**

Before we learn about `break`, make sure to learn about

- for loop
  - if...else
  - while loop

//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud

using System;

namespace Main

{

{

1

```
static void Main(string[] args)
```

1

```
int i = 1;
```

do  
f

{

```
Console.WriteLine("C# while Loop: Iteration {0}", i);
```

```
if (i == 3)  
    break;
```

```
i++;  
}  
} while (i <= 5);  
  
Console.ReadKey();  
}  
}
```

## continue Statement

هيا هيا بقى

## C# continue Statement

In C#, we use the continue statement to skip a current iteration of a loop.

When our program encounters the continue statement, the program control moves to the end of the loop and executes the test condition (update statement in case of for loop).

The syntax for continue is:

```
continue;
```

Before we learn about continue, make sure to learn about

- for loop
- while loop
- if...else

```
//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud  
  
using System;  
  
namespace Main  
{  
    internal class Program  
    {  
  
        static void Main(string[] args)  
        {  
  
            for (int i = 1; i <= 5; ++i)  
            {  
  
                if (i == 3)  
                {  
                    continue;  
                }  
            }  
        }  
    }  
}
```

```

        }

        Console.WriteLine(i);
    }

    Console.ReadKey();
}

}
}

```

## Arrays Are Bound

في ال C++ كان ممكن تزود العناصر اللي في ال array هنا لا طالما حددت العناصر مره ماينفعش تعدل على عددها

**Arrays are bound, you cannot exceed their limits as you do in C++**

the following code will give you error because you tried to exceed the length of the array.

```

// create an array
int[] numbers = { 1, 2, 3 };

numbers[0] = 1;
numbers[1] = 2;
numbers[2] = 3;

numbers[3] = 4; ✘

Console.ReadKey()
}

```

The screenshot shows a C# code editor with the following code:

```

// create an array
int[] numbers = { 1, 2, 3 };

numbers[0] = 1;
numbers[1] = 2;
numbers[2] = 3;

numbers[3] = 4; ✘

Console.ReadKey()
}

```

A tooltip window titled "Exception Unhandled" appears over the line `numbers[3] = 4;` with the message: "System.IndexOutOfRangeException: 'Index was outside the bounds of the array.'". Below the message are links: "Show Call Stack", "View Details", "Copy Details", "Start Live Share session", and "Exception Settings".

## Array Declaration

بتعمل ال array زيها زي ال C++ بس عشان تحدد عدد العناصر بتقول new وبعدها نوع ال array وبعدها العدد

```

// declare an array
int[] age;

// allocate memory for array
age = new int[5];

```

```
int[] age = new int[5];
```

## C# Array Declaration

In C#, here is how we can declare an array.

```
datatype[] arrayName;
```

Here,

- `dataType` - data type like `int`, `string`, `char`, etc
- `arrayName` - it is an identifier

Let's see an example,

```
int[] age;
```

Here, we have created an array named `age`. It can store elements of `int` type.

### But how many elements can it store?

To define the number of elements that an array can hold, we have to allocate memory for the array in C#. For example,

```
// declare an array
int[] age;

// allocate memory for array
age = new int[5];
```

Here, `new int[5]` represents that the array can store 5 elements. We can also say the size/length of the array is 5.

**Note:** We can also declare and allocate the memory of an array in a single line. For example,

```
int[] age = new int[5];
```

## Array initialization

شرح محتاج

In C#, we can initialize an array during the declaration. For example,

```
int [] numbers = {1, 2, 3, 4, 5};
```

Here, we have created an array named numbers and initialized it with values **1, 2, 3, 4**, and **5** inside the curly braces.

Note that we have not provided the size of the array. In this case, the C# automatically specifies the size by counting the number of elements in the array (i.e. 5).

In an array, we use an **index number** to determine the position of each array element. We can use the index number to initialize an array in C#. For example,

```
// declare an array
int[] age = new int[5];

//initializing array
age[0] = 12;
age[1] = 4;
age[2] = 5;
...
```

**Note:**

- An array index always starts at 0. That is, the first element of an array is at index 0.
- If the size of an array is 5, the index of the last element will be at 4 (5 - 1).

## **Access Array Elements**

مش محتاجه شرح

## Access Array Elements

We can access the elements in the array using the index of the array. For example,

```
// access element at index 2  
array[2];  
  
// access element at index 4  
array[4];
```

Here,

- `array[2]` - access the 3rd element
- `array[4]` - access the 5th element

```
//ProgrammingAdvises.com  
//Mohammed Abu-Hadhoud  
  
using System;  
  
namespace Main  
{  
    internal class Program  
    {  
  
        static void Main(string[] args)  
        {  
  
            // create an array  
            int[] numbers = { 1, 2, 3 };  
  
            //access first element  
            Console.WriteLine("Element in first index : " + numbers[0]);  
  
            //access second element  
            Console.WriteLine("Element in second index : " + numbers[1]);  
  
            //access third element  
            Console.WriteLine("Element in third index : " + numbers[2]);  
  
  
            //through loop  
            Console.WriteLine("\nAccess array using loop:\n");  
            for (int i = 0; i < numbers.Length; i++)  
            {  
                Console.WriteLine("Element in index {0} : {1} ", i, numbers[i]);  
            }  
  
            Console.ReadKey();  
        }  
}
```

```
}
```

## 2 Dimensional Array

هوا نفسه بتاع ال C++ بس اختلاف بسيط وهو انك بدل ما كنت بتكتبه كده `int x[][]` لا بقيت تكتبه كده `new int x[,]` عشان تعين حجمه وطبعا بتحط ال `new`

### Two-Dimensional Array Declaration

Here's how we declare a 2D array in C#.

```
int[ , ] x = new int [2, 3];
```

Here, `x` is a two-dimensional array with **2** elements. And, each element is also an array with **3** elements.

So, all together the array can store **6** elements (**2 \* 3**).

Note: The single comma `[ , ]` represents the array is 2 dimensional.

### Two-Dimensional Array initialization

In C#, we can initialize an array during the declaration. For example,

```
int[ , ] x = { { 1, 2 ,3}, { 3, 4, 5 } };
```

Here, `x` is a 2D array with two elements `{1, 2, 3}` and `{3, 4, 5}`. We can see that each element of the array is also an array.

We can also specify the number of rows and columns during the initialization. For example,

```
int [ , ] x = new int[2, 3] { {1, 2, 3}, {3, 4, 5} };
```

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {
        static void Main(string[] args)
        {

            //initializing 2D array
            int[,] numbers = { { 12, 13 }, { 55, 77 } };

            // access first element from the first row
            Console.WriteLine("Element at index [0, 0] : " + numbers[0, 0]);
        }
    }
}
```

```

        // access first element from second row
        Console.WriteLine("Element at index [1, 0] : " + numbers[1, 0]);

        Console.ReadKey();
    }
}

```

## for each loop

هيا نفسها بتاعت ال C++ بس اصبح اسمها **for each** وبدل النقطتين : بتكتب **in**

## C# foreach loop

We will learn about foreach loops (an alternative to for loop) and how to use them with arrays and collections.

C# provides an easy to use and more readable alternative to for loop, the foreach loop when working with arrays and collections to iterate through the items of arrays/collections. The foreach loop iterates through each item, hence called foreach loop.

### Syntax of foreach loop

```

foreach (element in iterable-item)
{
    // body of foreach loop
}

```

Here iterable-item can be an array or a class of collection.

**Important : See Both Examples attached.**

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            char[] gender = { 'm', 'f', 'm', 'm', 'm', 'f', 'f', 'm', 'm', 'f' };
            int male = 0, female = 0;
            foreach (char g in gender)
            {

```

```

        if (g == 'm')
            male++;
        else if (g == 'f')
            female++;
    }
    Console.WriteLine("Number of male = {0}", male);
    Console.WriteLine("Number of female = {0}", female);

    Console.ReadKey();
}

}
}

```

//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud

```

using System;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            char[] myArray = { 'H', 'e', 'l', 'l', 'o' };

            foreach (char ch in myArray)
            {
                Console.WriteLine(ch);
            }

            Console.ReadKey();
        }
    }
}

```

## Find Minimum and Maximum Element Using System.Linq

فيه مكتبه اسمها **System.Linq** ودي بتقدر من خلالها تعمل بعض العمليات زي انك تجيب ال **min** وال **max** اللي في ال **array** وانك ترتيب العناصر وتحتار عناصر معينه وهكذا

### Array Operations using System.Linq

In C#, we have the **System.Linq** namespace that provides different methods to perform various operations in an array. For example,

#### Example: Find Minimum and Maximum Element

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

```

```

// provides us various methods to use in an array
using System.Linq;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            // Note that we used System.Linq above.

            int[] numbers = { 51, -1, 2, 14, 18, 40, 178 };

            // get the minimum element
            Console.WriteLine("Smallest Element: " + numbers.Min());

            // Max() returns the largest number in array
            Console.WriteLine("Largest Element: " + numbers.Max());

            Console.ReadKey();
        }
    }
}

```

## Count , Sum And Average Using System.Linq

المكتبه linq فيها انك تجمع عناصر ال array وتحسب ال average بقى لهم وتعدهم

### Array Operations using System.Linq

In C#, we have the `System.Linq` namespace that provides different methods to perform various operations in an array. For example,

#### Example: Count, Sum, Average

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadoud

using System;

// provides us various methods to use in an array
using System.Linq;

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            // Note that we used System.Linq above.
        }
    }
}

```

```
int[] numbers = { 20, 22, 19, 18, 1 };

// compute Count
Console.WriteLine("Count : " + numbers.Count());

// compute Sum
Console.WriteLine("Sum : " + numbers.Sum());

// compute the average
Console.WriteLine("Average: " + numbers.Average());

Console.ReadKey();

}

}

}
```

## C# Math

فاکر کلاس ال cmath اللي كان في ال C++ ؟

هنا بقى فيه واحد زي اسمه Math

# C# Math

The C# Math class has many methods that allows you to perform mathematical tasks on numbers.

## Math.Max(x,y)

The `Math.Max(x,y)` method can be used to find the highest value of  $x$  and  $y$ .

example:

```
Math.Max(5, 10);
```

## Math.Min(x,y)

The `Math.Min(x,y)` method can be used to find the lowest value of  $x$  and  $y$ .

example:

```
Math.Min(5, 10);
```

## Math.Sqrt(x)

The `Math.Sqrt(x)` method returns the square root of  $x$

*example:*

```
Math.Sqrt(64);
```

## Math.Abs(x)

The `Math.Abs(x)` method returns the absolute (positive) value of  $x$ .

*example:*

```
Math.Abs(-4.7);
```

## Math.Round()

`Math.Round()` rounds a number to the nearest whole number.

*example:*

```
Math.Round(9.99);
```

```
//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud
```

```
using System;
```

```
namespace Main
```

```
{
```

```
internal class Program
```

```
{
```

```
    static void Main(string[] args)
```

```
{
```

```
        Console.WriteLine("Max of 5, 10 is: {0}", Math.Max(5, 10));
```

```
        Console.WriteLine("Min of 5, 10 is: {0}", Math.Min(5, 10));
```

```
        Console.WriteLine("Squar Root of 64 is: {0}", Math.Sqrt(64));
```

```
        Console.WriteLine("Absolute (positive) value of -4.7 is: {0}", Math.Abs(-4.7));
```

```
        Console.WriteLine("Round of 9.99 is: {0}", Math.Round(9.99));
```

```
        Console.ReadKey();
```

```
}
```

```
}
```

## C# Methods

نفس طريقة ال C++

# C# Methods

A **method** is a block of code which only runs when it is called.

You can pass data, known as parameters, into a method.

Methods are used to perform certain actions, and they are also known as **functions**.

**C# is a fully OOP language , you cannot create a method outside class.**

### Create a Method

A method is defined with the name of the method, followed by parentheses () . C# provides some pre-defined methods, which you already are familiar with, such as `Main()` , but you can also create your own methods to perform certain actions:

Example:

```
static void MyMethod()
{
    // code to be executed
}
```

You should use **static** if you want to call the method without having object.

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void PrintMyName()
        {
            Console.WriteLine("Mohammed Abu-Hadhoud");
        }

        static void Main(string[] args)
        {

            PrintMyName();
        }
}
```

```
        Console.ReadKey();  
    }  
}
```

## Method Parameters

زی ال C++

# C# Method Parameters

## Parameters and Arguments

Information can be passed to methods as parameter. Parameters act as variables inside the method.

They are specified after the method name, inside the parentheses. You can add as many parameters as you want, just separate them with a comma.

```
//ProgrammingAdvices.com  
//Mohammed Abu-Hadhoud
```

```
using System;  
  
namespace Main  
{  
    internal class Program  
    {  
  
        static void PrintMyInfo(string Name, byte Age)  
        {  
            Console.WriteLine("Name= {0} , Age= {1}", Name, Age);  
        }  
        static void Main(string[] args)  
        {  
  
            PrintMyInfo("Mohammed-AbuHadhoud", 45);  
  
            Console.ReadKey();  
        }  
    }  
}
```

## Default Parameter Value

زی ال C++

بس لو هتعملها في متغير واحد اعملها في اخر متغير يالما تعملها في كل المتغيرات

# C# Default Parameter Value

## Default Parameter Value

You can also use a default parameter value, by using the equals sign (=).

[See the Code](#)

```
//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void PrintMyInfo(string Name, byte Age, string Address = "No Address")
        {
            Console.WriteLine("Name= {0} , Age= {1}, Address= {2}", Name, Age, Address);
        }

        static void Main(string[] args)
        {

            //First we did not provide the address it's optional
            PrintMyInfo("Mohammed-AbuHadhoud", 45);

            //second we provided the address
            PrintMyInfo("Mohammed-AbuHadhoud", 45, "Amman-Jordan");

            Console.ReadKey();

        }
    }
}
```

## Return Values

شرح محتاجه

# C# Return Values

## Return Values

In the previous lessons, we used the `void` keyword in all examples, which indicates that the method should not return a value.

If you want the method to return a value, you can use a primitive data type (such as `int` or `double`) instead of `void`, and use the `return` keyword inside the method.

//ProgrammingAdvices.com

```

//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static string GetMyName()
        {
            return "Mohammed-AbuHadhoud";

        }
        static void Main(string[] args)
        {

            Console.WriteLine("My Name is {0}", GetMyName());

            Console.ReadKey();

        }
    }
}

```

## Named Arguments

هنا بيقولك لو يتستدعي function معينه ومش عايز تديها ال parameters بالترتيب او مش عايز كل شويه ترجع تبعص عال parameter بتكتب اسم ال parameters وبعده نقطتين : وبعده القيمه

## C# Named Arguments

### Named Arguments

It is also possible to send arguments with the `key: value` syntax.

That way, the order of the arguments does not matter.

```

//ProgrammingAdvises.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static void MyMethod(string child1, string child2, string child3)
        {
            Console.WriteLine("The youngest child is: " + child3);
        }
    }
}

```

```

static void Main(string[] args)
{
    //see the order of sending parameters is not important.

    MyMethod(child3: "Omar", child1: "Saqer", child2: "Hamza");

    Console.ReadKey();

}
}

```

## Method Overloading

مش محتاجه شرح

## Method Overloading

With **method overloading**, multiple methods can have the same name with different parameters.

```

//ProgrammingAdvices.com
//Mohammed Abu-Hadhoud

using System;

namespace Main
{
    internal class Program
    {

        static int Sum(int Num1, int Num2)

        {
            return Num1 + Num2;
        }

        static int Sum(int Num1, int Num2, int Num3)

        {
            return Num1 + Num2 + Num3;
        }

        static int Sum(int Num1, int Num2, int Num3, int Num4)

        {
            return Num1 + Num2 + Num3 + Num4;
        }

        static void Main(string[] args)
    }
}

```

```

{
    //we have 3 diffrent methods but with the same name.

    Console.WriteLine(Sum(10, 20));
    Console.WriteLine(Sum(10, 20, 30));
    Console.WriteLine(Sum(10, 20, 30, 40));
    Console.ReadKey();
}

}
}
}

```

## Exceptions

زی ال C++

# C# Exceptions - Try..Catch

## C# Exceptions

When executing C# code, different errors can occur: coding errors made by the programmer, errors due to wrong input, or other unforeseeable things.

When an error occurs, C# will normally stop and generate an error message. The technical term for this is: C# will throw an **exception** (throw an error).

## C# try and catch

The **try** statement allows you to define a block of code to be tested for errors while it is being executed.

The **catch** statement allows you to define a block of code to be executed, if an error occurs in the try block.

The **try** and **catch** keywords come in pairs:

Syntax:

```

try
{
    // Block of code to try
}
catch (Exception e)
{
    // Block of code to handle errors
}

```

//ProgrammingAdvices.com  
//Mohammed Abu-Hadoud  
  
using System;

```

namespace Main
{
    internal class Program
    {

        static void Main(string[] args)
        {

            try
            {
                int[] myNumbers = { 1, 2, 3 };
                Console.WriteLine(myNumbers[10]);
            }
            catch (Exception e)
            {
                Console.WriteLine(e.Message);
            }
            Console.ReadLine();

        }
    }
}

```

## Random Function In C#

فاكير لما كنا بنعمل **function** مخصوص عشان نجيب رقم عشوائي بين رقمين ؟  
هذا الموضوع جاهز

### Generate Random Number in Min to Max Range

Use the `Next(int min, int max)` overload method to get a random integer that is within a specified range.

Example: Generate Random Integers in Range Copy

```

Random rnd = new Random();

for(int j = 0; j < 4; j++)
{
    Console.WriteLine(rnd.Next(10, 20)); // returns random integers >= 10 and < 20
}

```

## Web vs Desktop Applications



ال browser هو برنامج محظوظ في السيرفر وانت عشان تستخدمه تحتاج زي جوجل مثلا

ال office ده برنامج بتنزله عالجهاز عندك عشان تستخدمه زي ال desktop application

## Web vs Desktop Applications

Web-based applications are programs that you access online. The function is delivered to your device from a remote server when you access it via your browser— it's not installed on your machine.

Desktop applications, are the programs that you download and install on your device. Everything they need to deliver your functionality is stored on your workstation. So, you can access the application through your desktop.

Examples: Word, Excel , Browsers, Photoshop...etc.

من مميزات ال desktop application :-

1- انك تقدر تستخدمه من غير انترنت الا لو كان البرنامج بيحتاج انه يوصل لاداتا في سيرفر عالانترنت

2- الخصوصيه والأمان كل حاجه بتعملها عالانترنت متراقبه انما ده عشان offline فمفيش حد بيوصلك

3- التكافه اقل من ال web application لأنك بتعمل البرنامج وتديه للعميل يسطبه عنده وخلصنا انما لو عملته ه يكون عندك سيرفرات و backups وصيانة وحوارات web application

4- ال desktop اسرع من الويب لانه يعتمد على سرعة الانترنت وكم يوزر داخل عالسيرفر وحاجات تانية انما ال desktop هو البرنامج أصلا عندك فكل موارد الجهاز بتاعتك متاحه ليك لوحدك وكمان فيه حاجات في ال desktop ماينفعش تعمله في الويب زي ال adobe photoshop مافيش تطبيق ويب يقدر يديك نفس الأداء

-: desktop application من عيوب ال

1- لازم تنزل البرنامج عندك وبيعتمد على المواصفات بتاعت الجهاز

2- لازم يكون مساحه عشان تنزل البرنامج زي الألعاب

3- لازم تعمل ال upgrade بنفسك

المواقف واحتياجات العميل هيا اللي بتحدد مين الأفضل بالنسبة له

## Pros and Cons (Advantages and Disadvantages) of Desktop Applications

### Offline Access

Internet access is not necessary while working with a desktop application, as it is installed in the system and you are leveraged to access it any time.

Still, there can be a situation when a person cannot find a stable connection. However, a desktop app will not impact its work, and all the procedures will be executed successfully, providing the required output.

Furthermore, once the application is installed, you can use it anywhere, anytime.

## More Privacy

In terms of data security, desktop solutions are considered more precisely, as overall data is stored on your machine. Therefore, you are the only authorized person to access, read, write and share it.

In addition, whenever someone works on a web-based app, cookies are generated, browser history is saved, enabling the companies to display ads.

Moreover, your device will be secure from man-in-the-middle and other threats in offline mode, and auto-downloading of any malware will also be prevented.

## Cost-Efficient

Applications based on Software as a Service are pretty expensive than desktop solutions, as the vendor has to host and maintain it constantly.

Also, you have to pay hefty amounts for accessing the services through a web app. On the other hand, most desktop-based applications are freely available, and if any one of them needs to be purchased, the user can use it for a long-time.

In addition, if an enterprise provides its desktop application to its stakeholders, it only has to host a web page with a link, and the company can do it in a small amount. In this way, desktop apps help both end-users and organizations to save money.

# Optimized Performance

If we compare the web and desktop apps based on speed, desktop applications always will be the dominant technology.

It directly links with the device and utilizes all the required resources appropriately. Users do not generally analyze command execution, but it is faster in desktop apps, as low bandwidth doesn't impact the performance.

Besides this, a wide array of features is offered compared to web solutions, as computer systems can handle many threads simultaneously. For instance, you can manually allocate the space to software to maximize its functioning.

## Cons (Disadvantages) of Desktop Applications

### Installation is necessary

While using a web-based application, a browser and internet access are the only basic requirements.

But, to use a desktop app, installing it on the device is a must. And, for the successful installation, aspects such as Operating system, RAM, processor, GUI, ROM, and many more are analyzed.

Therefore, it makes the desktop app a little a bit expensive. Moreover, it is difficult to switch between systems to complete the work, as it is native to a single OS and can't be accessed on any other system.

# More Space Utilization

Desktop Software requires larger storage space on the machine's hard disk to deploy its necessary packages and associated files.

Also, any new file the user adds is stored on the system, utilizing more space, which loads the system.

In addition, the cost of maintenance can be increased with this if there is insufficient space on the hard disk installed on the device, as the user needs to modify the disk with a larger capacity.

## Upgraded Manually

Downloading a new file whenever an update is available is sometimes a hectic piece of work, as you have to wait for its compatibility check, installation, and then rebooting the system for successful configuration.

In addition, it also consumes more space and processing cores, which can degrade the performance due to a shortage of resources.

## Wrapping Up

## Wrapping Up

Web Application and Desktop Application both are heavily selected from small to large scale organizations, as per their need and budget.

Both these solutions have their advantages and disadvantages.

However, desktop-based software is being considered for internal use in enterprises, and web-based apps are mostly preferred for users at remote locations. It depends upon your requirements.

## Introduction About Controls

### Part 1

بص کده عالتطبيق ده

The screenshot shows a Windows application titled "Pizza Order". The main title is "MAKE YOUR PIZZA" in large red letters. On the left, there are three radio buttons for "Size": "Small", "Medium" (which is selected), and "Large". Below that is a section for "Crust Type" with two radio buttons: "Thin Crust" (selected) and "Thick Crust". In the center, there's a "Toppings" section with four checkboxes: "Extra Cheese", "Onion", "Mushrooms", "Olives", "Tomatoes", and "Green Peppers". Under "Where To Eat", there are two radio buttons: "Eat In" (selected) and "Take Out". At the bottom are two buttons: "Order Pizza" and "Reset Form". To the right, an "Order Summary" section displays: "Size: Medium", "Toppings: No Toppings", "Crust Type: Thin Crust", "Where to Eat: Eat In", and "Total Price: \$0". A cursor arrow is pointing towards the "Order Pizza" button.

الشاشة دي اسمها **form** يعني شاشه او نموذج  
ال **form** ده جواه مجموعه من ال **controls** كل واحد فيهم ليه الخصائص بتاعته  
كل واحدة من دول اسمه **radio button** بيسمحلك انك تختار واحد بس من مجموعه من الاختيارات

A close-up view of the "Crust Type" section of the form. It shows the word "Crust Type" in blue and three radio buttons: "Small" (selected), "Medium", and "Large". Below this, there are two more radio buttons: "Eat In" (selected) and "Take Out".

كل واحدة من دول اسمه **check box** وبيسمحلك انك تختار اكتر من اختيار

## Toppings

- |  |   |
|--|---|
| <input checked="" type="checkbox"/> Onion  | <input checked="" type="checkbox"/> Extra Chees |
| <input checked="" type="checkbox"/> Olives | <input checked="" type="checkbox"/> Mushrooms   |
| <input type="checkbox"/> Green Peppers     | <input checked="" type="checkbox"/> Tomatoes    |

كل واحد من دول اسمه label

# MAKE YOUR PIZZA

No Toppings

Toppings:

Size: Small

Order Summary

Eat In.

Where to Eat:

Thick Crust

Crust Type:

\$30

Total Price:

كل واحد من دول اسمه button

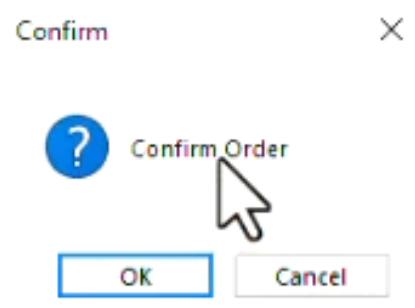
Reset Form

Order Pizza

أي اختيار بتختاره بيشغل حاجه اسمها حدث او event ده اسمه on click event

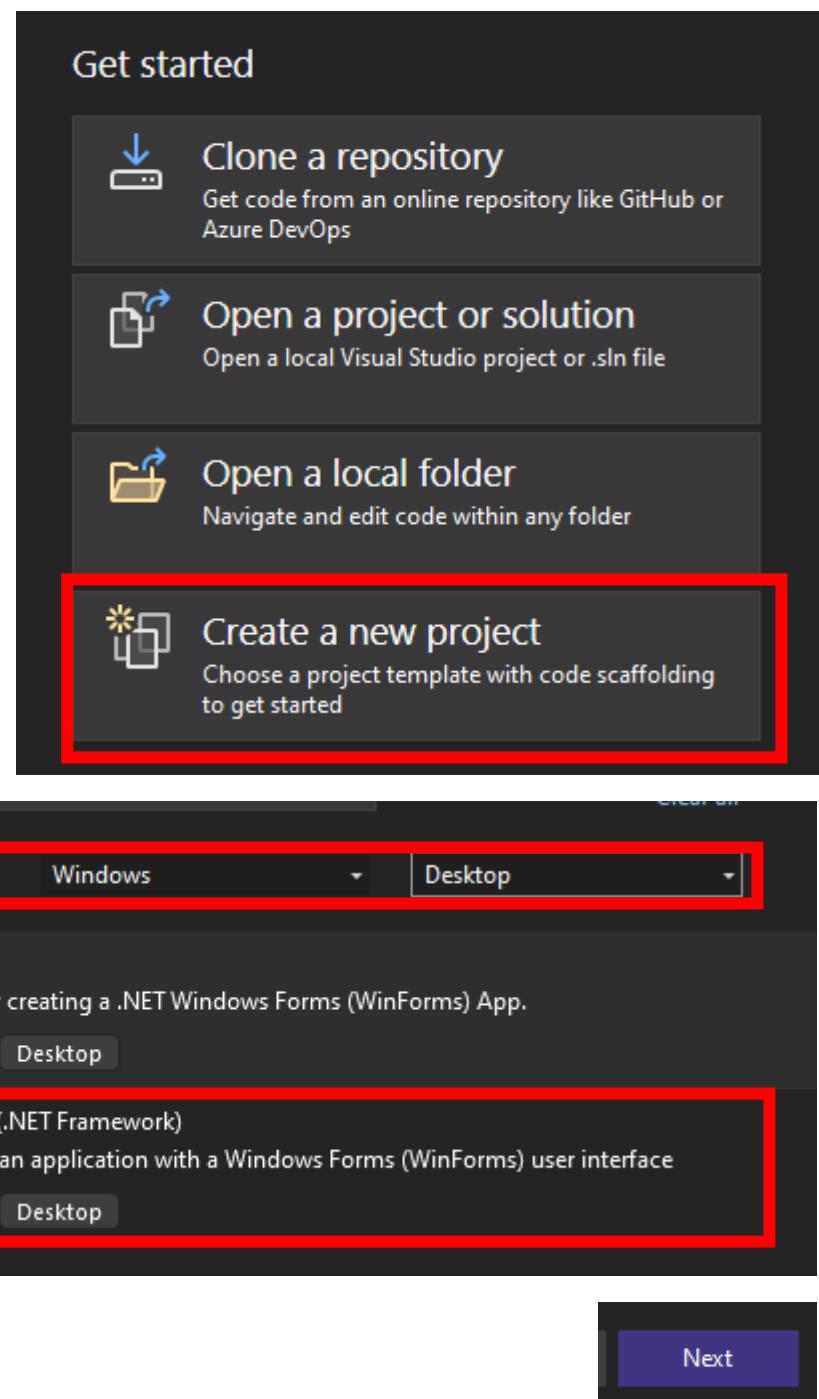
تقدير جواه تكتب الكود اللي عايزه يتنفذ

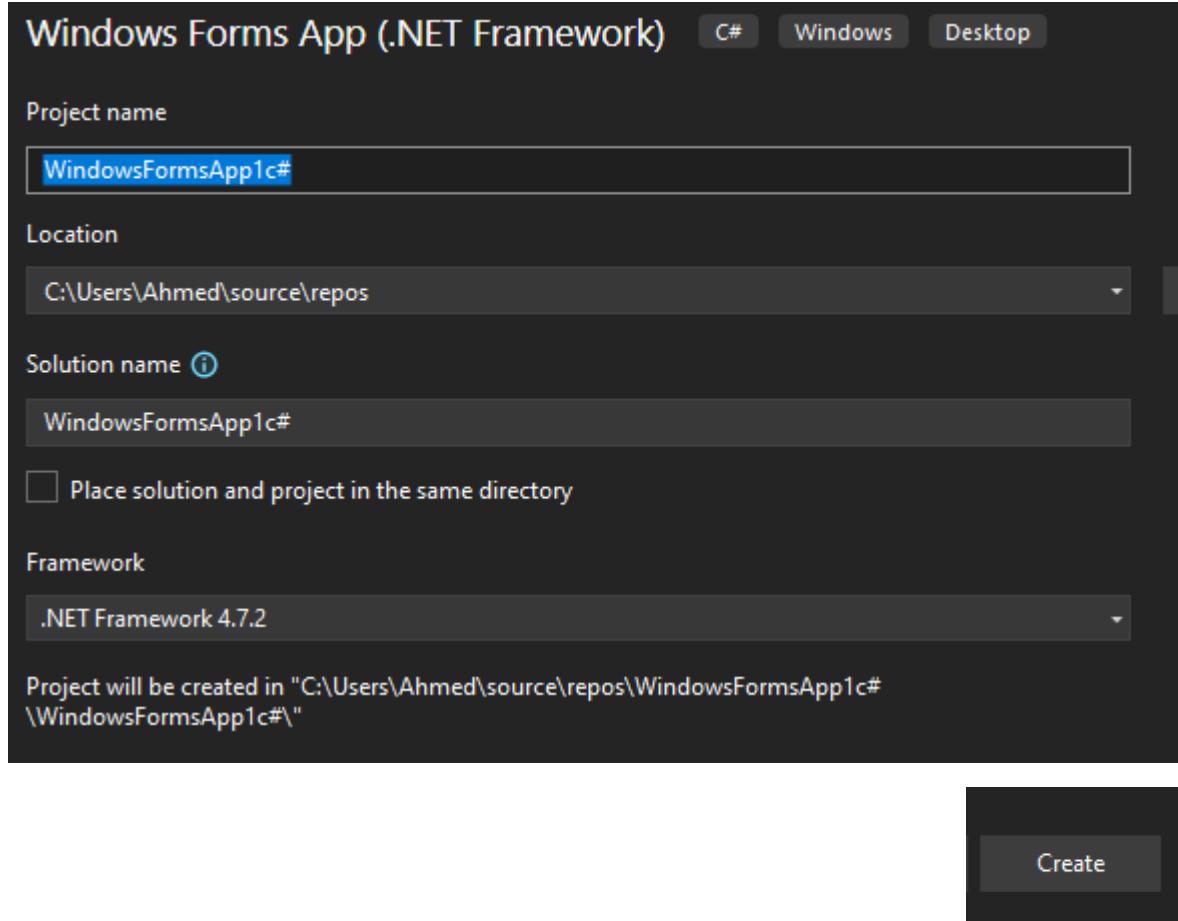
ده كده اسمه massage box بتديه البيانات اللي عايز تعرضها وبيعرضها لك



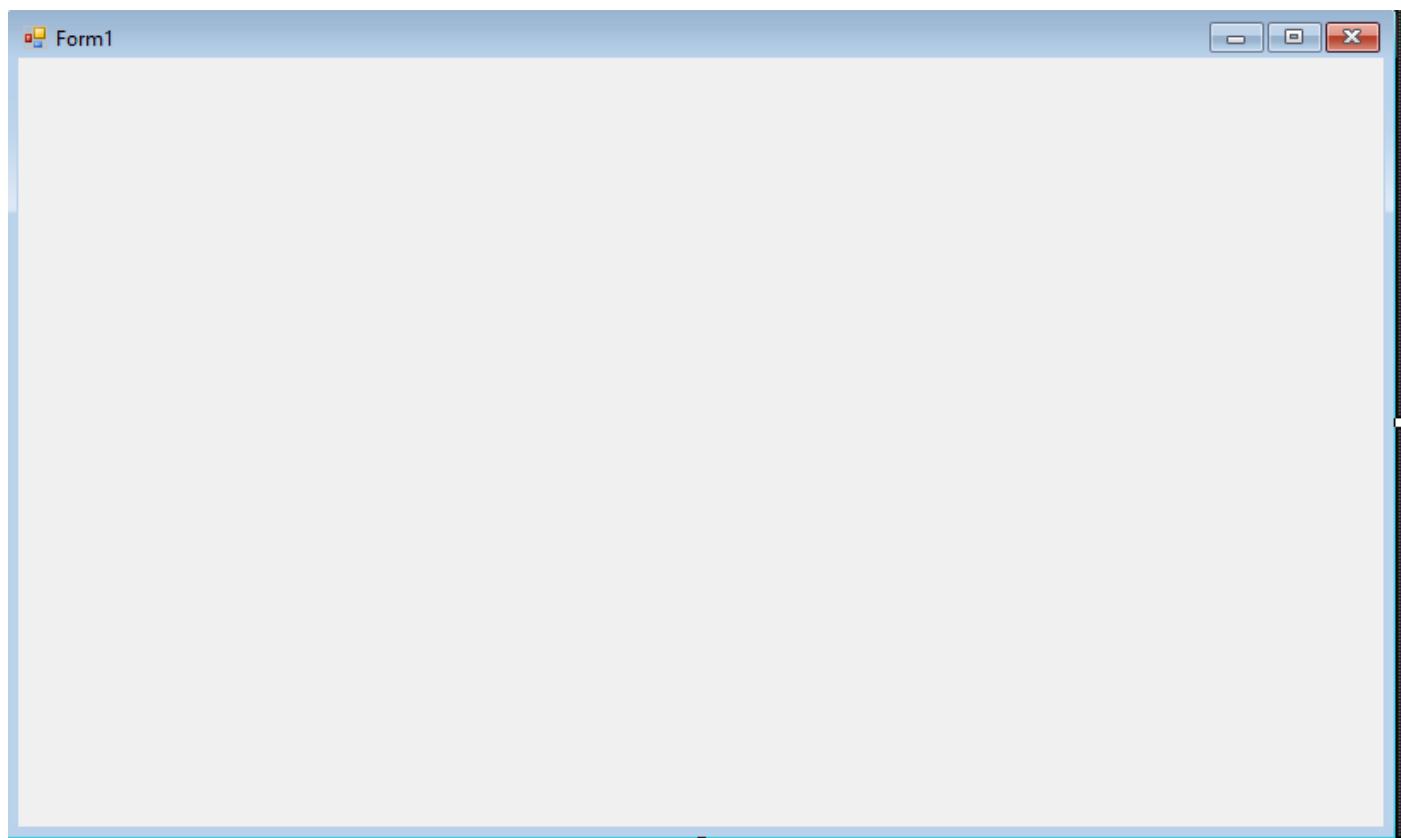
## Part 2 - First Windows Forms Application

عشان نعمل project لازم نعمل desktop application من نوع windows form application





ده كده اسمه form بتقدر تضيف فيه controls وتحكم فيها

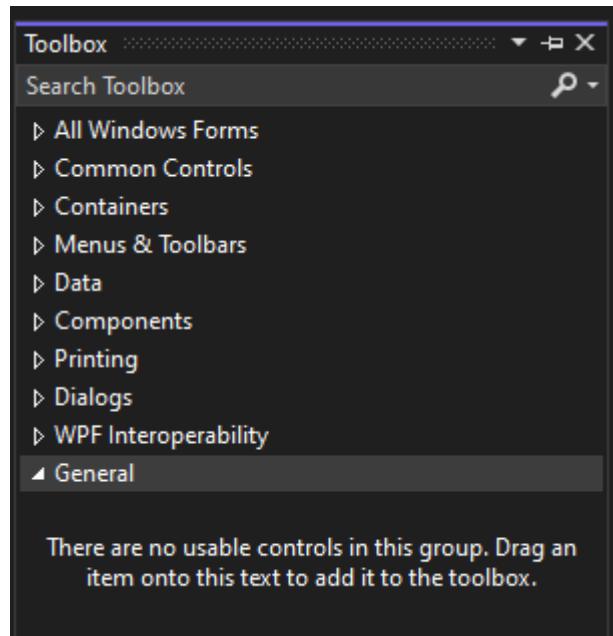


دلوقي احنا عاوزين نجيب ال controls اللي هنشتغل بيها

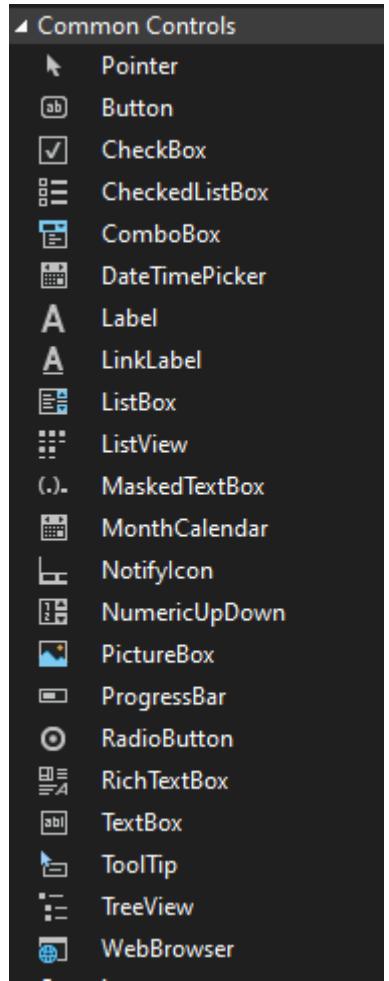
من view >> tool box او من الاختصار ctrl+alt+x

View	Git	Project	Build	Debug	Test	Ar
<> Code					Ctrl+Alt+0	
❑ Designer					Shift+F7	
❑ Solution Explorer					Ctrl+Alt+L	
❖ Git Changes					Ctrl+0, Ctrl+G	
❖ Git Repository					Ctrl+0, Ctrl+R	
❖ Team Explorer					Ctrl+\, Ctrl+M	
❖ Test Explorer					Ctrl+E, T	
❑ Bookmark Window					Ctrl+K, Ctrl+W	
❖ Call Hierarchy					Ctrl+Alt+K	
❖ Class View					Ctrl+Shift+C	
❖ Code Definition Window					Ctrl+Shift+V	
❖ Object Browser					Ctrl+Alt+J	
❖ Error List					Ctrl+\, E	
❖ Output					Alt+2	
❖ Resource View					Ctrl+Shift+E	
❖ Toolbox					Ctrl+Alt+X	

هتلacieh فتحلك القايمه بتاعت ال controls ومتقسمه



ال حاجات اللي بتسخدمها في كل مشروع common controls

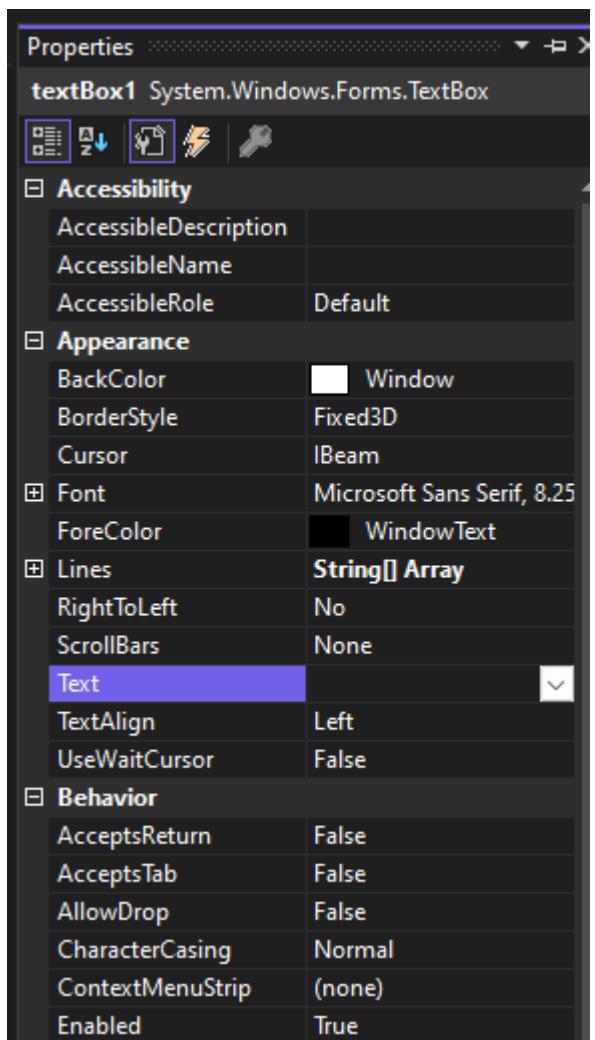
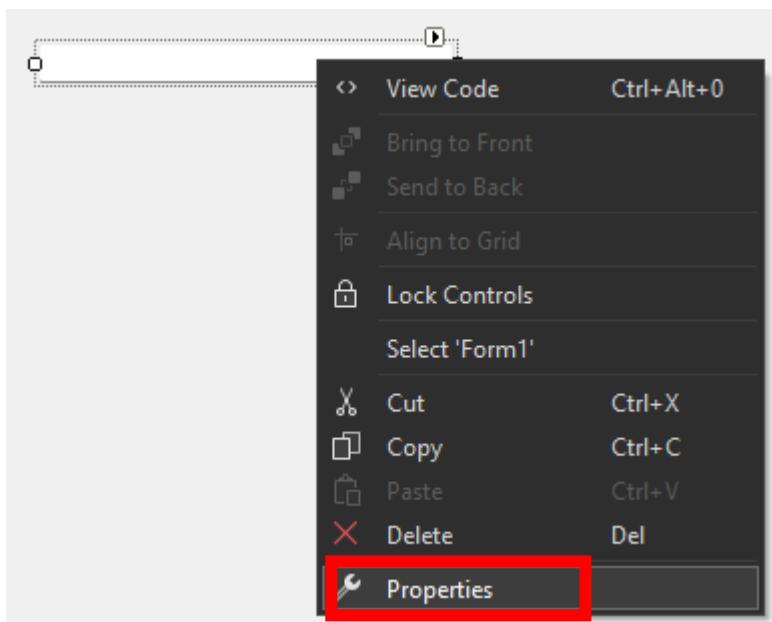


لو عايز حاجه ادخل فيها معلومات عندي حاجه اسمها **text box** بتدوس عليه وبرسمه بالموس عال form



ال **text box** ليه خصائص بتتحكم فيه من خلالها عشان تجييها بتحده بالماوس وتدوس على زرار ال f4

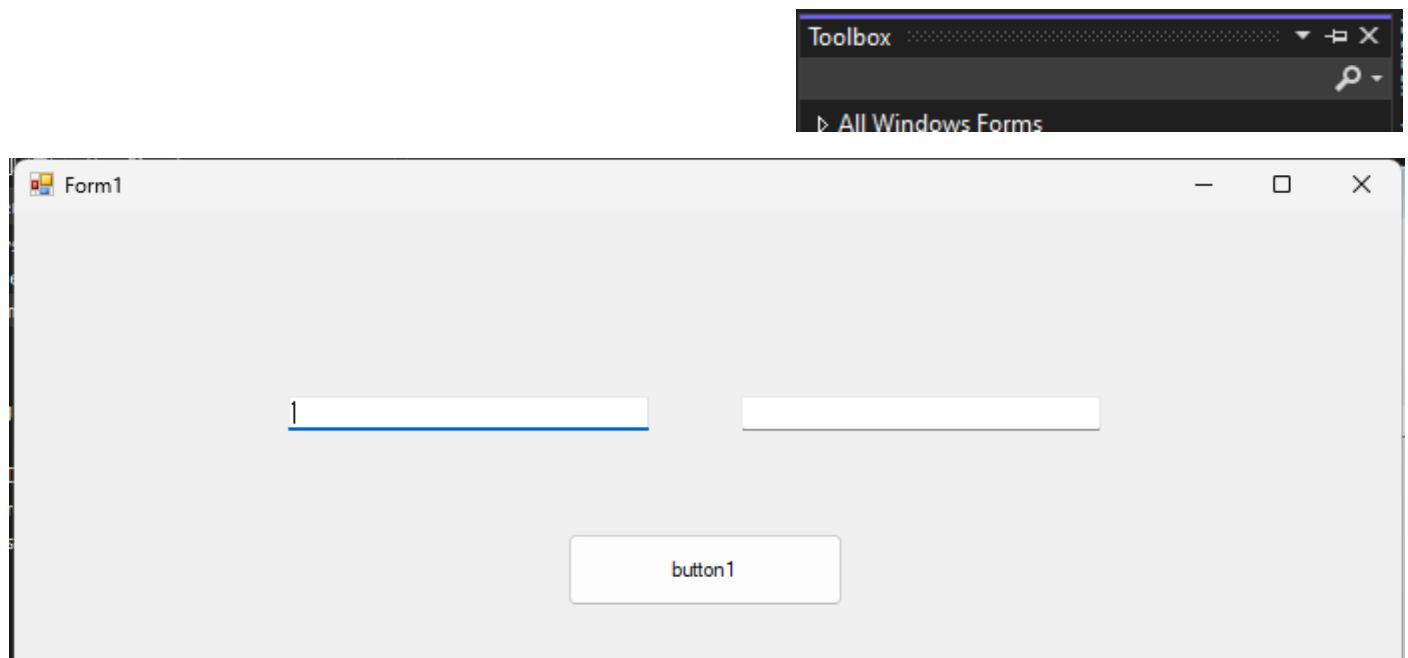
ياما كليك يمين عليه وتختر **properties**



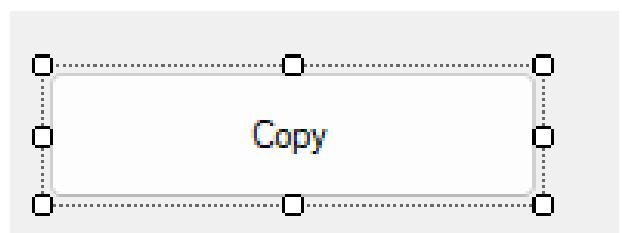
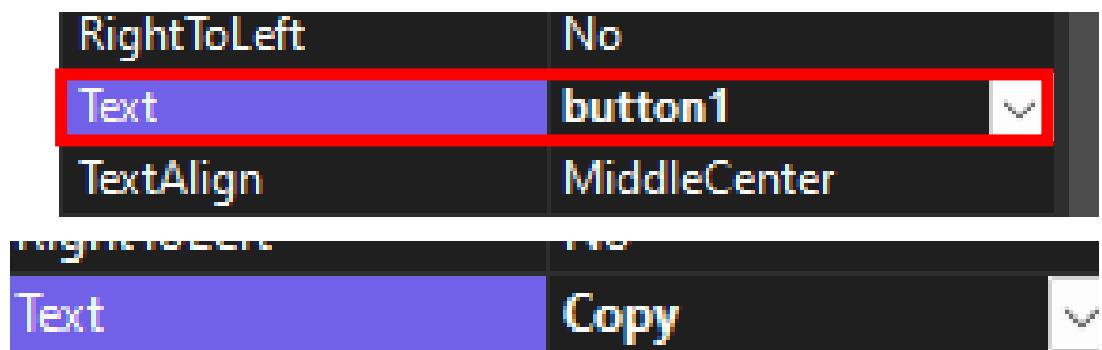
عاوزين نعمل **text box** تاني ونعمل **button** لما ندوس عليه ناخد النص اللي في ال **الأول ننسخه في الثاني**

تقدر تعمل **text box** تاني عن طريق نسخ ال **text box** اللي موجود حاليا ولصقه في نفس الفورم

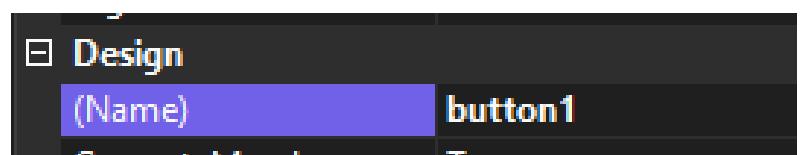
تقدر من هنا تبحث عن اسم ال control



عشان نعدل على اسم ال button اللي بيطلع لليوزر بجipp الخصائص بتاعته هلاقية خاصية اسمها text بتعدل الكلام اللي فيه هيتغير



فيه فرق بين اسم الزرار اللي بتعامل معاه برمجيا واسمه اللي بيظهر لليوزر  
الاسم البرمجي اهو سببه دلوقتني ماتعدلهوش



انا دلوقتني عندي text box1 عايز لما ادوس على ال button ياخد الكلام منه وينسخه في ال text box2  
وعندي لما بكتب اسم العنصر وبعد نقطه بتظهر لي الخصائص بتاعته كلها

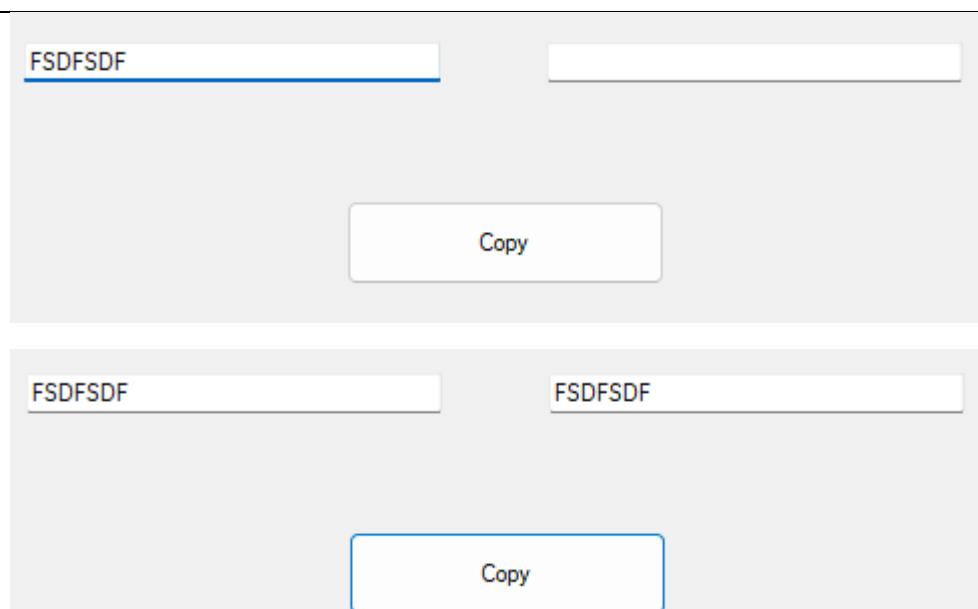
يعني عايز لما ادوس عالزرار أقوله `textbox2.Text = textBox1.Text`  
 عايز اعمل الحوار ده لما ادوس عالزرار  
 طيب عشان اعمل كده لازم اروح للزرار واتكتب الكود فيه هعملها ازاي؟  
 قالك دوس عالزرار مرتين هيوديك لل kod وهيفتح لك event اسمه `button1_Click` بيت استدعاءه لما اليوزر بي عمل click عالزرار  
 هقوم كاتب جواه الكود اللي قولتلك عليه

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp1c_
{
  public partial class Form1 : Form
  {
    public Form1()
    {
      InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
      textBox2.Text = textBox1.Text;
    }
  }
}
  
```



طيب انا عاوز اعمل زرار تاني لما الماوس ييجي عليه من غير ما ادوس على حاجه يعمل COPY

## Form1.cs [Design]

C# WindowsFormsApp1c#

### Design

(Name)

button2

## Properties

button2 System.Windows.Forms.Button



BackgroundImage



(none)

BackgroundImage

Events



Tile

### Mouse

MouseDown

MouseEnter

MouseEnter

MouseHover

MouseLeave

MouseMove

MouseUp

### Property Changed

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp1c_
{
    public partial class Form1 : Form
```

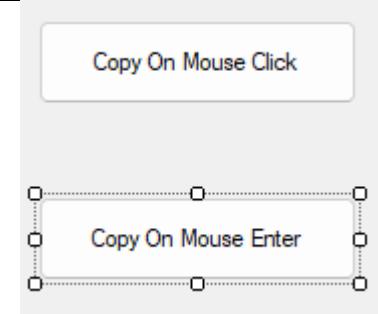
```

{
    public Form1()
    {
        InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        textBox2.Text = textBox1.Text;
    }

    private void MouseEnter(object sender, EventArgs e)
    {
        textBox2.Text = textBox1.Text;
    }
}

```



ال events كتير منها :-

- لما تكون دايس عالزرار : Mouse down

- لما الماوس بيجي عالزرار : Mouse enter

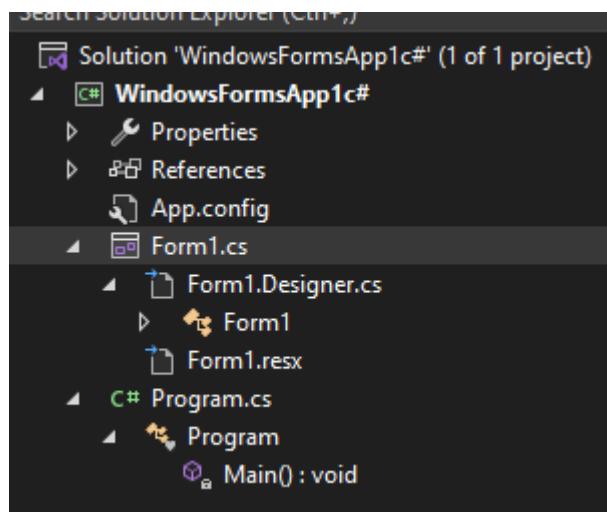
- لما الماوس يتحرك عالزرار : Mouse hover

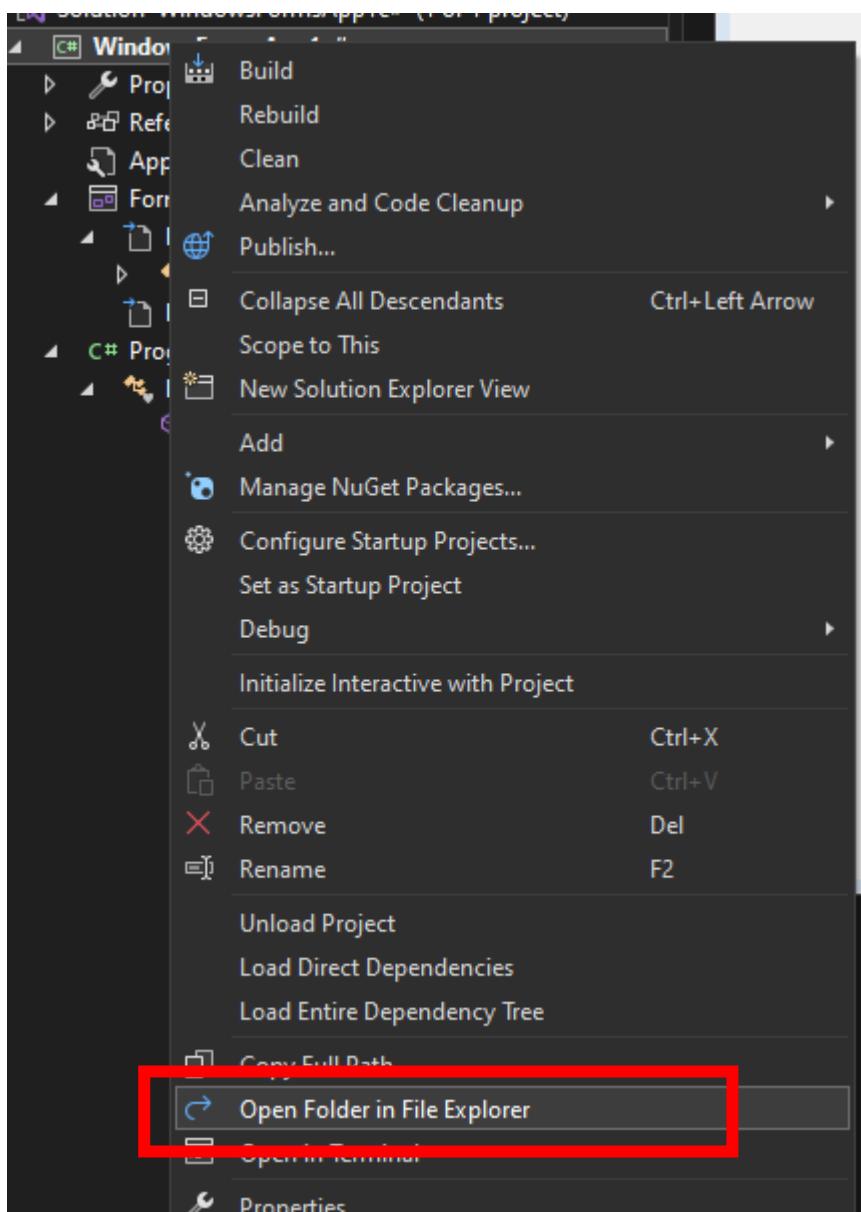
- لما الماوس يطلع بره الزرار يعني يكون واقف عليه وبعدين يخرج : Mouse leave

- لما تضغط كليك عالزرار مایحصلش حاجه غير لما ترفع صوباعك عن الزرار : Mouse up

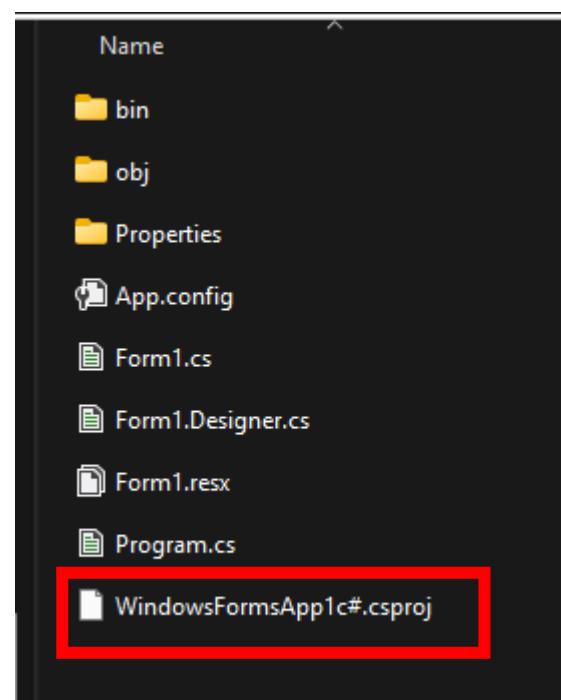
و فيه حاجات كتير

عشان تجيب ملفات المشروع



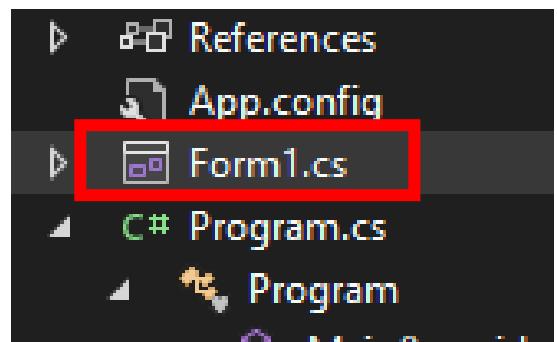


دي ملفات البرنامج كلها بيشتغل من الملف اللي عليه المربع الاحمر

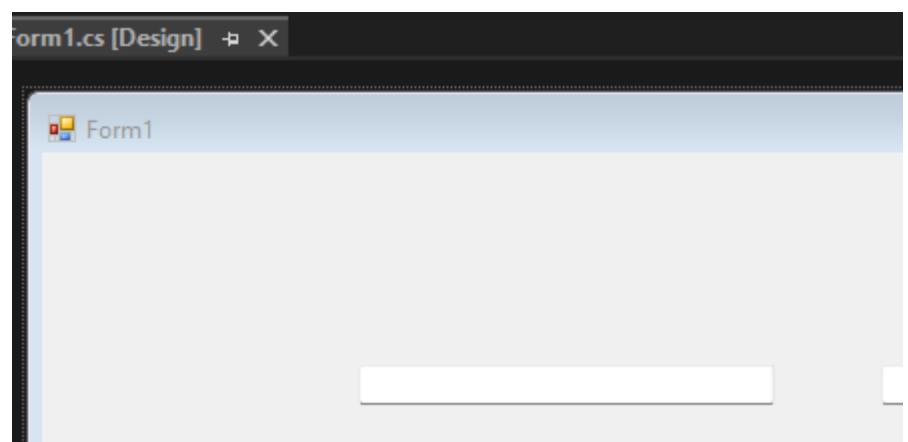


### Part3

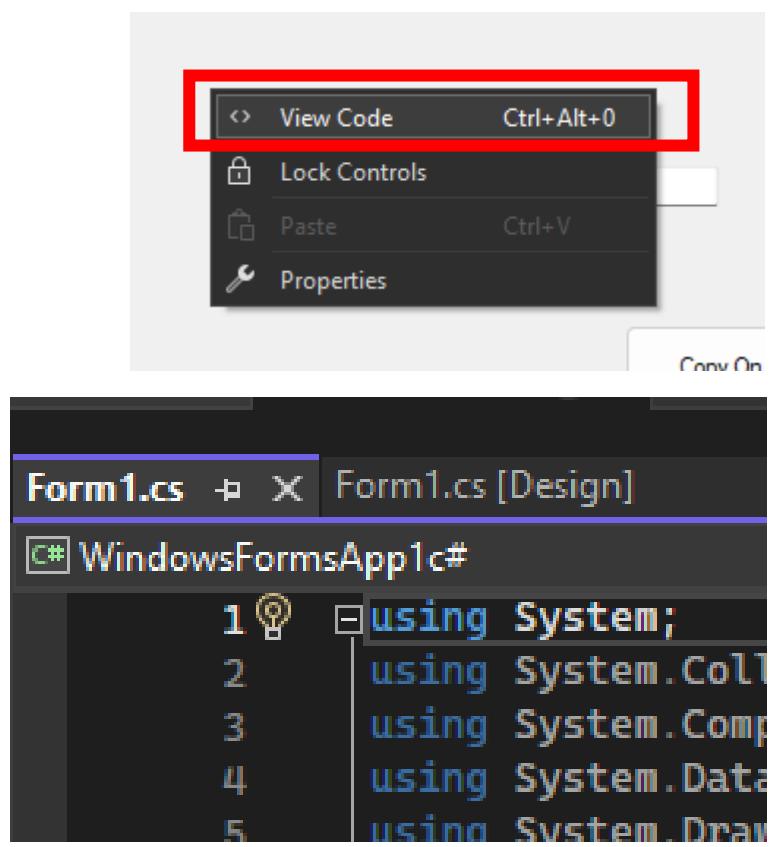
لو قفلت الشاشات بتاعت الفورم والكود ترجع تفتحها ازاي؟  
من القايمه اللي في الجنب هتدوس علي ده مرتين بالماوس



هيتحلّك شاشة الفورم

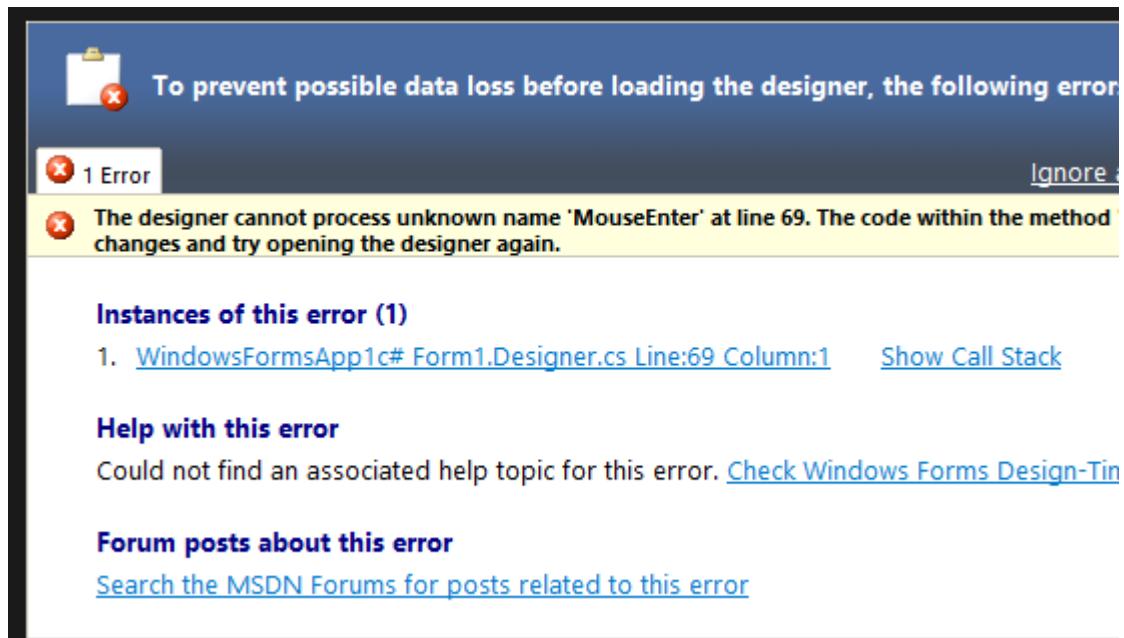


عشان تروح عالكود كليك يمين عالفورم وبعدها تختار view code

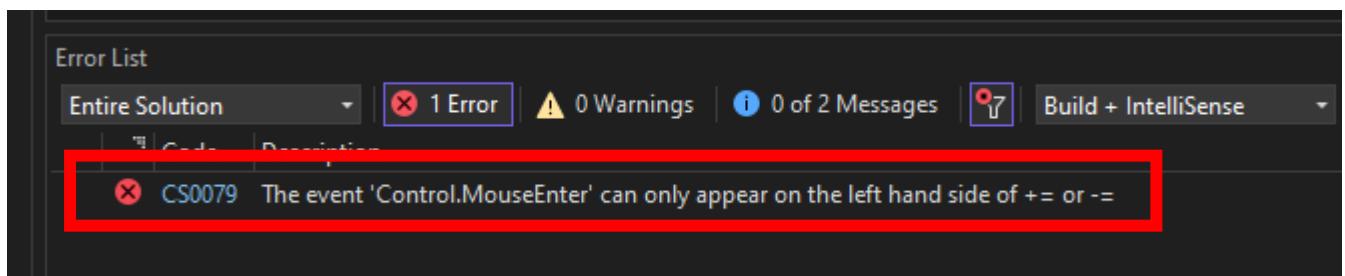


لو انت فاتح الكود وعايز تفتح ال form نفسها بتدوس shift+f7 يالما بتجيبيها زي ما جييتها الأول  
طيب لو جيت مثلا مسحت كود هتلطعلك رسالة خطأ

```
1 reference
private void button1_Click(object sender, EventArgs e)
{
    textBox2.Text = textBox1.Text;
}
```



عشان تحلها بتدوس double click بالماوس على ده



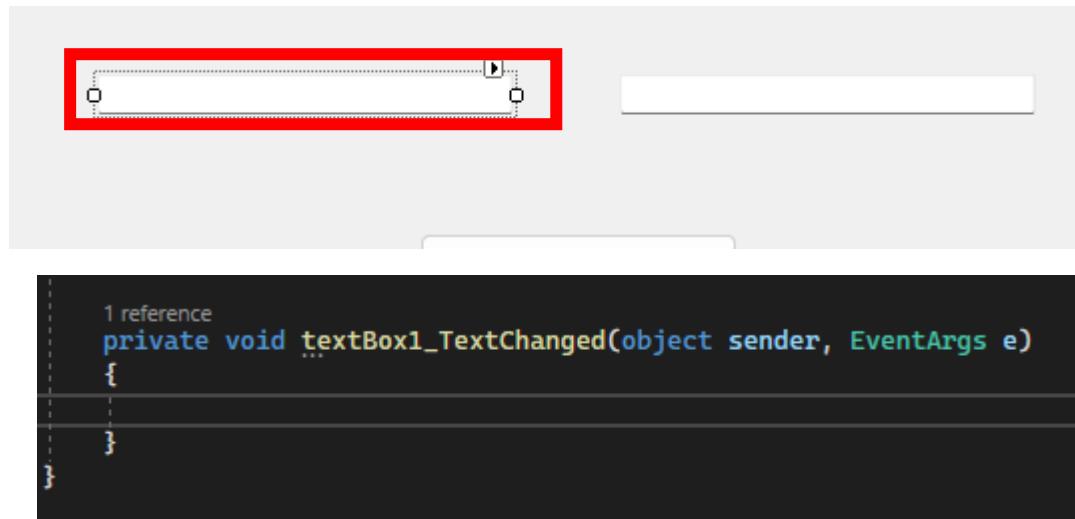
وتقوم شايل السطر ده

```
this.button2.Text = "Copy On Mouse Enter";
this.button2.UseVisualStyle.BackColor = tw;
this.button2.MouseEnter += new System.EventHandler(this.MouseEventHandler);
// Form1
//
```

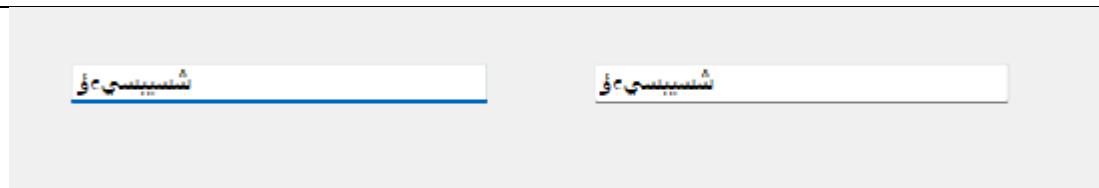
كده تقدر تروح تكتبه تاني لو عايز

طيب دلوقتي عايزين واحنا بنكتب في ال 1 text box الكلام يطلع تلقائي في ال 2 كانه لايپ

كده انا عايز اعمل event اثناء الكتابه او اثناء تغيير ال text في ال text box كل اللي هعمله اني هدوس مرتين عال text box الاقويه جابلي كود بتاع function اسمها text changed هطلع فيها نفس السطر اللي كنت بحطه وخلاص



```
private void textBox1_TextChanged(object sender, EventArgs e)
{
    textBox2.Text = textBox1.Text;
}
```



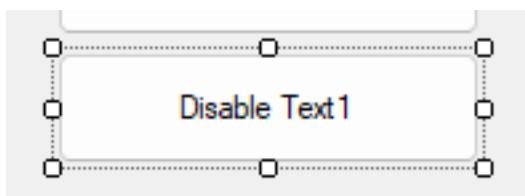
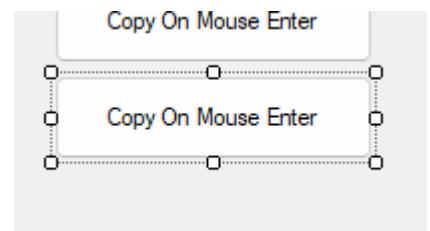
دلوتني عاوزين 2 text box ماحداش يقدر يعدل عليه كل اللي هعمله انى هخليله disabled

ContextMenuStrip	(none)
Enabled	True
HideSelection	True
ImeMode	False

ContextMenuStrip	(none)
Enabled	False
HideSelection	True



عاوزين نعمل زرار لما ندوس عليه يعمل text box1 لـ disable



```
private void button3_Click(object sender, EventArgs e)
{
    textBox1.Enabled = false;
}
```

عايزين نعمل زرار تاني يخليها enabled



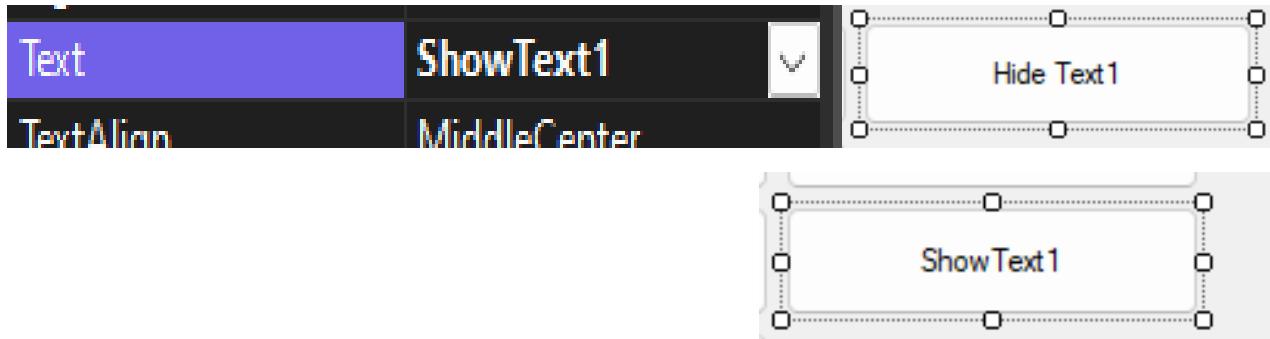
```
private void button4_Click(object sender, EventArgs e)
{
    textBox1.Enabled = true;
}
```

عاوزين نعمل زرار ندوس عليه يخلي text box1 يعمله hidden يعني مايظهرش للิوزر



```
private void button5_Click(object sender, EventArgs e)
{
    textBox1.Visible=false;
}
```

عاوزين نعمل زرار يعمل `text1` لـ `unhide`



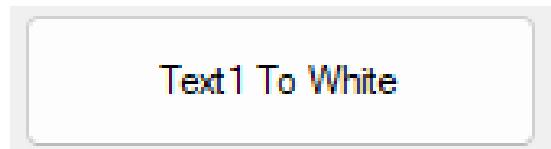
```
private void button6_Click(object sender, EventArgs e)
{
    textBox1.Visible = true;
}
```

عاوزين نعمل زرار يخلي لون الخلفيه بناءت `text1` بلون احمر



```
private void button7_Click(object sender, EventArgs e)
{
    textBox1.BackColor = Color.Red;
}
```

زرار تاني يخلي اللون ابيض



```
private void button8_Click(object sender, EventArgs e)
{
    textBox1.BackColor = Color.White;
}
```

الزرار اللي بيخليلون احمر عاوزينه يخلي لون زرار ال `copy` كمان احمر

```
private void button7_Click(object sender, EventArgs e)
```

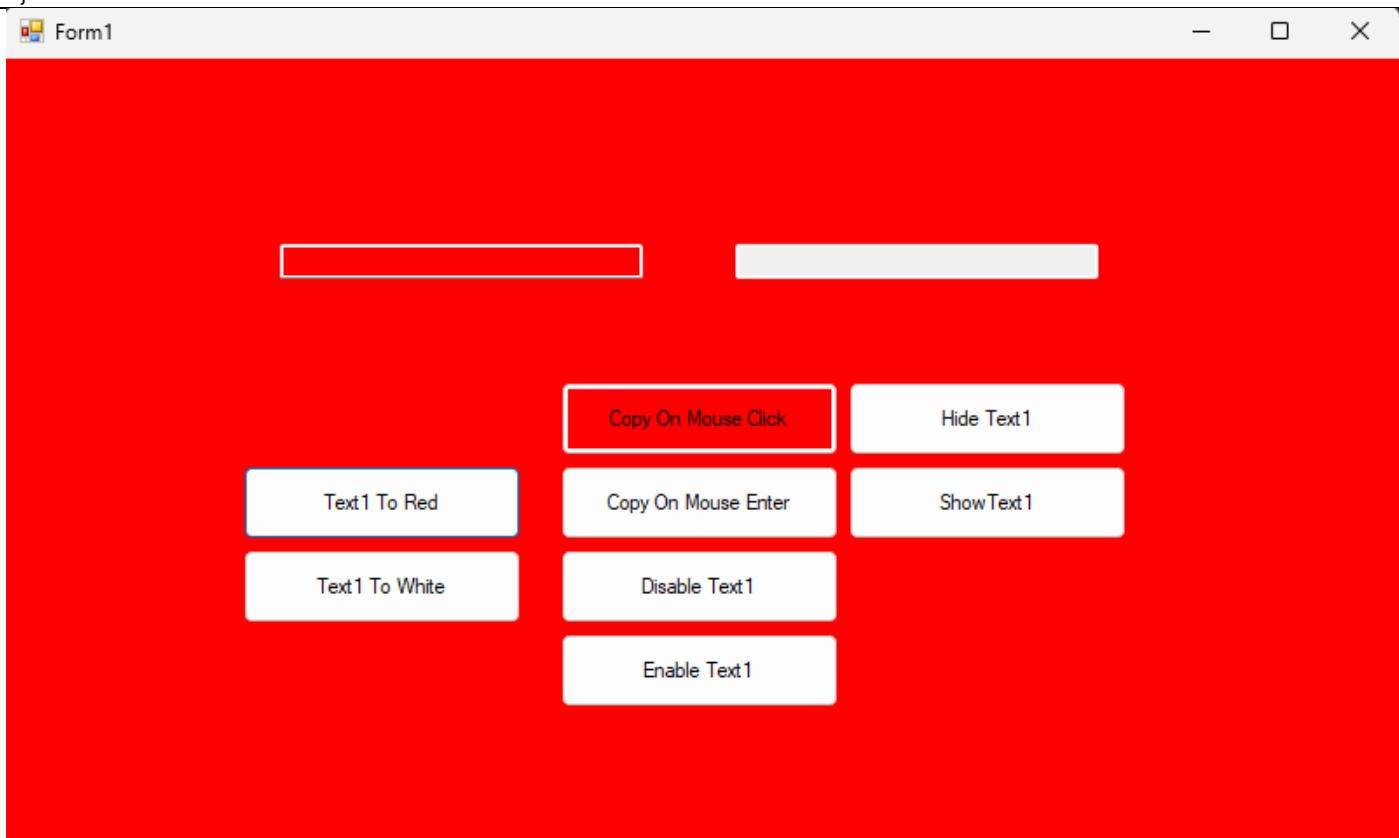
```
{
    textBox1.BackColor = Color.Red;
    button1.BackColor = Color.Red;
}
```

الخصائص دي موجوده في كل ال controls حتى في الفورم

طيب ازاي نوصل للفورم؟

عن طريق انك تكتب this

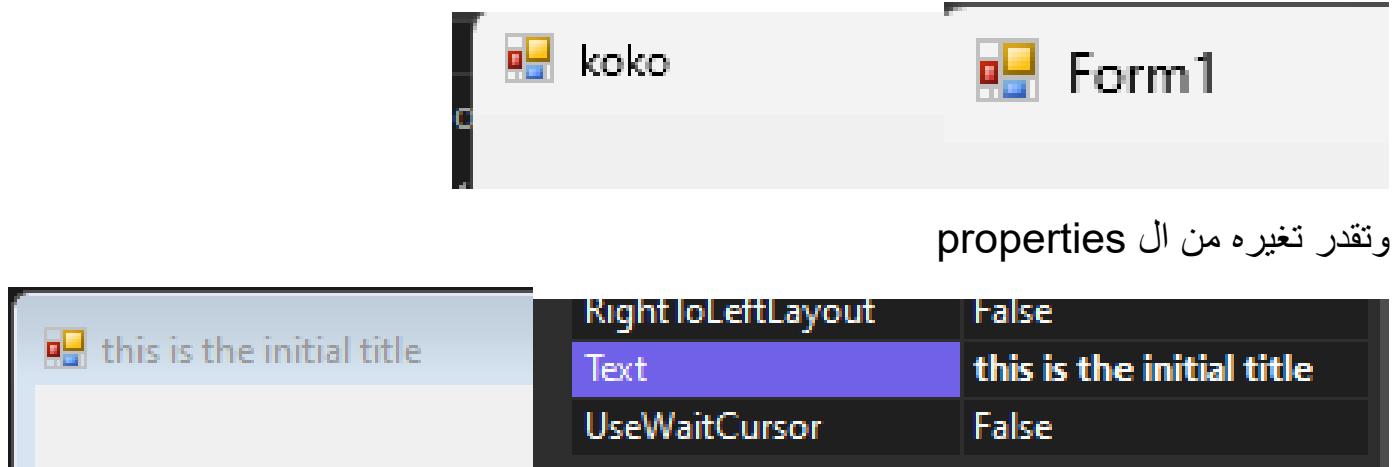
```
private void button7_Click(object sender, EventArgs e)
{
    textBox1.BackColor = Color.Red;
    button1.BackColor = Color.Red;
    this.BackColor = Color.Red;
}
```



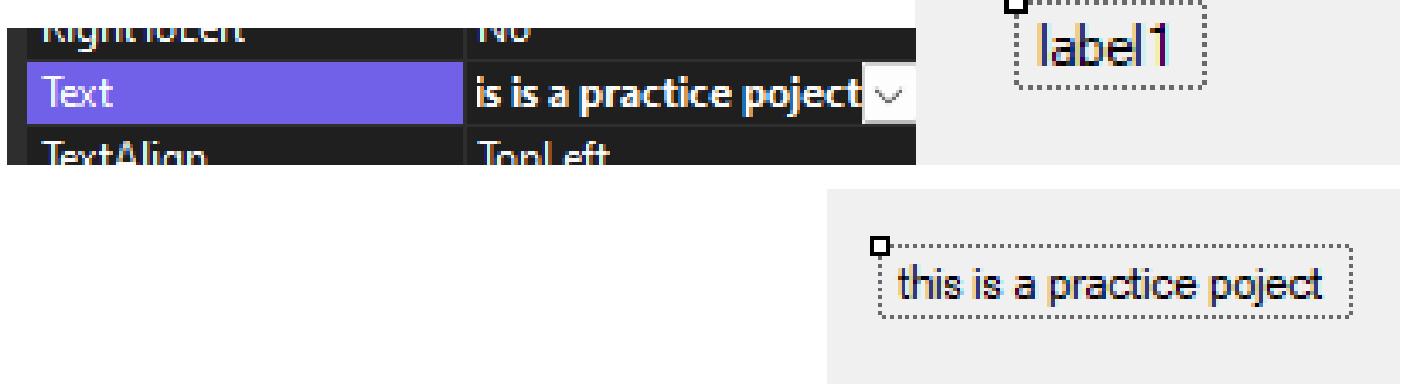
عاوزين نعمل زرار يغير ال TITLE بتاع الفورم



```
private void button9_Click(object sender, EventArgs e)
{
    this.Text = "koko";
}
```

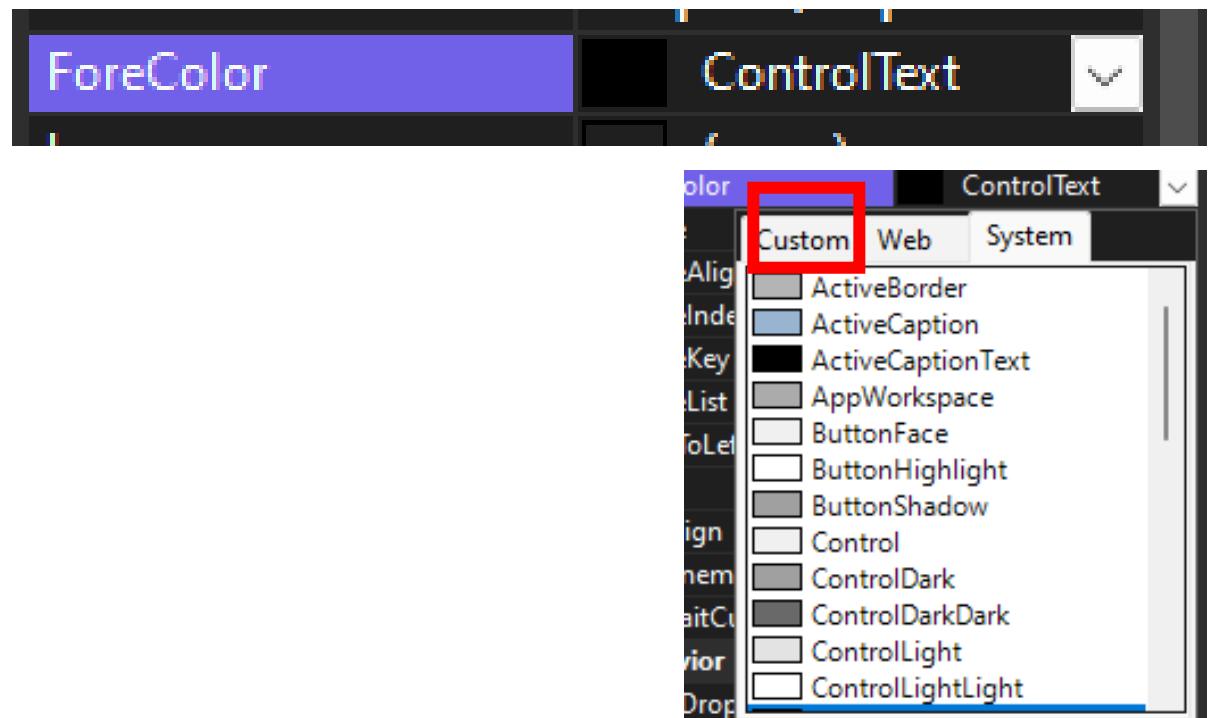
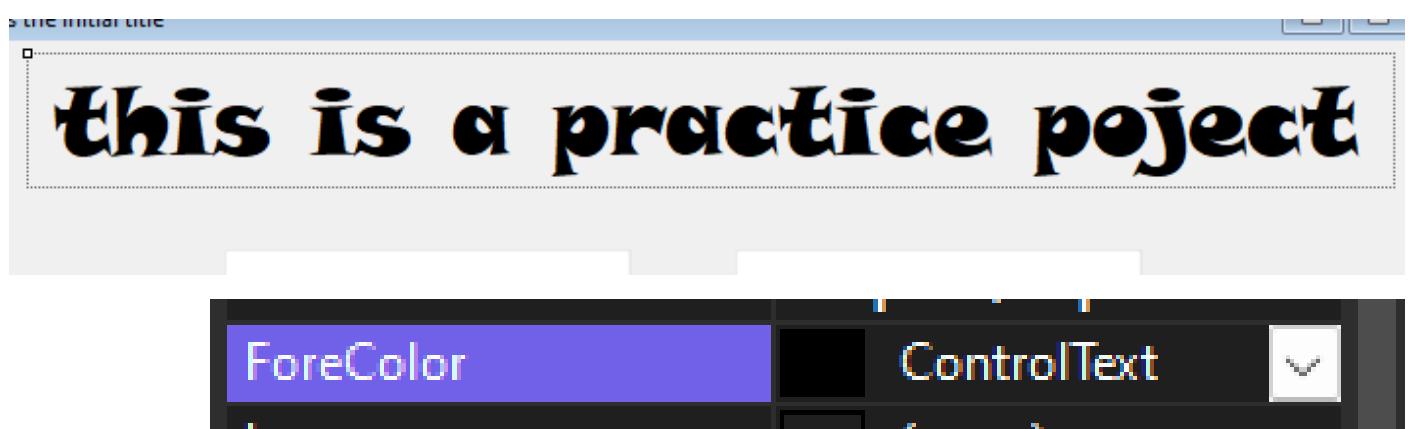
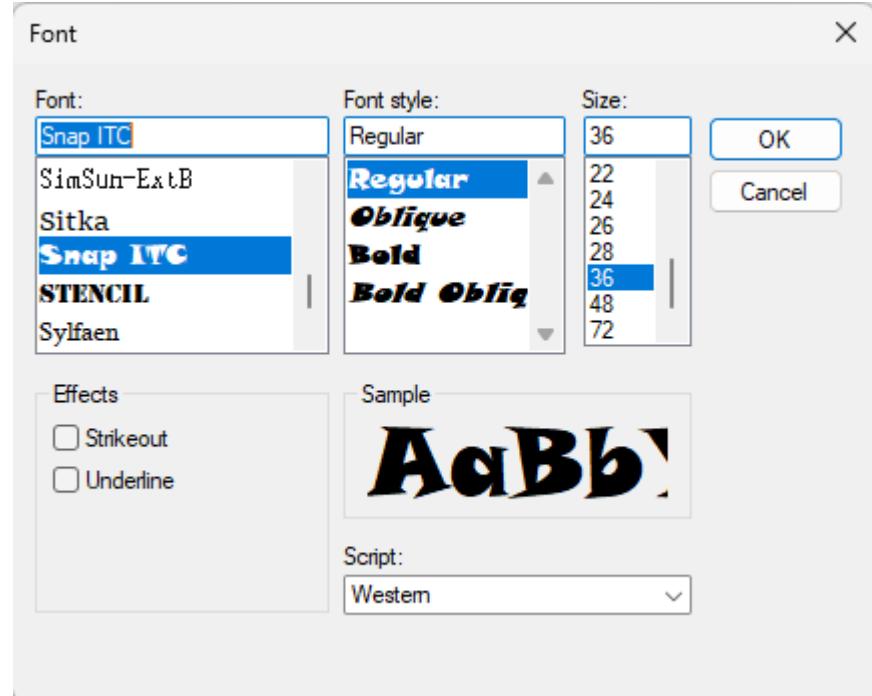


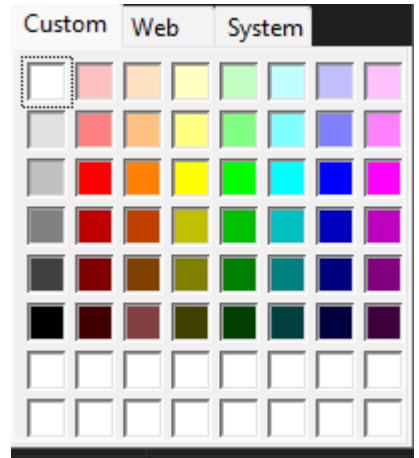
تقدير تغير ال icon بقى ابعتا عن ال form وتنظر في ال task bar لما تصغره فيه label تاني اسمه control



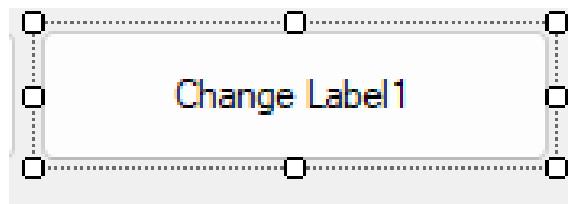
هنغير في ال font بقى وتقدير تعمل نفس الكلام في باقي ال controls







وتقدر تختار ال controls كلهم وتنظمهم في الفوم زي ماانت عايز  
هعمل زرار يغير النص المكتوب في ال label



```
private void button10_Click(object sender, EventArgs e)
{
    label1.Text = "Mohammed Abu-Hadhoud";
}
```

# Mohammed Abu-Hadhoud

عاوزين نكتب في ال text1 يغير المتوب في ال label  
هيكون بنفس الطريقه اللي عملناها قبل كده

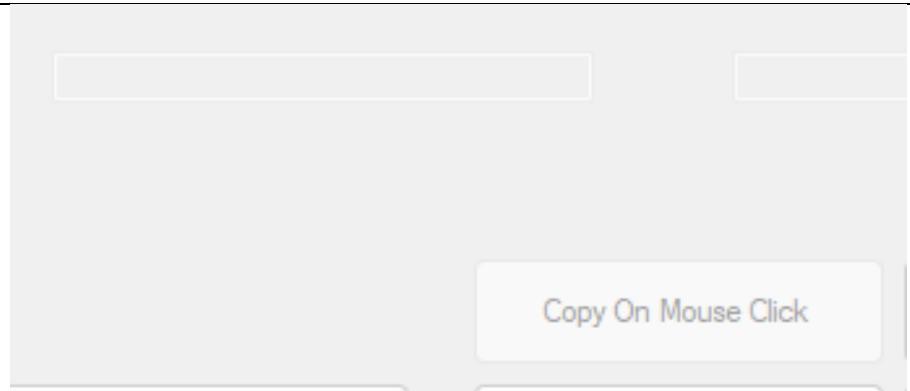
```
private void textBox1_TextChanged(object sender, EventArgs e)
{
    textBox2.Text = textBox1.Text;
    label1.Text = textBox1.Text;
}
```

# asdasds

asdasds|

نقدر نخلی الزرار بتاع ال copy يتعمله disable مع ال text1

```
private void button3_Click(object sender, EventArgs e)
{
    textBox1.Enabled = false;
    button1.Enabled = false;
}
```



عاوزين الزرار الثاني يعمله enable

```
private void button4_Click(object sender, EventArgs e)
{
    textBox1.Enabled = true;
    button1.Enabled = true;
}
```

## Part 4 : Some formatting and Alignments

هنعمل زرار ونعدل في ال appearance بتاعه واللي هنعمله هنا موجود في كل ال controls الثانيه عادي

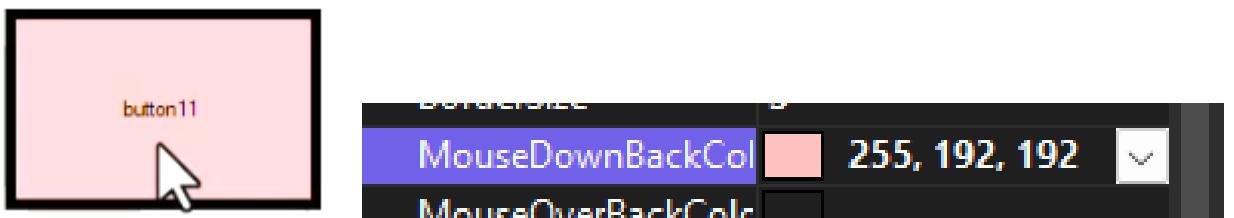


دي بتخلی الزرار بارز او مسطح

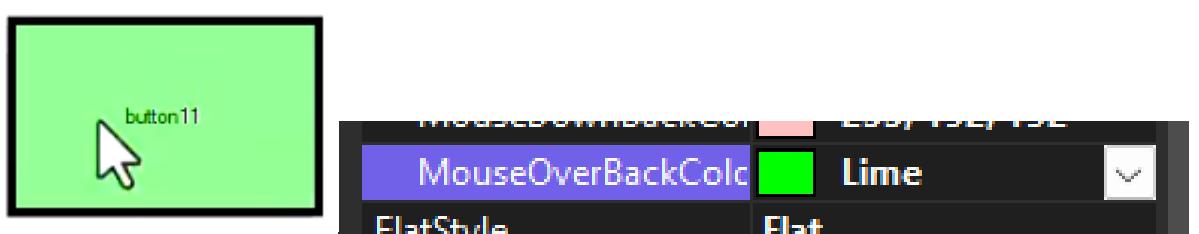




عاوز اغير لون الزرار وانا بدوس عليه يعني وانا عامل mouse down يعني ضاغط عليه من غير ماسبيه

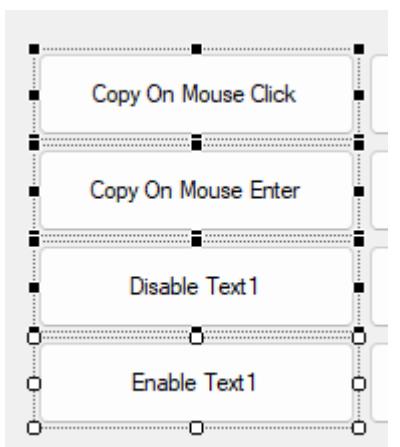


لما الماوس بيتعدي عالزرار بيغير لونه عاوز اغير اللون ده

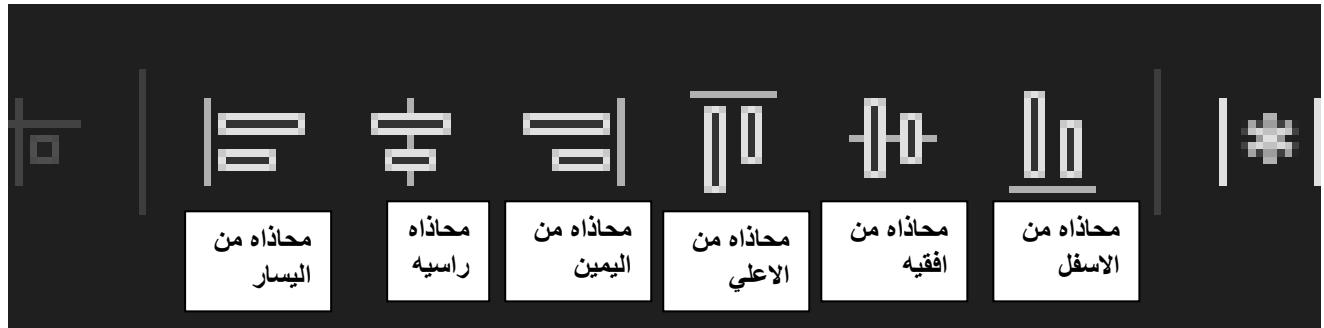


أي حاجه مكتوب عليها سيستم فهii بتتبع اعدادات السيستم عندك

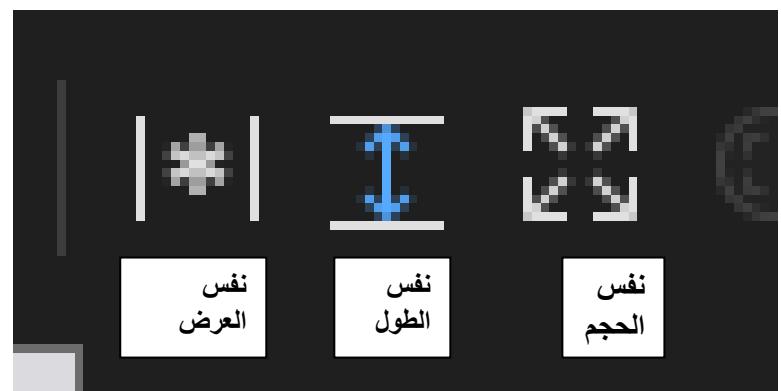
طيب دلوقتي عاوزين نعمل محاذاه لل controls مع بعض  
بختار اول control اللي عايزة المحاذاه تكون عليه  
وبعدين بدوس على ctrl وبحدد ال controls اللي عايزة تعمل محاذاه علي اول واحد



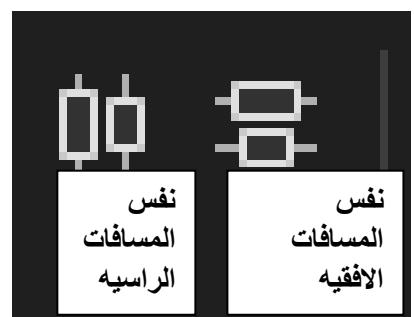
اللي بالاسود هي عملوا محاذاه على الأبيض



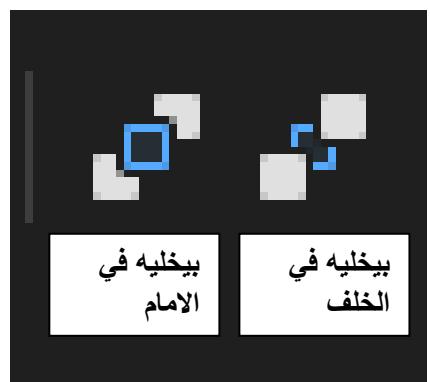
طيب عايزين نخلی عدد من ال controls يكونوا نفس ال size بتاع control اخر  
هنحددهم زي ما عملنا قبل كده وبعدين نروح هنا



طيب عاوزين نخلی المسافات بين العناصر متساوية



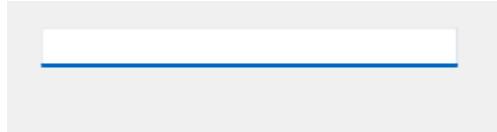
لو عندك control محظوظ فوق control تاني تقدر تتحكم في اظهاره انه يكون ظاهر او وتداري ورا  
الآخر



## Part 5 : TabIndex, Tabstop, and New Forms (Default, Show, ShowDialog, Close)

ال desktop application لازم تقدر تشتغل عليه من غير ماوس وان شغل اليوزر مايتأثرش ان كان عنده ماوس او لا

انا دلوقي فتح البرنامج لقيت ال focus معمول عال text1



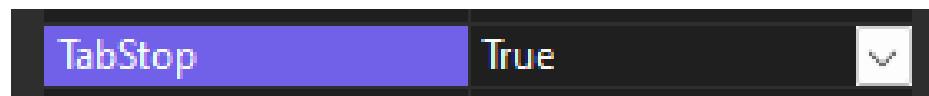
ولو دوست علي زرار ال tab هينزل علي control ثاني وهكذا



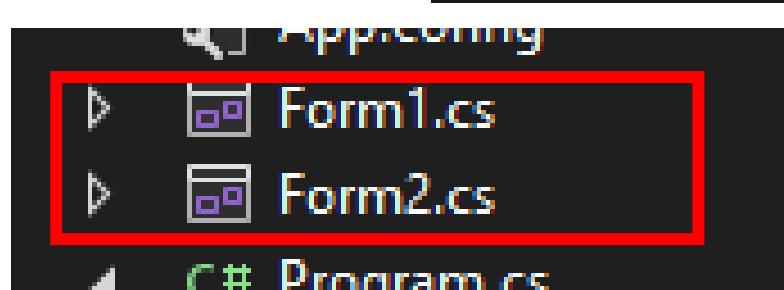
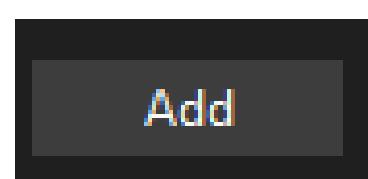
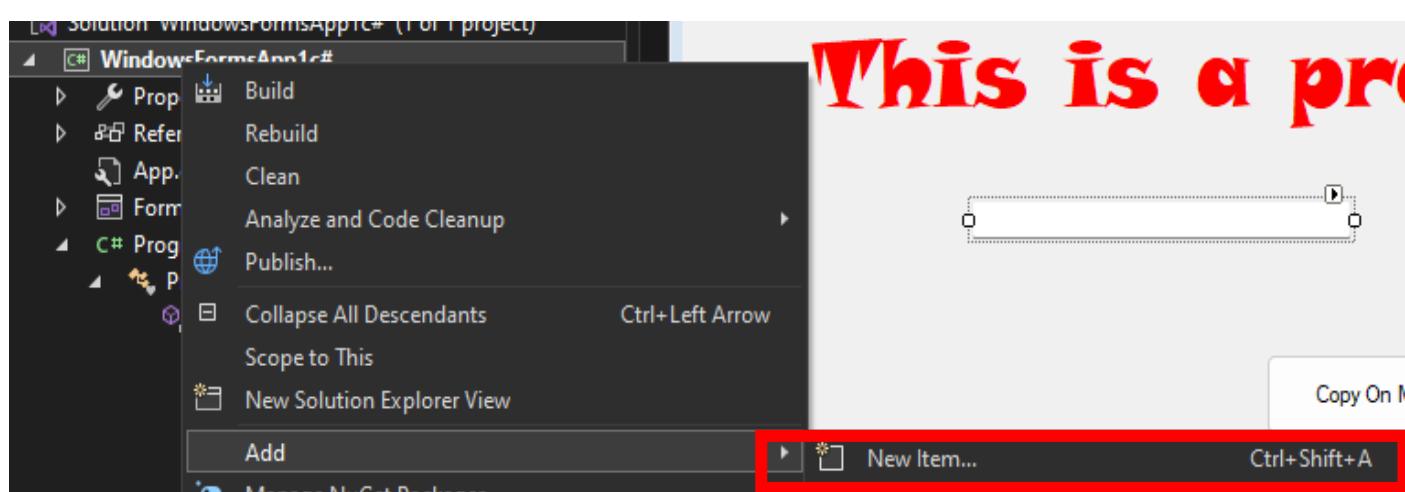
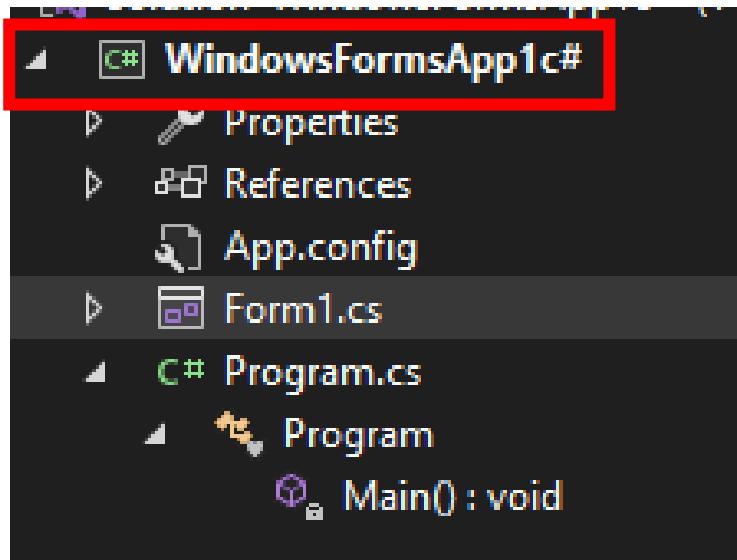
انا دلوقي عاوز ارتب تحركات ال focus دي بما يناسب البرنامج بتاعي هنا فيه حاجه اسمها tab stop وحاجه تانيه اسمها tab index بروح علي ال properties بتاعت كل control وبغير الترتيب من هنا



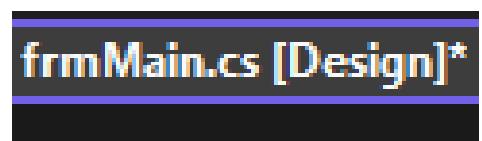
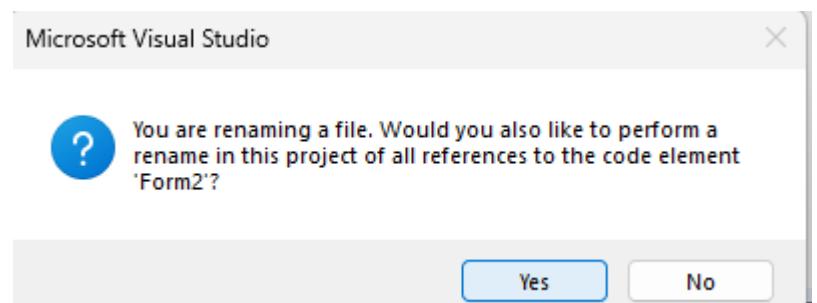
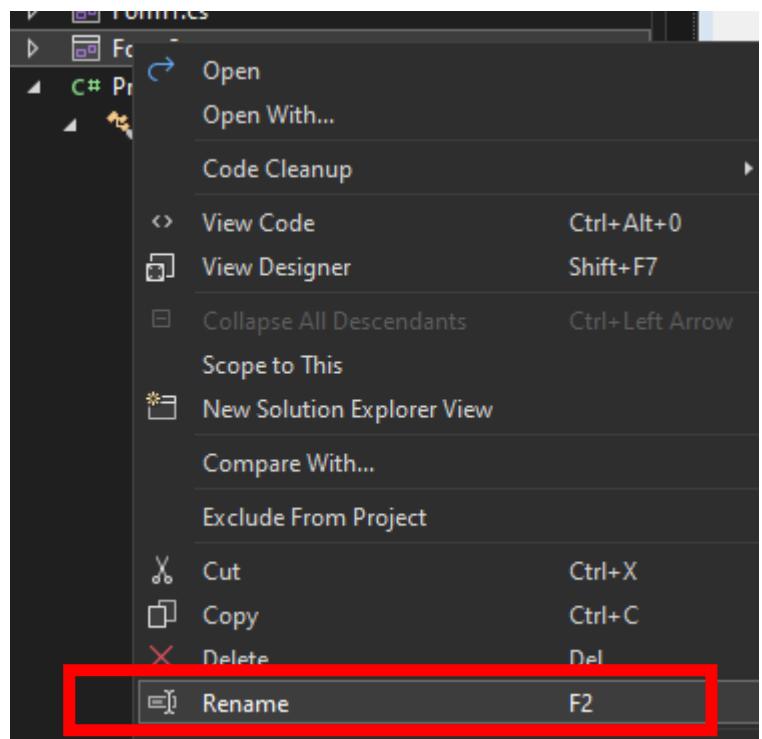
وبتمسك كل واحد تحط ترتيبه في الحركه علي كيفك طيب لو فيه control معين مش عايز ال tab بيجي عليها بروح للخصائص بتاعتته وبغير ال tab stop اعملها false



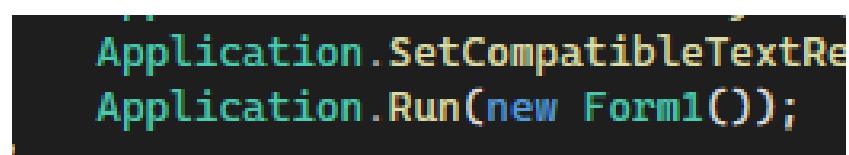
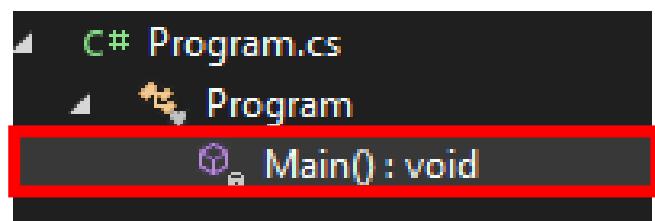
طيب دلوقي عاوزين نعمل form جديد كليك يمين عال windows form add>>new item وبعدها نختار project وبعدين



عاوز اغير اسمه



عاوز اخليه هو اللي يستغل الأول لما افتح البرنامج



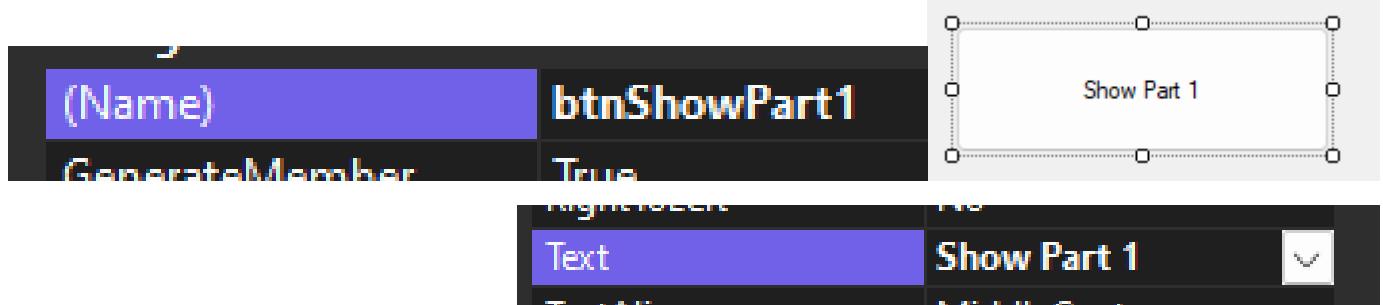
غيره و اخليه الفورم اللي انا عايزه



هنجير ال title بتاعه



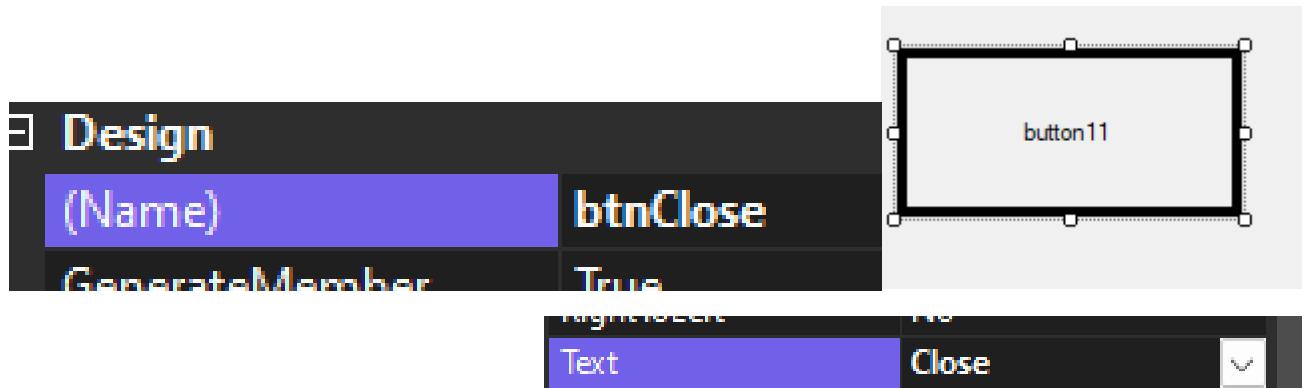
هحط فيه زرار ينقلنا عالفورم الثاني



يفضل انك تعمل اختصارات لاسماء ال controls زي الازرار تبقي txt وال btn وبعدها الوظيفه بناعاته عشان تسهل عليك الوصول ليها برمجيا وبعدين نعمل ال event لما ندوس عليه ينقلنا للفورم الثاني

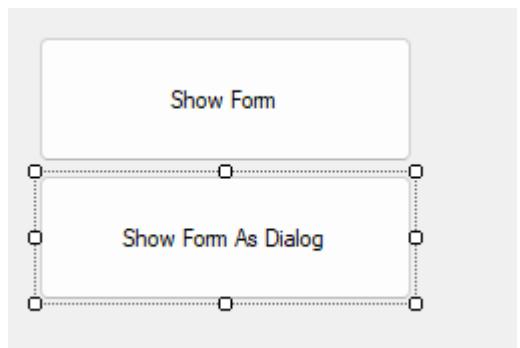
```
private void btnShowPart1_Click(object sender, EventArgs e)
{
    Form frm1 = new Form1();
    frm1.Show();
}
```

عاوزين نروح للفورم الثاني ونخلي الزرار الكبير يقفل الفورم



```
private void btnClose_Click(object sender, EventArgs e)
{
    this.Close();
}
```

هنعمل زرار تاني في ال show form as dialog main form ودي فايدتها انك لما تنقل لفورم تاني ال main form ماتقدرش تعمل عليه أي حاجه غير لما تغلق الفورم الجديد اللي انت فاتحه



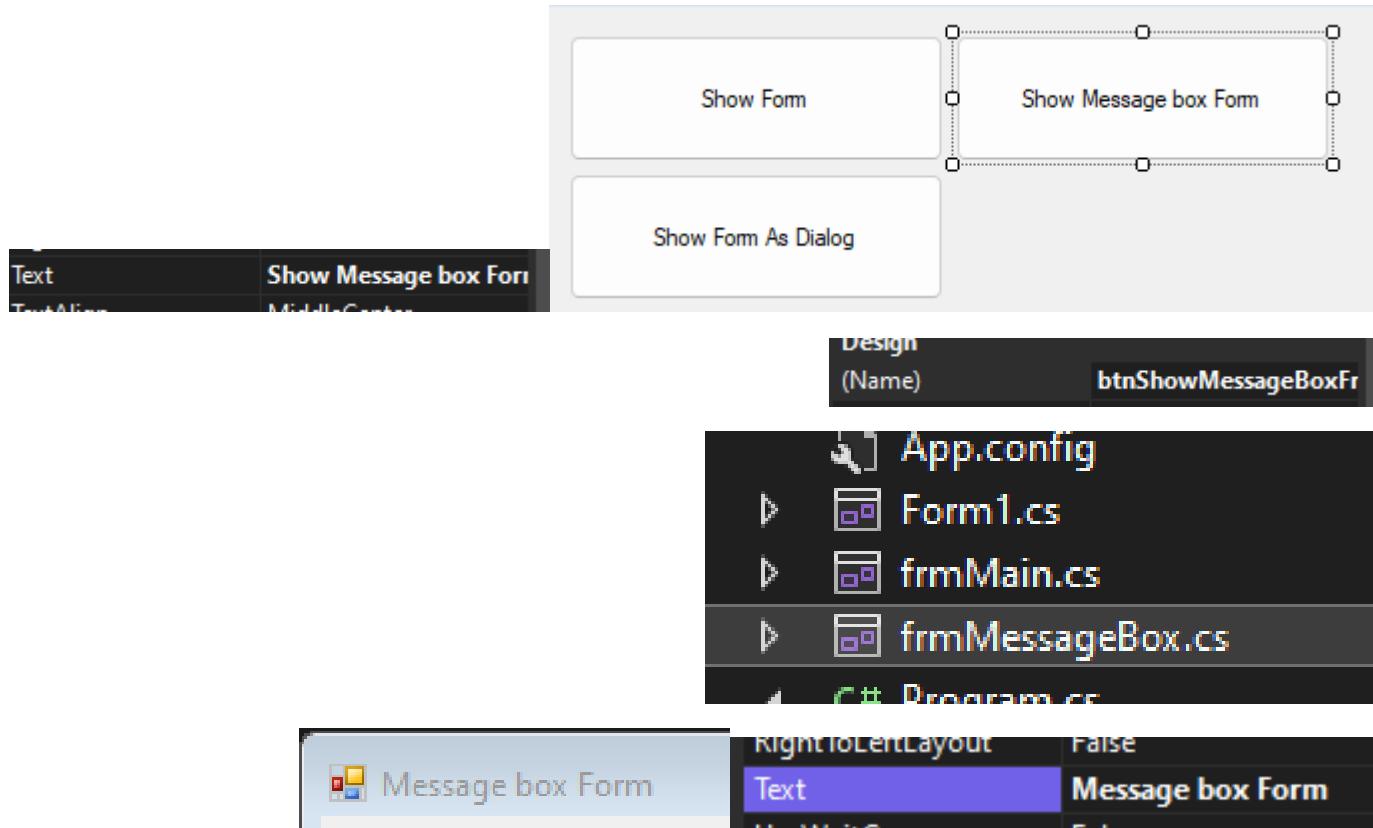
```

private void button1_Click(object sender, EventArgs e)
{
    Form frm1 = new Form();
    frm1.ShowDialog();
}

```

## Part 6 : MessageBox

ال message box ممكن تستخدمها عشان تبيّن رساله معينه لليوزر او تاخذ منه input  
هنعمل فورم خاص بال message box ونعمل زرار في ال main form ينقّلنا للفورم ده



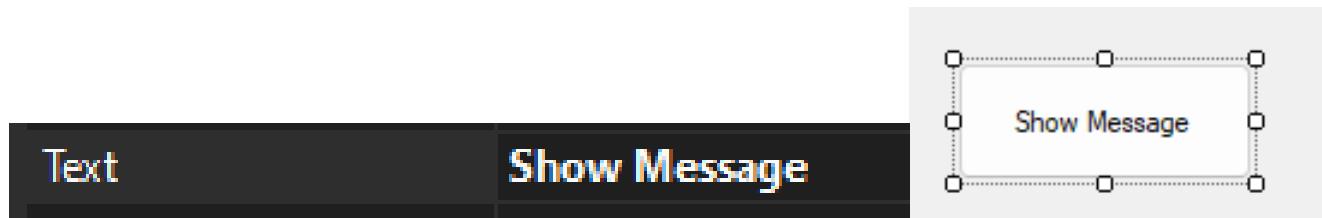
وده ال event بتاع الزرار

```

private void btnShowMessageBoxFrm_Click(object sender, EventArgs e)
{
    Form frmMessageBox = new frmMessageBox();
    frmMessageBox.ShowDialog();
}

```

وهنعمل زرار في الفورم الجديد



ودلوقتي هنخلي الزرار بطلعنا message box

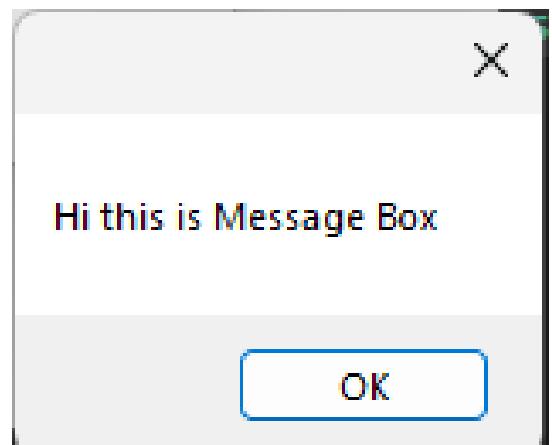
```

private void button1_Click(object sender, EventArgs e)
{
}

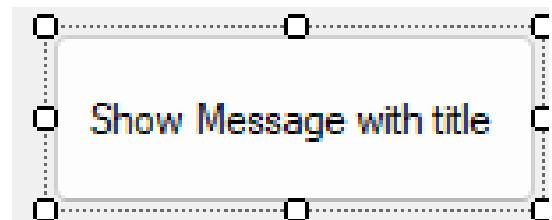
```

```
    MessageBox.Show("Hi this is Message Box");  
}
```

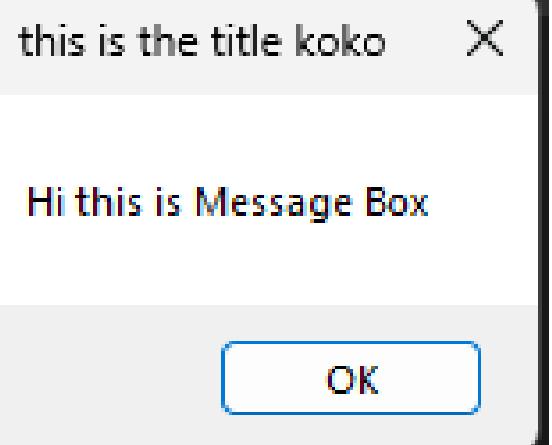
وده الشكل اللي طلع



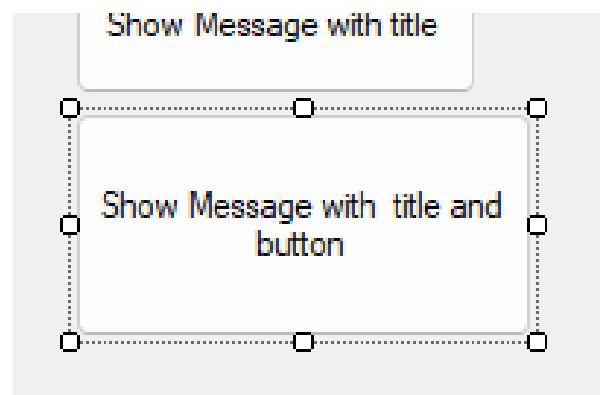
عاوزين نعمله title و هنعمله في زرار جديد عشان نشوف الفرق



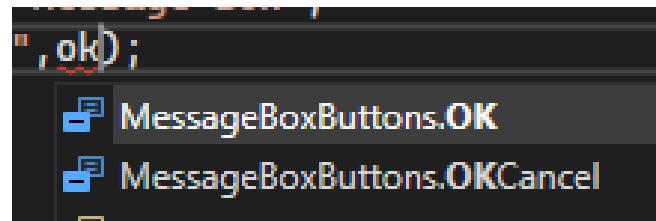
```
private void button2_Click(object sender, EventArgs e)  
{  
    MessageBox.Show("Hi this is Message Box","this is the title koko");  
}
```



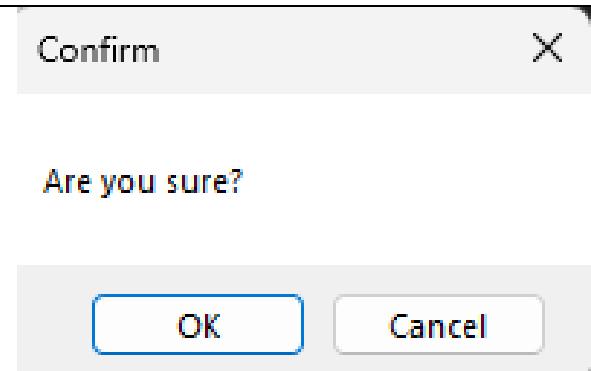
قدر اضيف زرار لـ message box



اول اختيار هو يظهر لك زرار ال ok بس والثاني زرار ال cancel وال ok



```
private void button3_Click(object sender, EventArgs e)
{
    MessageBox.Show("Are you sure?",
        "Confirm", MessageBoxButtons.OKCancel);
}
```

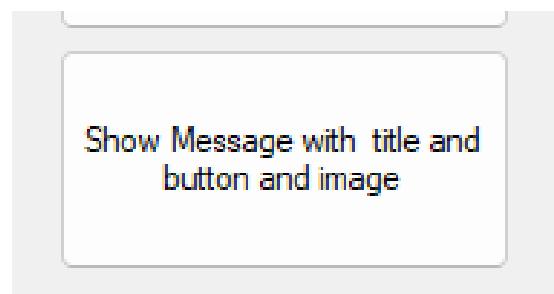


طيب ازاي هنعرف انه اليوزر داس على ok ولا cancel if statement هنحط وخلاص

```
private void button3_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Are you sure?",
        "Confirm", MessageBoxButtons.OKCancel) == DialogResult.OK) {

        MessageBox.Show("User Pressed Ok");
    }
}
```

عاوزين نضيف صوره في ال message box



```
private void button4_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Are you sure?",
        "Confirm",
        MessageBoxButtons.OKCancel,
        MessageBoxIcon.Question) == DialogResult.OK)
    {
```

```
    MessageBox.Show("User Pressed Ok");  
};
```

Confirm



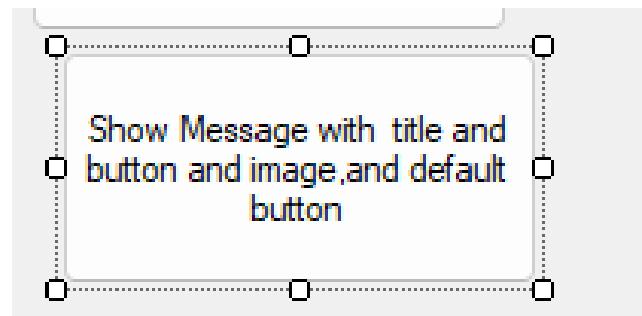
Are you sure?

OK

Cancel

فيه أيونات تانية تقدر تختار منها

دلوقي انا عاوز ال default button او الزرار اللي عليه focus يكون ال cancel عشان  
مايدوسش enter بالغلط



```
private void button5_Click(object sender, EventArgs e)  
{  
    if (MessageBox.Show("Are you sure?",  
        "Confirm",  
        MessageBoxButtons.OKCancel,  
        MessageBoxIcon.Question,MessageBoxDefaultButton.Button2) == DialogResult.OK)  
    {  
  
        MessageBox.Show("User Pressed Ok");  
    };
```

Confirm



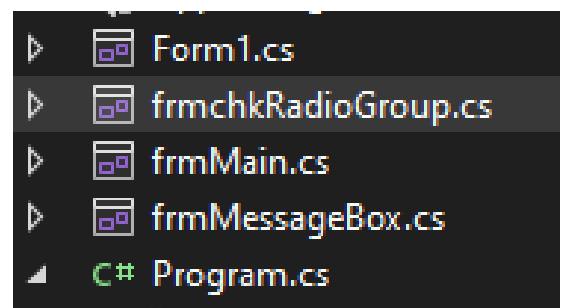
Are you sure?

OK

Cancel

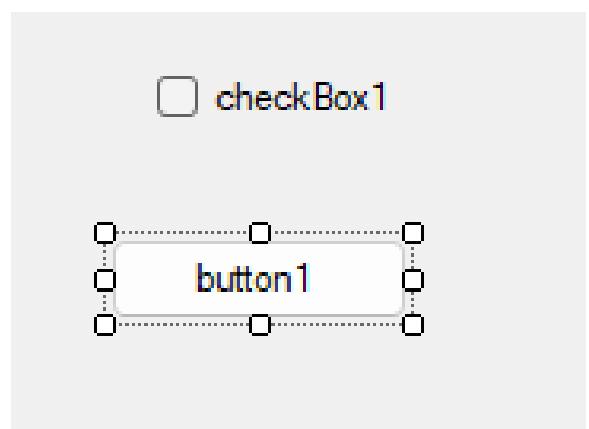
**Part 7: CheckBox, RadioButton, GroupBox, and Tag**

## هعمل فورم جديد

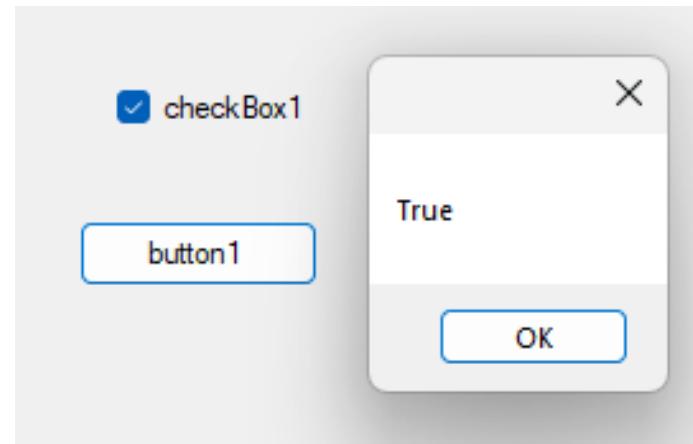
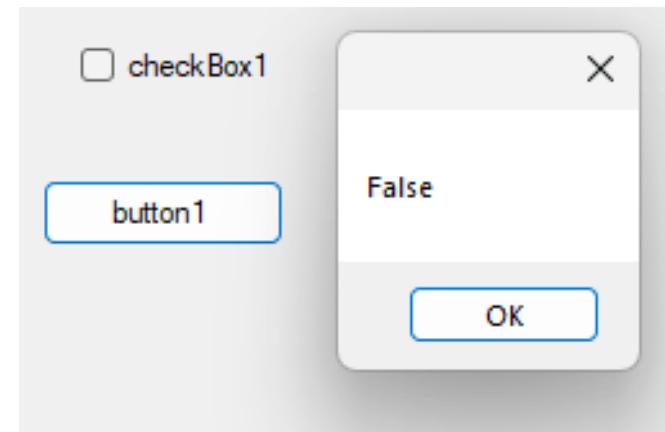


```
private void button2_Click(object sender, EventArgs e)
{
    Form frm1 = new frmchkRadioGroup();
    frm1.ShowDialog();
}
```

اول حاجه هنضيفها هيا **check box** وده بيدي اليوزر انه يحدد اكتر من اختيار وهنحط كمان زرار يطلعنا رسالة بتقول ان تم اختيار ال **check box** ده ولا لا



```
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show(checkBox1.Checked.ToString());
}
```

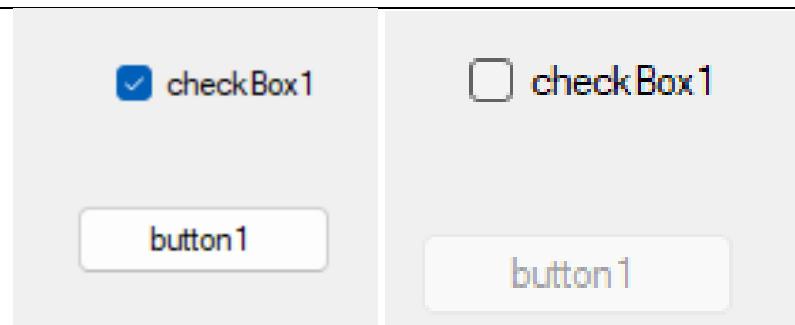


بقدر استخدم الكلام ده في if statement لو اتعمله checked اعمل كذا ولو unchecked اعمل كذا  
برضه بتقدر تعمل عليه event انه مثلا لو هوا checked تعمل للزرار enable او disable  
بتدوس عليه مرتين بيطلعلك event اسمه checked changed يعني الحدث اللي بيحصل لو اتعمل  
uncheck او check  
هوا هنا السطر ده بيرجع bool

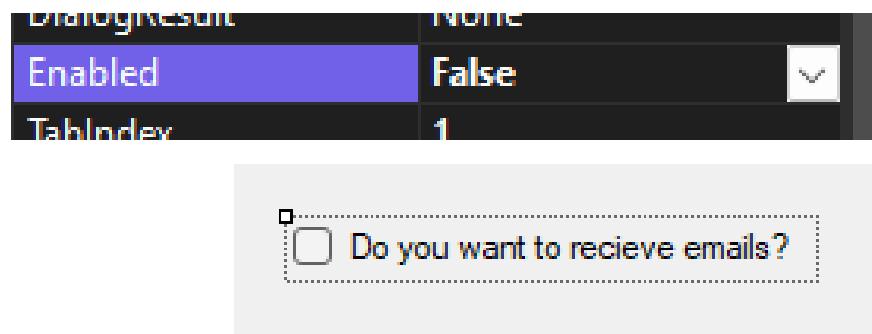
**checkBox1.Checked**

فتقدر تستخدمه كده

```
private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
    button1.Enabled = checkBox1.Checked;
}
```



وتقدر تخلّي ال default بتاع الزرار disabled



زي ما قولنا انك يفضل تخلي الاسم البرمجي مختصر وبعد الاختصار تكتب الوظيفه بقاعدته وهذا بيقولك انك تخلی الاختصار بتاع ال chk يبقى check box

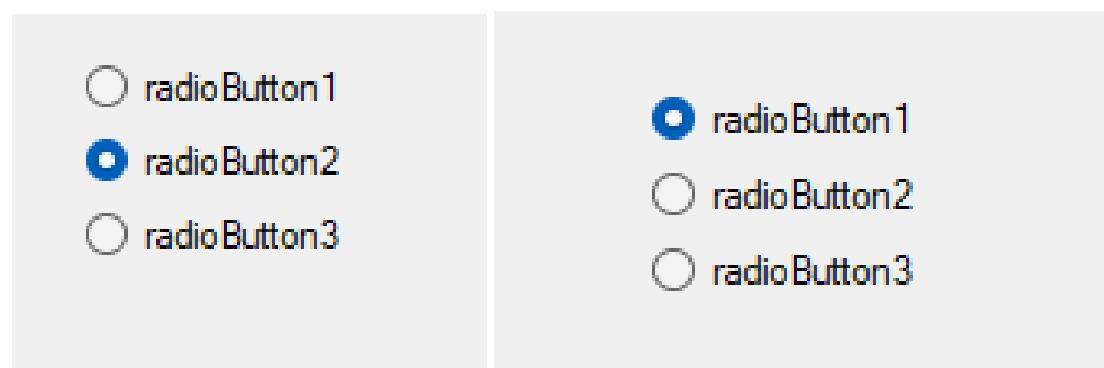
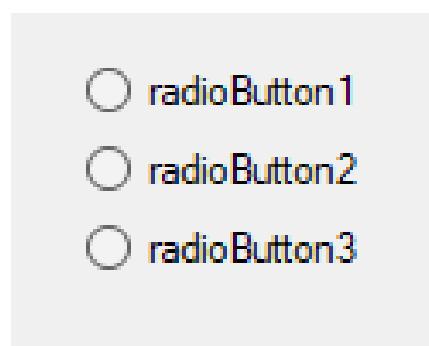


طيب عرفنا كده انه ال check box بتقدر تختار اكتر من خيار في نفس الوقت

طيب لو عاوز اختيار خيار واحد بس اعم ايه؟

قالاك فيه حاجه اسمها radio button

عاوزين نعمل 3 منهم طيب ونجرب



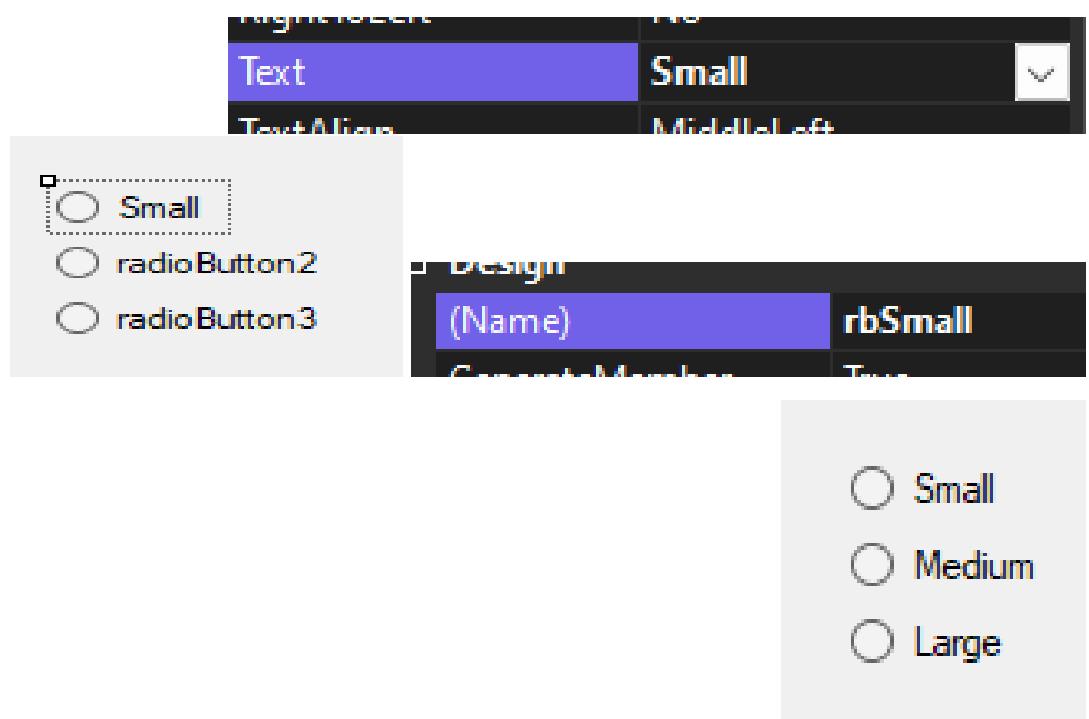
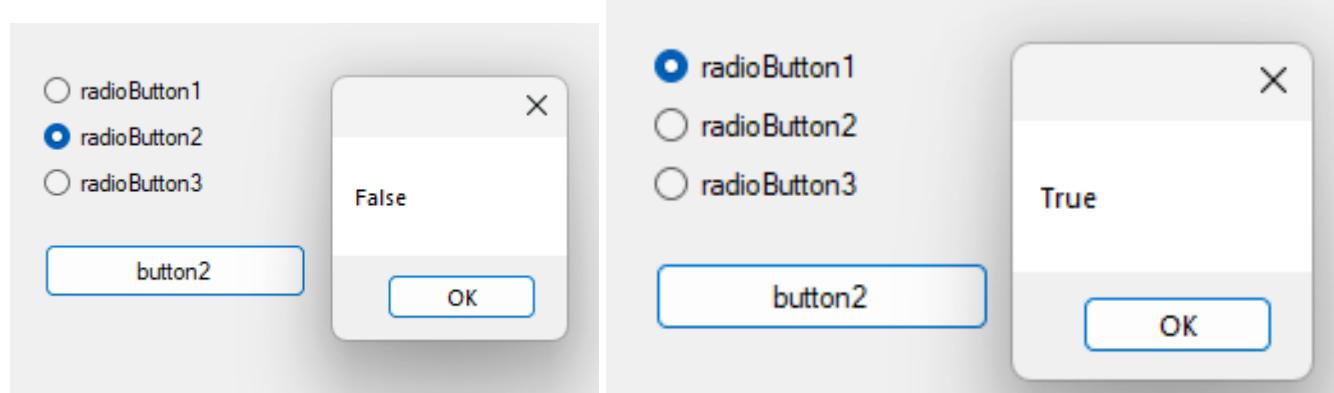
هنعمل زرار تاني يعرفنا انه ال radio button رقم 1 ان كان enabled ولا disabled



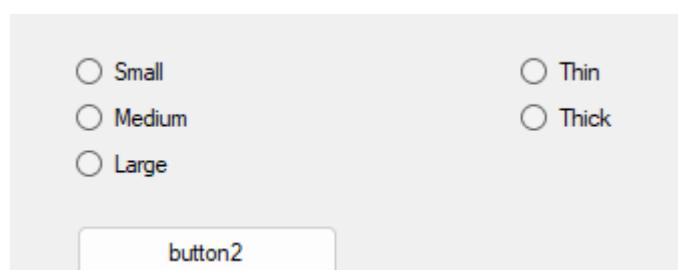
```

private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show(radioButton1.Checked.ToString());
}

```

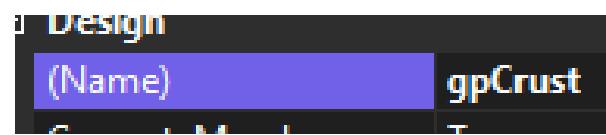
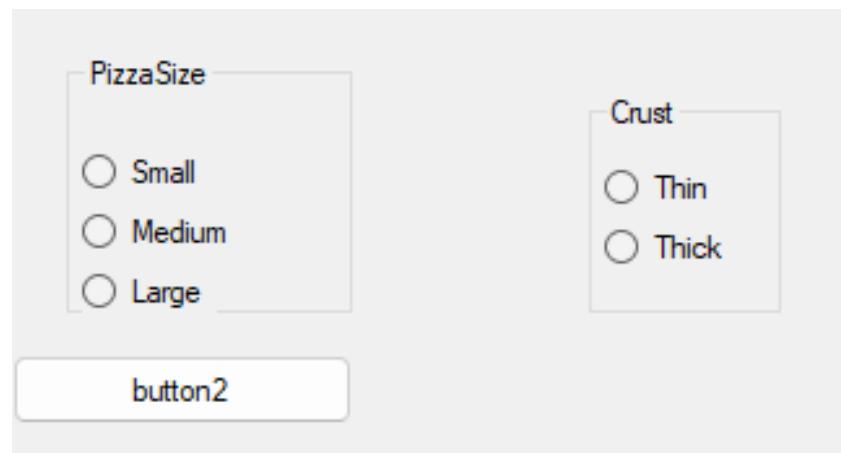
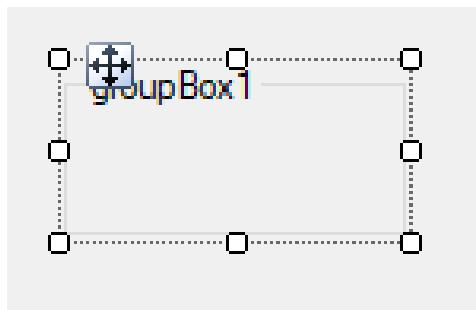


طيب افرض مثلا عاوز اعمل مجموعه تانيه بتعبر مثلا عن عجينة البيتزا عاوزها رفيعه ولا تخينه

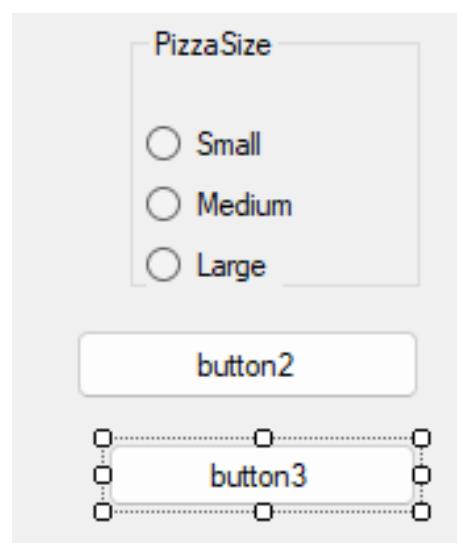


لو شغلتهم علي كده البرنامج مش هيعرف ان دول لوحدهم ودول لوحدهم فلازم اعمل جروب يضم  
ال حاجات الي ليها علاقه بعض

الجروب ده اسمه **group box** بيكون ليه **text** خاص بييه باخد ال **radio buttons** قص ولصق جوه  
ال **group box** ده



هنا بيقولك برض ان ال group box ليه properties زى انك تخفيه وتنظهره او تعمله disable



```

private void button3_Click(object sender, EventArgs e)
{
    gpSize.Enabled = false;
}

```



أي control موجود عندك فيه خاصيه اسمها tag وده عباره عن متغير مخفي تقدر تحفظ فيه داتا عشان لو احتاجتها للبرمجه بعد كده  
مثلا انت عايز تعمل login وتحط في ال tag بتابع ال password كلمة السر الحقيقيه وتقارنها باللي اليوزر دخلها

## Pizza Project Requirements

ارجو من الجميع عمل مشروع البيتزا كما جاء في أول درس

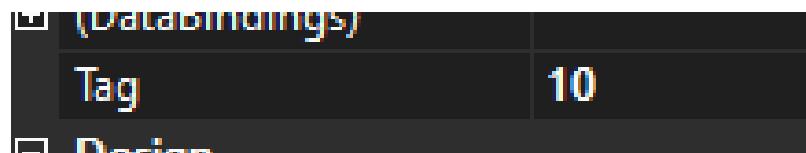
### Part 1

Size	Toppings	Order Summary	
<input type="radio"/> Small	<input checked="" type="checkbox"/> Extra Chees	<input type="checkbox"/> Onion	<b>Size:</b> Large
<input checked="" type="radio"/> Medium	<input type="checkbox"/> Mushrooms	<input type="checkbox"/> Olives	<b>Toppings:</b>
<input type="radio"/> Large	<input checked="" type="checkbox"/> Tomatoes	<input type="checkbox"/> Green Peppers	Extra Chees, Tomatos
Crust Type	Where To Eat	Crust Type: Think Crust Where to Eat: Take Out.	
<input checked="" type="radio"/> Thin Crust	<input type="radio"/> Eat In <input checked="" type="radio"/> Take Out	Total Price: \$50	
<input type="radio"/> Think Crust			

## Pizza Project (Solution)

جيننا عند كل Radio Button وعند كل check box حطينا سعر في ال tag بتابعه

ماعدا ال radio buttons بتو ع where to eat لأننا مش عاملين لهم سعر يعني مثلا عند الحجم ال small حطينا في ال tag بتاعه 20 وال medium 30 وال large 40 وجينا عند ال thin crust وكتبنا في ال tag بتاعه 0 وال thick 10 عند كل نوع topping حطينا 5 في ال tag



ال tag بيتحزن في شكل string وعشان تستخدمه لازم تحوله لو كان رقم هوا هنا استخدم السطر ده في التحويل عشان يحول ل float  
Convert.ToString()

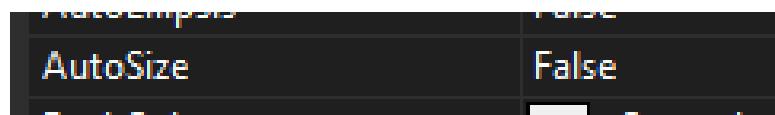
طيب قبل ماتبدأ تشتعل فكر كويس في انك عاوز تكتب كود منظم وسهل القراءه وتطبق فيه سياسة فرق تسد

فالحنا دلوقتي عاوزين نعمل function تحسبنا السعر بتاعت الاوردر طيب السعر ده هيكون مجموع الاختيارات اللي اليوزر اختارها من كل group يبقى نعمل لكل جروب function لوحدها تحسبله السعر بتاعه ملحوظه:-

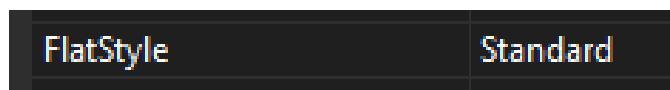
هنا احنا عاملين ال label بتاع ال topping كبير عشان يقدر ينزل سطر جديد



بتقدر تعمل ده عن طريق خاصيتين واحده اسمها auto size والدي بتعملها false عشان مايخليش حجم ال label على قد ال text



تاني خاصيه انك تعمل ال standard style تخليه flat style عشان ينزل سطر جديد



```
float GetSelectedSizePrice()
{
    if (rbSmall.Checked)
        return Convert.ToString(rbSmall.Tag);
```

هنا بنحسب السعر بتاع ال size فبقوله لو اختار ال radio button ده هيرجع السعر اللي في ال tag بتاعه بعد

مايحله

طبعا هنا لازم اليوزر يختار اختيار واحد  
ماينفعش اكتر من واحد لانه radio  
button

```
else if (rbMedium.Checked)  
  
    return Convert.ToSingle( rbMedium.Tag);  
  
else  
    return Convert.ToSingle( rbLarge.Tag);  
  
}  
  
float GetSelectedCrutPrice()
```

```
{  
    if (rbThinCrust.Checked)  
  
        return Convert.ToSingle( rbThinCrust.Tag);  
  
    else  
        return Convert.ToSingle(rbThickCrust.Tag);  
  
}
```

```
float CalculateToppingsPrice()  
{  
  
    float ToppingsTotalPrice = 0;  
  
    if (chkExtraChees.Checked)  
    {  
        ToppingsTotalPrice += Convert.ToSingle( chkExtraChees.Tag) ;  
    }  
  
    if (chkOnion.Checked)  
    {  
        ToppingsTotalPrice += Convert.ToSingle(chkOnion.Tag);  
    }  
  
    if (chkMushrooms.Checked)  
    {  
        ToppingsTotalPrice += Convert.ToSingle(chkMushrooms.Tag);  
    }  
  
    if (chkOlives.Checked)  
    {  
        ToppingsTotalPrice += Convert.ToSingle(chkOlives.Tag);  
    }  
  
    if (chkTomatos.Checked)  
    {  
        ToppingsTotalPrice += Convert.ToSingle(chkTomatos.Tag);  
    }  
  
    if (chkGreenPeppers.Checked)  
    {  
        ToppingsTotalPrice += Convert.ToSingle(chkTomatos.Tag);  
    }  
  
    return ToppingsTotalPrice;
```

هنا بنفس الطريقة بيحسب سعر ال  
crust

هنا بيحسب سعر ال toppings كلها  
فيعمل متغير من النوع float وبيبدأ  
يوزد عليه قيمة ال tag بتاع كل  
check box

وطبعا عشان ده check box فالليوزر  
يقدر يختار اكتر من عنصر او  
مايختارش خالص

<pre> }  float CalculateTotalPrice() {     return GetSelectedSizePrice() + CalculateToppingsPrice() + GetSelectedCrutPrice(); }  void UpdateTotalPrice() {     lblTotalPrice.Text = "\$" + CalculateTotalPrice().ToString(); }  void UpdateSize() {     UpdateTotalPrice();      if (rbSamll.Checked)     {         lblSize.Text = "Small";         return;     }      if (rbMedium.Checked)     {         lblSize.Text = "Medium";         return;     }      if (rbLarge.Checked)     {         lblSize.Text = "Large";         return;     } }  void UpdateCrust() {     UpdateTotalPrice();     if (rbThinCrust.Checked)     {         lblCrustType.Text = "Thin Crust";         return;     }      if (rbThickCrust.Checked)     {         lblCrustType.Text = "Thick Crust";         return;     } }  void UpdateWhereToEat() {     UpdateTotalPrice();      if (rbEatIn.Checked) </pre>	<p>و هنا بحسب السعر الإجمالي يدوب بجمع اللي راجع من ال functions</p> <p>هنا بنعدل على ال label بتاع السعر</p> <p>هنا بنعدل عال label بتاع ال size ومعاه بنعدل السعر</p> <p>هنا بنعدل عال label بتاع ال crust والسعر</p> <p>هنا بنعدل عال label بتاع where to eat والسعر</p>
--	---

```

    {
        lblWhereToEat.Text = "Eat In.";
        return;
    }

    if (rbTakeOut.Checked)
    {
        lblWhereToEat.Text = "Take Out.";
        return;
    }
}

```

```
void UpdateToppings()
```

```
{
```

```
    UpdateTotalPrice();
```

```
    string sToppings = "";
```

```
    if (chkExtraChees.Checked)
```

```
{
```

```
    sToppings += " Extra Chees";
```

```
}
```

```
    if (chkOnion.Checked)
```

```
{
```

```
    sToppings += ", Onion";
```

```
}
```

```
    if (chkMushrooms.Checked)
```

```
{
```

```
    sToppings += ", Mushrooms";
```

```
}
```

```
    if (chkOlives.Checked)
```

```
{
```

```
    sToppings += ", Olives";
```

```
}
```

```
    if (chkTomatos.Checked)
```

```
{
```

```
    sToppings += ", Tomatos";
```

```
}
```

```
    if (chkGreenPeppers.Checked)
```

```
{
```

```
    sToppings += ", Green Peppers";
```

```
}
```

```
    if (sToppings.StartsWith(", "))
```

```
{
```

```
    sToppings = sToppings.Substring(1, sToppings.Length - 1).Trim();
```

```
}
```

```
    if (sToppings == "")
```

```
        sToppings = "No Toppings";
```

```
lblToppings.Text = sToppings;
```

هنا احنا بنعدل عال **label** بتاع ال **toppings** وبدل مانقعد كل شويه نشوف ايه اللي مكتوب ونشيله لا احنا كل مره نكتبهم من الأول ونريح دماغنا من وجع القلب ده

وفي الاخر خالص بيقوله لو كان فيه فاصله في البدايه شيلها

<pre> } void UpdateOrderSummary() {     UpdateSize();     UpdateToppings();     UpdateCrust();     UpdateWhereToEat();     UpdateTotalPrice(); } </pre>	<p>وهنا احنا مجمعين كل ال functions عشان لما يفتح الفورم يستدعيها فيعدل على كل حاجه كانه بيعمل initialization للفورم</p>
<pre> void ResetForm() {     //reset Groups     gbSize.Enabled = true;     gbToppings.Enabled = true;     gbCrustType.Enabled = true;     gbWhereToEat.Enabled = true;      //reset Size     rbMedium.Checked = true;      //reset Toppings.     chkExtraChees.Checked = false;     chkOnion.Checked = false;     chkMushrooms.Checked = false;     chkOlives.Checked = false;     chkTomatos.Checked = false;     chkGreenPeppers.Checked = false;      //reset CrustType     rbThinCrust.Checked = true;      //reset Where to Eat     rbEatIn.Checked = true;      //Reset Order Button     btnOrderPizza.Enabled = true; } </pre>	<p>وهنا عشان يرجع الاختيارات للوضع الأصلي بتاعها</p>

## نيجي بقى لـ events

<pre> private void btnOrderPizza_Click(object sender, EventArgs e) {     if (MessageBox.Show("Confirm Order", "Confirm",         MessageBoxButtons.OKCancel,         MessageBoxIcon.Question) == DialogResult.OK)     {         MessageBox.Show("Order Placed Successfully",         "Success",         MessageBoxButtons.OK,         MessageBoxIcon.Information);          btnOrderPizza.Enabled = false;         gbSize.Enabled = false;         gbToppings.Enabled = false;     } } </pre>	<p>ده ال event بتاع زرار ال order pizza مجرد انه بيعمل message box ولو اليوزر داس ok هيعمل disable للعناصر</p> <p>ولو داس على cancel هيطلع message box بتقوله update your order تانيه بتقوله</p>
---	--

```

        gbCrustType.Enabled = false;
        gbWhereToEat.Enabled = false;

    }

    else

        MessageBox.Show("Update your order", "Update",
            MessageBoxButtons.OK,
            MessageBoxIcon.Exclamation);
    }

    private void rbMedium_CheckedChanged(object sender,
EventArgs e)
{
    UpdateSize();
}

private void rbLarge_CheckedChanged(object sender,
EventArgs e)
{
    UpdateSize();
}

private void rbSamll_CheckedChanged(object sender,
EventArgs e)
{
    UpdateSize();
}

private void chkExtraChees_CheckedChanged(object
sender, EventArgs e)
{
    UpdateToppings();
}

private void chkOnion_CheckedChanged(object sender,
EventArgs e)
{
    UpdateToppings();
}

private void chkMushrooms_CheckedChanged(object
sender, EventArgs e)
{
    UpdateToppings();
}

private void chkOlives_CheckedChanged(object sender,
EventArgs e)
{
    UpdateToppings();
}

private void chkTomatos_CheckedChanged(object sender,
EventArgs e)
{
    UpdateToppings();
}

private void chckGreenPeppers_CheckedChanged(object
sender, EventArgs e)

```

ودي ال events بتاعت ال radio buttons وال check boxes ببستدعي فيها ال update الخاصه بيها

```

{
    UpdateToppings();
}

private void rbThinCrust_CheckedChanged(object sender,
EventArgs e)
{
    UpdateCrust();
}

private void rbThickCrust_CheckedChanged(object
sender, EventArgs e)
{
    UpdateCrust();
}

private void rbEatIn_CheckedChanged(object sender,
EventArgs e)
{
    UpdateWhereToEat();
}

private void rbTakeOut_CheckedChanged(object sender,
EventArgs e)
{
    UpdateWhereToEat();
}

```

وهنا ده الحدث بتاع الفورم لما يحمل اسمه  
on load

```

private void Form1_Load(object sender, EventArgs e)
{
    UpdateOrderSummary();
}

private void btnResetForm_Click(object sender, EventArgs
e)
{
    ResetForm();
}

```

وده الحدث بتاع زرار ال reset

## More about TextBox : Focus, MaxLength, Password, RightToLeft, Locked,Multiline, WordWrap

هنتكلم عن properties تانيه

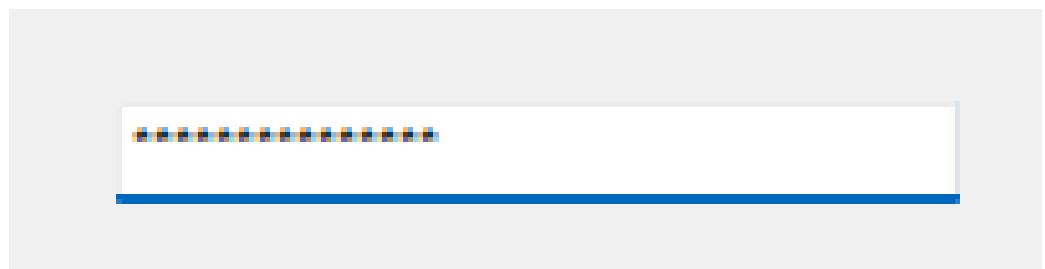
هنعمل فورم جديد نشرح فيه



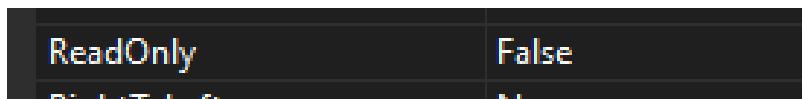
و هنعمل فيه text box جديد



فيه في ال properties حاجه اسمها `password char` ودي بتحط فيها أي رمز ولما تشغيل البرنامج وتحجي تكتب هتلaci الحروف كلها طلعت بالرمز ده



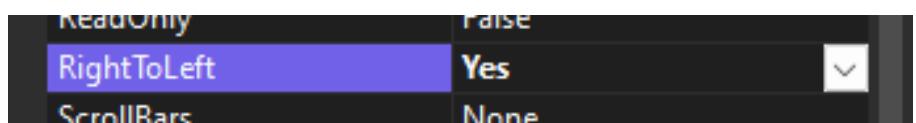
خاصية ال `read only` بتخليله تسمح او تمنع الكتابه في ال `textbox`



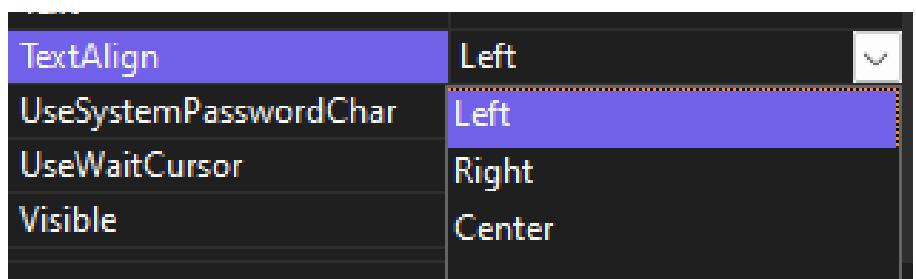
خاصية ال `max length` بتخليله تحدد عدد الحروف اللي اليوزر يقدر يدخلها



خاصية ال `right to left` لو عايز تدعم الكتابه بالعربي بتخليل اتجاه النص من اليمين لليسار  
لو اخترت `inherit` هيأخذ قيمته من ال `parent` بتاعه اللي هو الفورم حاليا  
والفورم فيه نفس الخاصية



ال `text alignment` بتخليله تختار يالما تخلي النص في اول ال `text box` او في اخره او في الوسط  
ومالوش علاقه بال `right to left`



ال `text box` الافتراضي بتاعه انه يكون سطر واحد تقدر تخليله اكتر من سطر من هنا

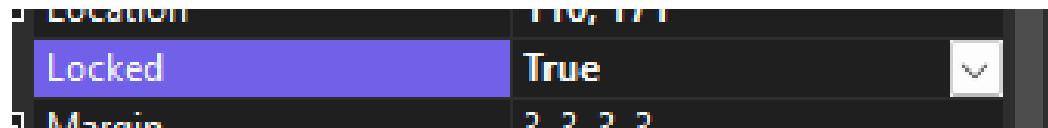


ال `word wrap` بتديلك خيار انك تخلي الكلام كله علي سطر واحد او لما تنتهي مساحة السطر ينزل السطر اللي بعده حسب مساحة ال `text box`

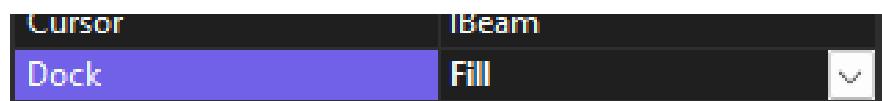
WordWrap

True

الخاصية `lock` بتخلي مكان العنصر ثابت مايتحرکش وانت بتعمل ال design



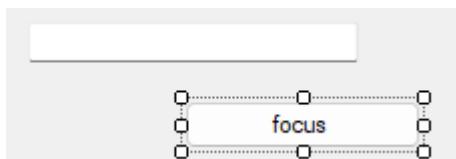
خاصية ال `dock` بتخلي مساحة ال `textbox` من مساحة العنصر اللي هيا موجوده جواه يعني لو محظوظ في `group box` بيأخذ مساحة ال `group box` كلها ولو محظوظ في فورم بيأخذ مساحة الفورم كلها



وبتقدر تختار انت عايزه يشغل أي مساحة



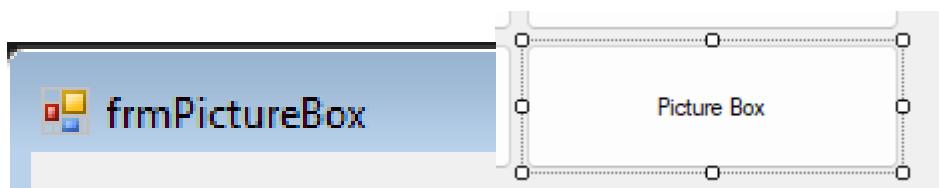
لو عايز أخلي ال `focus` بالكود



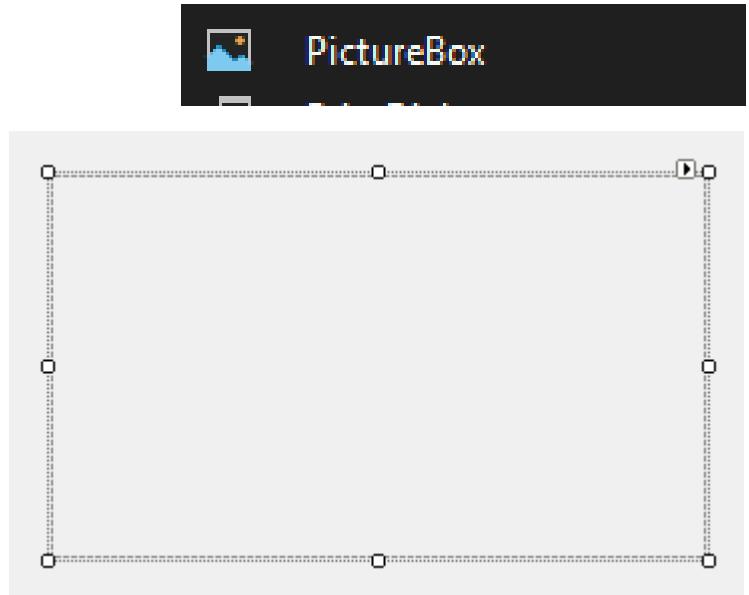
```
private void button1_Click(object sender, EventArgs e)
{
    textBox7.Focus();
}
```

## PictureBox

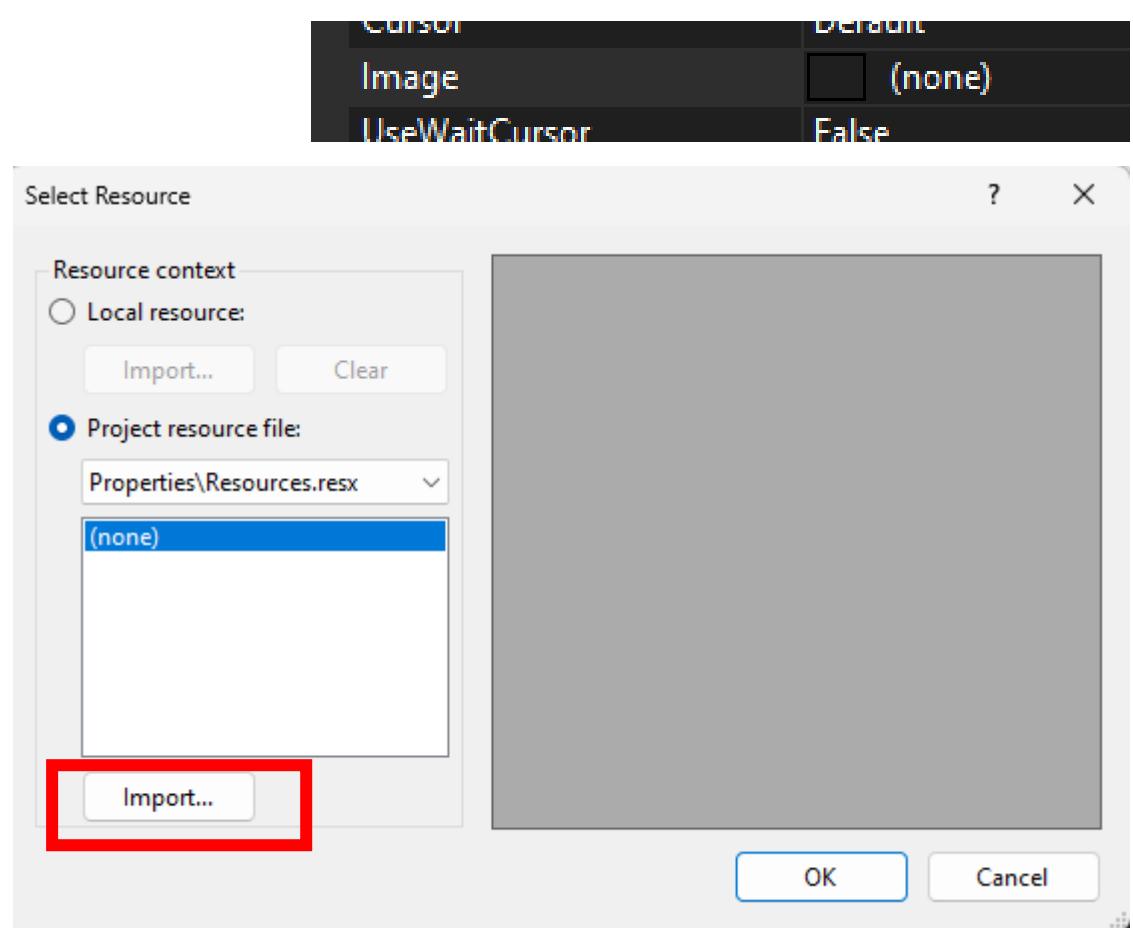
هنعمل `form` جديد



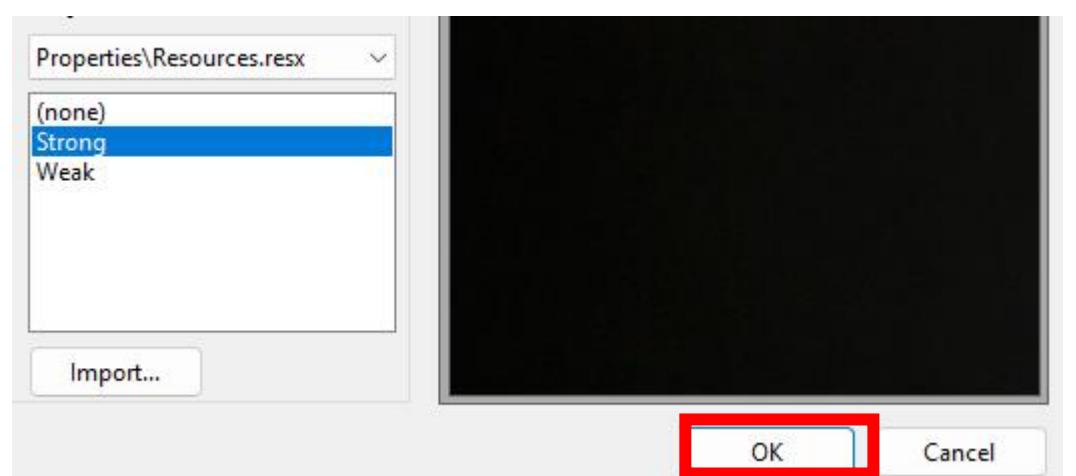
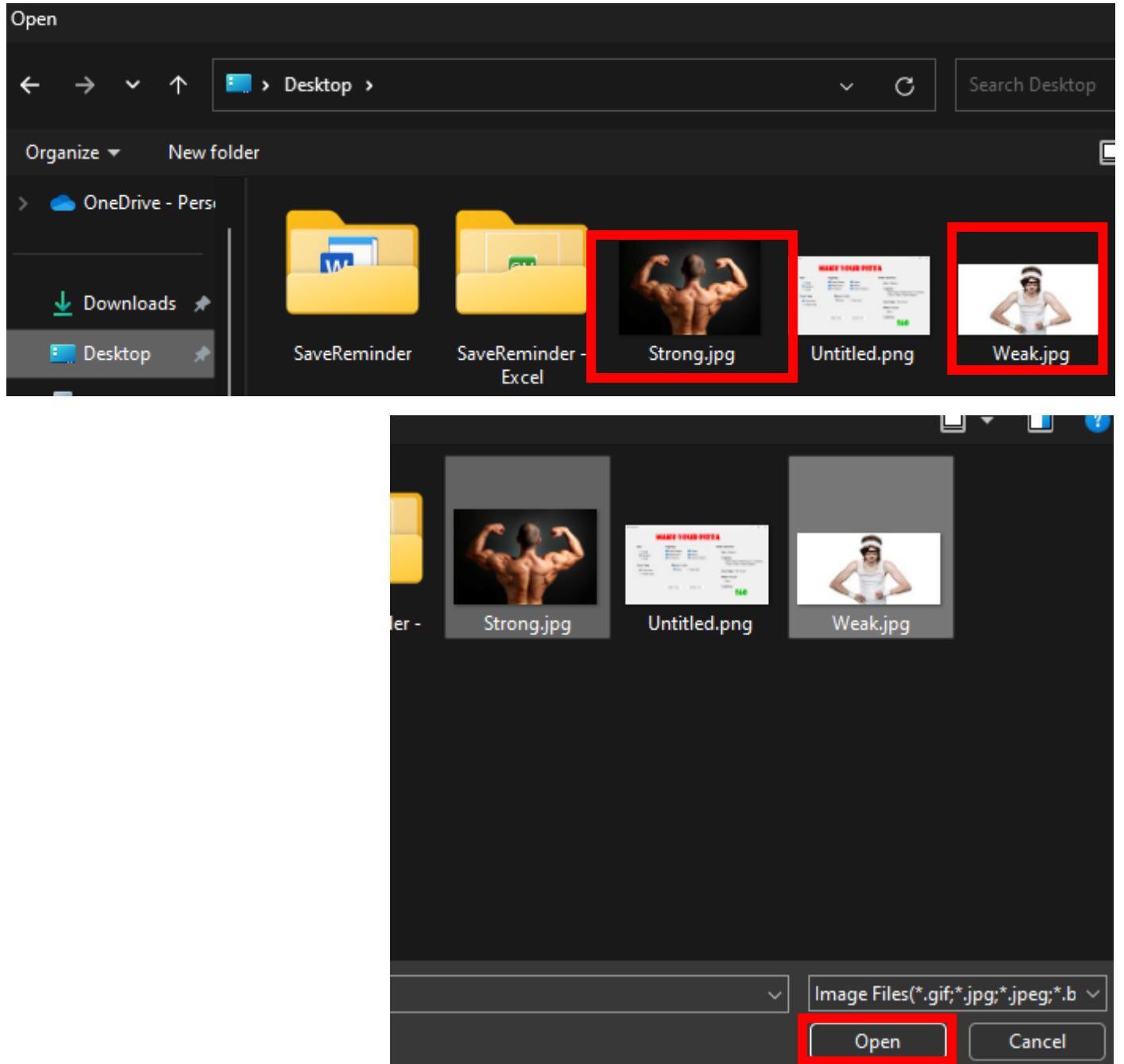
من ال `tool box` هنختار `picture box`



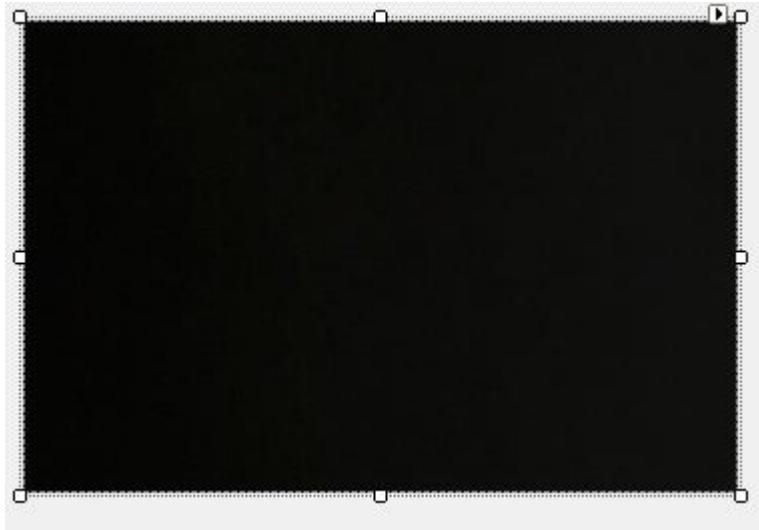
دلوقي فيه خاصيه اسمها `image` ودي بتضيف الصوره في ال `picture box` project resource file من ملف عندك او من هنختار اننا نجيها من ال `project resource file` لانه اسهل



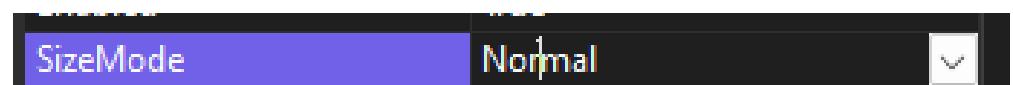
جيبيت صورتين وسميتهم وحطتيهم عالديسكتوب والاسم اللي مسميهم بيده اللي هستخدمه برمجيا



كده الصور اتنقلت لملف اسمه resources في البرنامج



عشان نضبط الصوره فيه خاصيه اسمها size mode

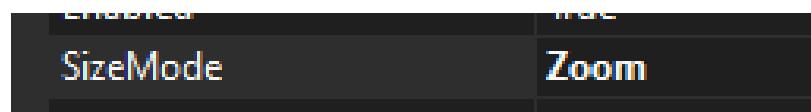


لو اخترت autosize هيخليلها بحجمها اللي هيا عليه

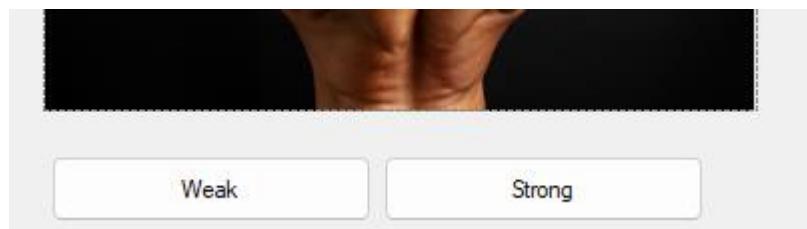
لو اخترت center image هيحطهالك في النص

لو اخترت stretch image يضبط ابعاد الصوره على ابعاد ال picture box اللي عندك

لو اخترت zoom بتخلصي الصوره تكتب وتصغر حسب حجم ال picture box



طيب انا عايز اعمل زرار يغير الصوره لصوره ال weak



```
private void button1_Click(object sender, EventArgs e)
{
    pictureBox1.Image = Resources.Weak;
}

private void button2_Click(object sender, EventArgs e)
{
    pictureBox1.Image = Resources.Strong;
}
```



Weak

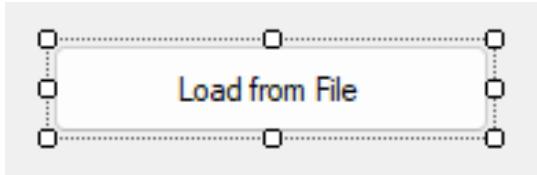
Strong



Weak

Strong

دلوقي عايزين نحمل الصورة من ال resources c drive مش من ال  
لو حملت الصور كلها في ال resources هتكبر حجم البرنامج بس هيكون اسرع في الحميل ولو عندك  
صور كتير ماتحطهاش كلها في ال resources



```
private void button3_Click(object sender, EventArgs e)
{
    pictureBox1.Image = Image.FromFile(@"C:\Users\Ahmed\Desktop\QUESTION.png");
}
```

لما تحط صوره حطها بصيغة PNG عشان ماتعملكش مشاكل

## PictureBox Exercise



GIRL



Boy  Girl  Book  Pen

## PictureBox Exercise Solution

هوا هنا عامل ال exercise ده عشان يعرفك على ال sender ده

```

private void rbPen_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Resources.Pen;
    lblTitle.Text = ((RadioButton)sender).Tag.ToString();
}

```

ال ده زی ما هو مكتوب هوا عباره عن object ده هوا ال control اللي جاي منه ال event يعني لو ال event معمول علي زرار معين ال sender ده بيكون الزرار نفسه وبالتالي ممكن تعمل عليه عمليات من غير ماتستدعي الزرار وتروح تشووف اسمه ايه والحوارات دي في حالتنا هنا احنا استخدمنا ال sender في اننا نأخذ الداتا اللي موجوده في ال tag بتاعه

```

private void rbBoy_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Resources.Boy;
    lblTitle.Text = ((RadioButton)sender).Tag.ToString();
}

private void rbGirl_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Resources.Girl;
    lblTitle.Text = ((RadioButton)sender).Tag.ToString();
}

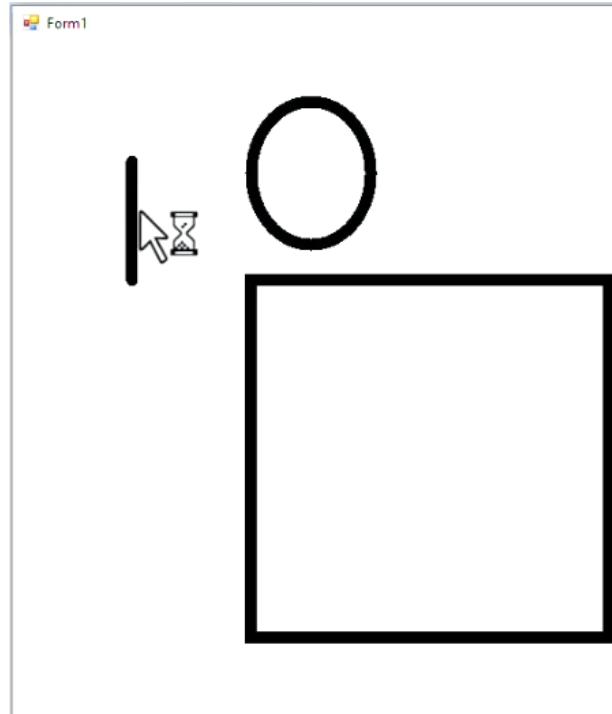
private void rbBook_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Resources.Book;
    lblTitle.Text = ((RadioButton)sender).Tag.ToString();
}

private void rbPen_CheckedChanged(object sender, EventArgs e)
{
    pictureBox1.Image = Resources.Pen;
    lblTitle.Text = ((RadioButton)sender).Tag.ToString();
}

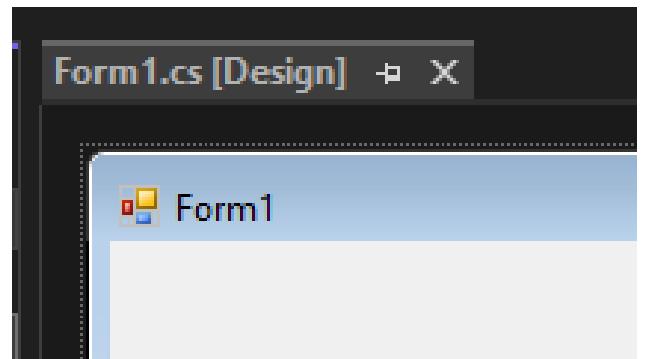
```

### **Draw Line , ellipse , and rectangle.**

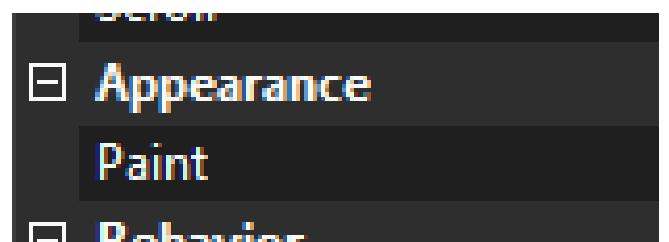
هنا بيقولك انك تقدر ترسم عالفورم



دلوقي عشان ترسم محتاج قلم فبتعرف object من النوع قلم وبتعرف سمك الخط واللون بتوعه  
وبتقول للقلم عن طريق الكود انه يرسم سواء دائيره او خط او اي حاجه  
عشان ترسم خط مثلا انت محتاج تحديد القلم اللي هترسم بيها ونقطه البدايه ونقطة النهايه  
نقطة البدايه بتتحدد بال x وال y ال x هوa بعد النقطه عن الاطار من اليسار وال y هوa بعد النقطه من  
الاطار من اعلي  
وكذلك نقطه النهايه  
هنعمل فورم جديد



وبعدين نحدد الفورم ونروح لـ events بتاعته هنلاقي اسمه paint event



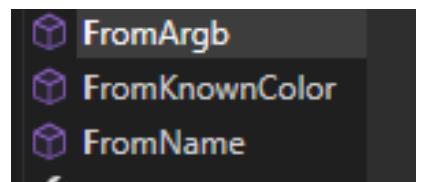
بتعمل double click عليه بتوديك للكود بتاعه

بعدين بنكتب الكود بتاعك :-

اول حاجه بتعرف object من النوع Color وده هيكون لون الخط وتقول يساوي  
هيطلب منك ارقام من صفر ل 255 بيمثوا اللون زي كده

```
Color Black = Color.FromArgb(255, 0, 0, 0);
```

وتقدر تختار مابين الوان موجوده



```
Color Black = Color.FromKnownColor(KnownColor.Black);
```

بعدين بتعرف object من النوع Pen وال constructor بتاعه بيأخذ اللون

```
Pen Pen = new Pen(Black);
```

بعدين بتحدد سمك الخط عن طريق width اسمها بتاخذ رقم

```
Pen.Width = 10;
```

لو عايزيين يرسم خط متقطع او منقط بنكتب السطر ده لو مش عايزيين هيعمل خط عادي

```
Pen.DashStyle = System.Drawing.Drawing2D.DashStyle.Dash;
```

لو بنرسم خط فيه اوبشن هنا بيحدد شكل نقطة البدايه والنهايه انك عاوزها تكون رأس سهم او دائره او غيره

احنا هنا عملناها مدوره

```
Pen.StartCap = System.Drawing.Drawing2D.LineCap.Round;  
Pen.EndCap = System.Drawing.Drawing2D.LineCap.Round;
```

تقدر تغير الشكل عن طريق انك تشيل كلمة round وتقلب وتشوف وتجرب

طيب خلينا نرسم بقى :-

عشان نرسم هناخد ال object ده ونستدعي منه ال functions بتاعت الرسم

```
private void Form1_Paint(object sender, PaintEventArgs e)  
{  
    Color Black = Color.FromArgb(255, 0, 0, 0);
```

أي function بتأخذ خمسه parameters القلم والمحورين بتوغ نقطة البدايه والمحورين بتوغ نقطة النهايه

```
e.Graphics.DrawLine(Pen, 100, 100, 100, 200);
e.Graphics.DrawRectangle(Pen, 200, 200, 300, 300);
e.Graphics.DrawEllipse(Pen, 200, 50, 100, 120);
```

وده الكود بتاع ال event

```
private void Form1_Paint(object sender, PaintEventArgs e)
{
    Color Black = Color.FromArgb(255,0,0,0);

    Pen pen = new Pen(Black);

    pen.Width = 10;

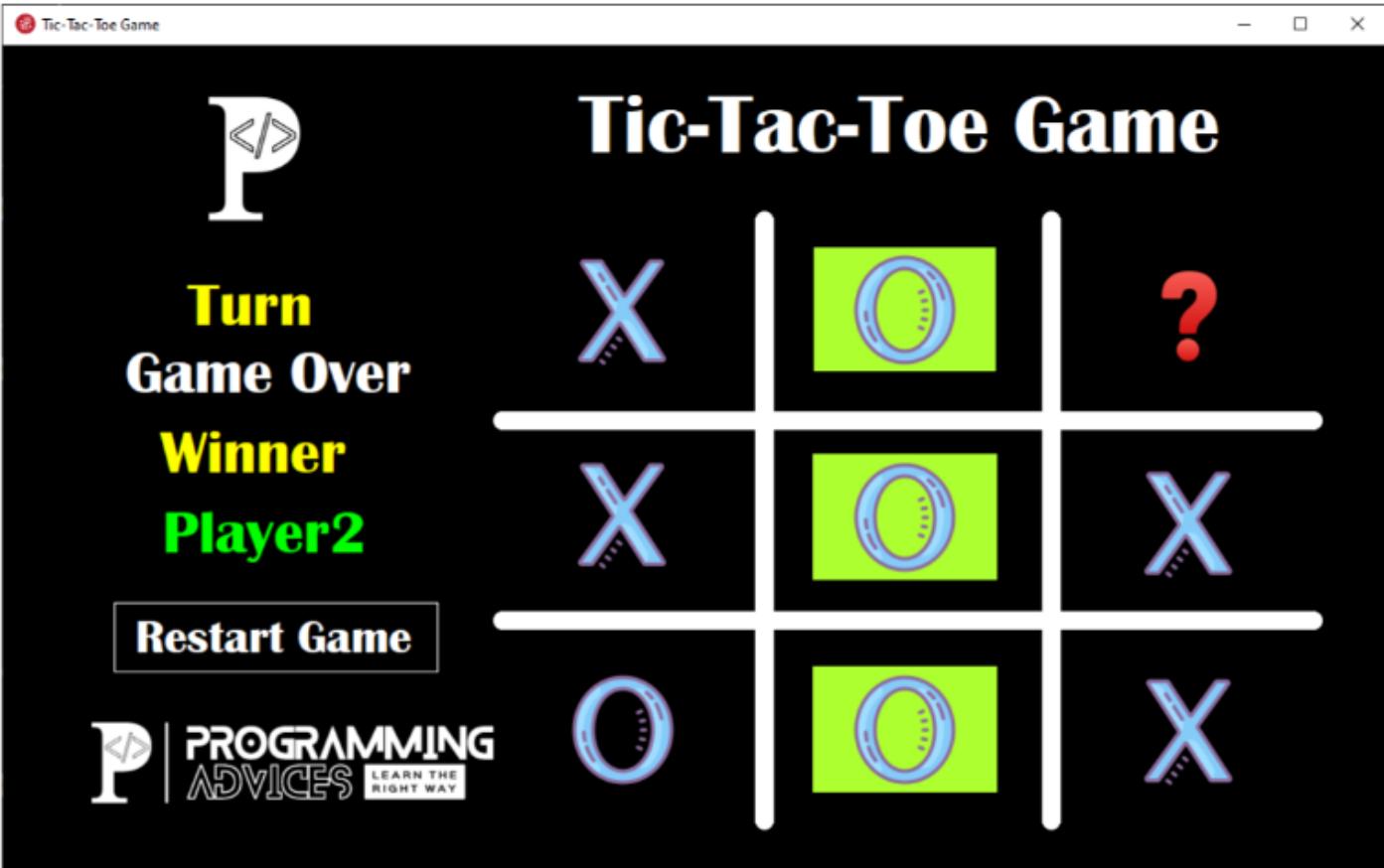
    // pen.DashStyle=System.Drawing.Drawing2D.DashStyle.Dash;

    pen.StartCap = System.Drawing.Drawing2D.LineCap.Round;
    pen.EndCap = System.Drawing.Drawing2D.LineCap.Round;

    e.Graphics.DrawLine(pen,100,100,100,200);
    e.Graphics.DrawRectangle(pen,200,200,300,300);
    e.Graphics.DrawEllipse(pen,200,50,100,120);
}
```

## Tic-Tac-Toe Game

### Requirement

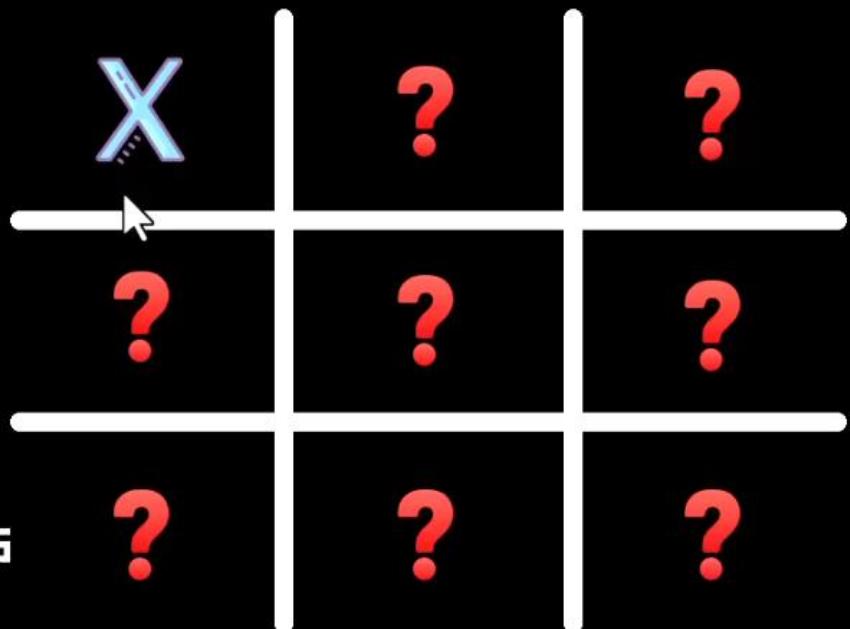


P  
⟨⟩

Turn  
Player 2  
Winner  
In Progress

**Restart Game****P** | PROGRAMMING  
ADVICES LEARN THE  
RIGHT WAY

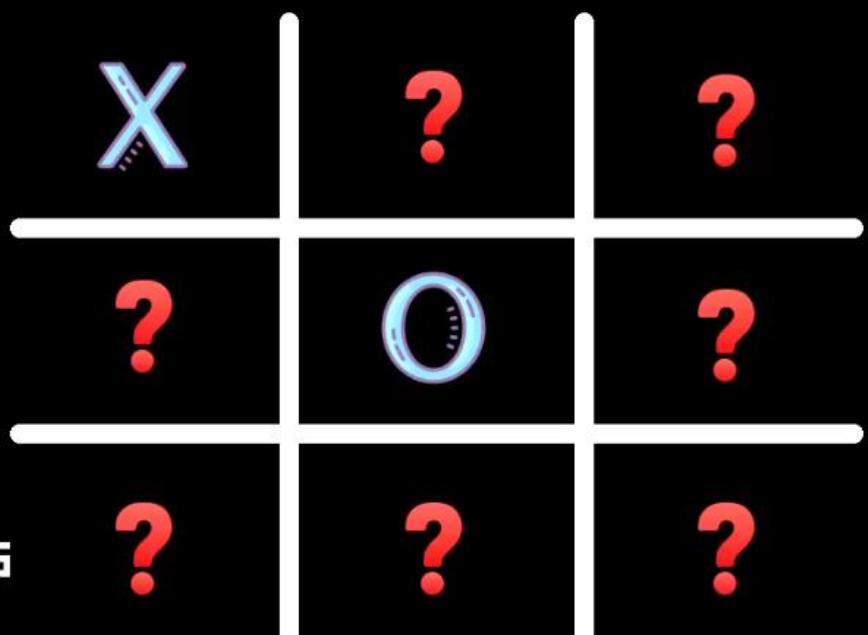
# Tic-Tac-Toe Game

P  
⟨⟩

Turn  
Player 1  
Winner  
In Progress

**Restart Game****P** | PROGRAMMING  
ADVICES LEARN THE  
RIGHT WAY

# Tic-Tac-Toe Game



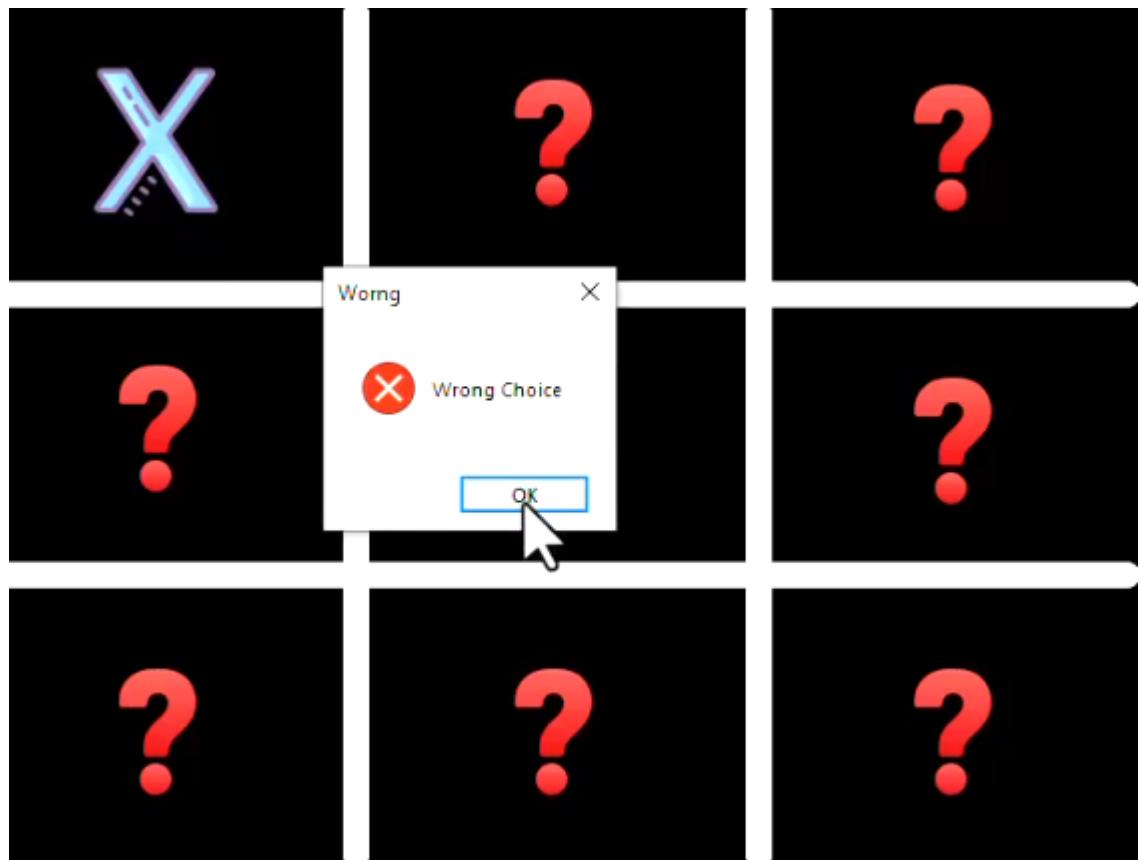
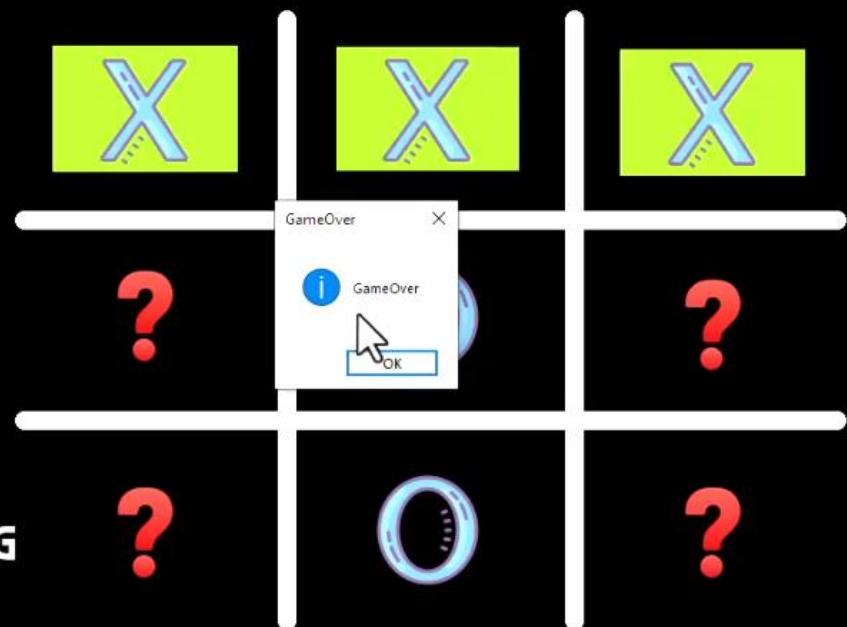
P  
⟨ ⟩

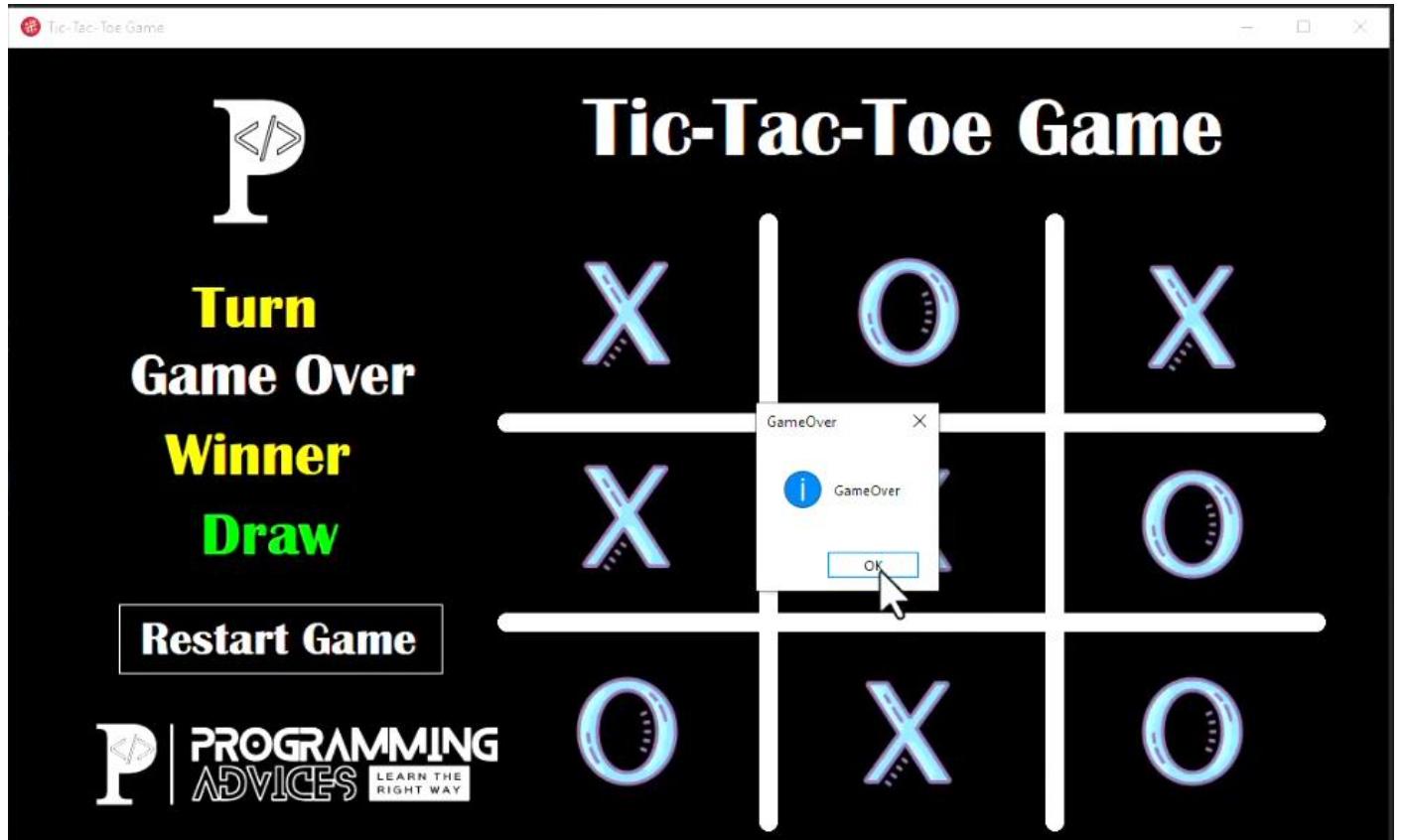
# Tic-Tac-Toe Game

Turn  
Game Over

Winner  
Player1

Restart Game





## Solution

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Runtime.InteropServices.WindowsRuntime;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Xml.Linq;
using XO_Game_Final.Properties;

namespace XO_Game_Final
{
    public partial class Form1 : Form
    {

        stGameState GameStatus;
        enPlayer PlayerTurn = enPlayer.Player1;

        enum enPlayer
        {
            Player1,
            Player2
        }

        enum enWinner
        {
            Player1,
            Player2,
        }
    }
}
```

```

        Draw,
        GameInProgress
    }

    struct stGameStatus
    {
        public enWinner Winner;
        public bool GameOver;
        public short PlayCount;
    }

}

public bool CheckValues(Button btn1, Button btn2, Button btn3)
{
    if (btn1.Tag.ToString() != "?" && btn1.Tag.ToString() == btn2.Tag.ToString() && btn1.Tag.ToString() == btn3.Tag.ToString())
    {
        btn1.BackColor = Color.GreenYellow;
        btn2.BackColor = Color.GreenYellow;
        btn3.BackColor = Color.GreenYellow;

        if (btn1.Tag.ToString() == "X")
        {
            GameStatus.Winner = enWinner.Player1;
            GameStatus.GameOver = true;
            EndGame();
            return true;
        }
        else
        {
            GameStatus.Winner = enWinner.Player2;
            GameStatus.GameOver = true;
            EndGame();
            return true;
        }
    }

    GameStatus.GameOver = false;
    return false;
}

void EndGame()
{
    lblTurn.Text = "Game Over";
    switch (GameStatus.Winner)
    {
        case enWinner.Player1:

```

```
        lblWinner.Text = "Player1";
        break;

    case enWinner.Player2:

        lblWinner.Text = "Player2";
        break;

    default:

        lblWinner.Text = "Draw";
        break;

    }

    MessageBox.Show("GameOver", "GameOver", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

public void CheckWinner()
{

    //checked rows
    //check Row1
    if (CheckValues(button1, button2, button3))
        return;

    //check Row2
    if (CheckValues(button4, button5, button6))
        return;

    //check Row3
    if (CheckValues(button7, button8, button9))
        return;

    //checked cols
    //check col1
    if (CheckValues(button1, button4, button7))
        return;

    //check col2
    if (CheckValues(button2, button5, button8))
        return;

    //check col3
    if (CheckValues(button3, button6, button9))
        return;

    //check Diagonal

    //check Diagonal1
    if (CheckValues(button1, button5, button9))
        return;

    //check Diagonal2
    if (CheckValues(button3, button5, button7))
        return;
}
```

```
}

public void Changelimage(Button btn)
{
    if (btn.Tag.ToString() == "?")
    {
        switch (PlayerTurn)
        {
            case enPlayer.Player1:
                btn.Image = Resources.X;
                PlayerTurn = enPlayer.Player2;
                lblTurn.Text = "Player 2";
                GameStatus.PlayCount++;
                btn.Tag = "X";
                CheckWinner();
                break;
            case enPlayer.Player2:
                btn.Image = Resources.O;
                PlayerTurn = enPlayer.Player1;
                lblTurn.Text = "Player 1";
                GameStatus.PlayCount++;
                btn.Tag = "O";
                CheckWinner();
                break;
        }
    }
    else
    {
        MessageBox.Show("Wrong Choice", "Wrong", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }

    if (GameStatus.PlayCount == 9)
    {
        GameStatus.GameOver = true;
        GameStatus.Winner = enWinner.Draw;
        EndGame();
    }
}

public Form1()
{
    InitializeComponent();
}

private void button1_Click(object sender, EventArgs e)
{
    Changelimage(button1);
}

private void button2_Click(object sender, EventArgs e)
{
    Changelimage(button2);
}
```

```
private void button3_Click(object sender, EventArgs e)
{
    Changelimage(button3);
}

private void button6_Click(object sender, EventArgs e)
{
    Changelimage(button6);
}

private void button5_Click(object sender, EventArgs e)
{
    Changelimage(button5);
}

private void button4_Click(object sender, EventArgs e)
{
    Changelimage(button4);
}

private void button9_Click(object sender, EventArgs e)
{
    Changelimage(button9);
}

private void button8_Click(object sender, EventArgs e)
{
    Changelimage(button8);
}

private void button7_Click(object sender, EventArgs e)
{
    Changelimage(button7);
}

private void Form1_Load(object sender, EventArgs e)
{
}

private void RestButton(Button btn)
{
    btn.Image = Resources.question_mark_96;
    btn.Tag = "?";
    btn.BackColor = Color.Transparent;
}

private void RestartGame()
{

    RestButton(button1);
    RestButton(button2);
    RestButton(button3);
    RestButton(button4);
    RestButton(button5);
    RestButton(button6);
    RestButton(button7);
    RestButton(button8);
    RestButton(button9);
}
```

```
PlayerTurn = enPlayer.Player1;
lblTurn.Text = "Player 1";
GameStatus.PlayCount = 0;
GameStatus.GameOver = false;
GameStatus.Winner = enWinner.GameInProgress;
lblWinner.Text = "In Progress";

}

private void btnRestart_Click(object sender, EventArgs e)
{
    RestartGame();

}

private void Form1_Paint(object sender, PaintEventArgs e)
{
    Color white = Color.FromArgb(255, 255, 255, 255);
    Pen whitePen = new Pen(white);
    whitePen.Width = 15;
    //whitePen.DashStyle = System.Drawing.Drawing2D.DashStyle.Dash;
    whitePen.StartCap = System.Drawing.Drawing2D.LineCap.Round;
    whitePen.EndCap = System.Drawing.Drawing2D.LineCap.Round;

    //draw Horizontal lines
    e.Graphics.DrawLine(whitePen, 400, 300, 1050, 300);
    e.Graphics.DrawLine(whitePen, 400, 460, 1050, 460);

    //draw Vertical lines
    e.Graphics.DrawLine(whitePen, 610, 140, 610, 620);
    e.Graphics.DrawLine(whitePen, 840, 140, 840, 620);

}
}
```

**Make one Event instead of 9 events**

هذا بيقولك انه في اللعبة تم استدعاء ال change image 9 مرات و عايز بن نستدعيها مره واحده

الفكرة انه اخذ copy من كود زرار من الـ 9

وَقَامَ حَيْ، عَلَيْهِمْ كُلُّهُمْ وَخَلَيْهِمْ الـ `event` الـ `onclick` اسْمُهُ بِنَادِي الـ `الْكُوْدَالِي`، عَلَيْنَاهُ

```
private void button_Click(object sender, EventArgs e)
{
    ChangelImage((Button)sender);
}
```



ChangeValues	
Click	button_Click
ClientSizeChanged	

اصبح ده الكود

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Runtime.InteropServices.WindowsRuntime;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Xml.Linq;
using XO_Game_Final.Properties;

namespace XO_Game_Final
{
    public partial class Form1 : Form
    {

        stGameStatus GameStatus;
        enPlayer PlayerTurn = enPlayer.Player1;

        enum enPlayer
```

```

{
    Player1,
    Player2
}

enum enWinner
{
    Player1,
    Player2,
    Draw,
    GameInProgress
}

struct stGameStatus
{
    public enWinner Winner;
    public bool GameOver;
    public short PlayCount;
}

public bool CheckValues(Button btn1, Button btn2, Button btn3)
{
    if (btn1.Tag.ToString() != "?" && btn1.Tag.ToString() == btn2.Tag.ToString() && btn1.Tag.ToString() ==
    btn3.Tag.ToString())
    {
        btn1.BackColor = Color.GreenYellow;
        btn2.BackColor = Color.GreenYellow;
        btn3.BackColor = Color.GreenYellow;

        if (btn1.Tag.ToString() == "X")
        {
            GameStatus.Winner = enWinner.Player1;
            GameStatus.GameOver = true;
            EndGame();
            return true;
        }
        else
        {
            GameStatus.Winner = enWinner.Player2;
            GameStatus.GameOver = true;
            EndGame();
            return true;
        }
    }

    GameStatus.GameOver = false;
    return false;
}

```

```
void EndGame()
{
    lblTurn.Text = "Game Over";
    switch (GameStatus.Winner)
    {

        case enWinner.Player1:
            lblWinner.Text = "Player1";
            break;

        case enWinner.Player2:
            lblWinner.Text = "Player2";
            break;

        default:
            lblWinner.Text = "Draw";
            break;
    }

    MessageBox.Show("GameOver", "GameOver", MessageBoxButtons.OK, MessageBoxIcon.Information);
}

public void CheckWinner()
{

    //checked rows
    //check Row1
    if (CheckValues(button1, button2, button3))
        return;

    //check Row2
    if (CheckValues(button4, button5, button6))
        return;

    //check Row3
    if (CheckValues(button7, button8, button9))
        return;

    //checked cols
    //check col1
    if (CheckValues(button1, button4, button7))
        return;

    //check col2
    if (CheckValues(button2, button5, button8))
        return;

    //check col3
    if (CheckValues(button3, button6, button9))
        return;

    //check Diagonal
}
```

```

//check Diagonal1
if (CheckValues(button1, button5, button9))
    return;

//check Diagonal2
if (CheckValues(button3, button5, button7))
    return;

}

public void Changelimage(Button btn)
{
    if (btn.Tag.ToString() == "?")
    {
        switch (PlayerTurn)
        {
            case enPlayer.Player1:
                btn.Image = Resources.X;
                PlayerTurn = enPlayer.Player2;
                lblTurn.Text = "Player 2";
                GameStatus.PlayCount++;
                btn.Tag = "X";
                CheckWinner();
                break;
            case enPlayer.Player2:
                btn.Image = Resources.O;
                PlayerTurn = enPlayer.Player1;
                lblTurn.Text = "Player 1";
                GameStatus.PlayCount++;
                btn.Tag = "O";
                CheckWinner();
                break;
        }
    }
    else
    {
        MessageBox.Show("Wrong Choice", "Wrong", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

if(GameStatus.PlayCount == 9)
{
    GameStatus.GameOver = true;
    GameStatus.Winner = enWinner.Draw;
    EndGame();
}

}

public Form1()
{
    InitializeComponent();
}

```

```

private void button_Click(object sender, EventArgs e)
{
    ChangelImage((Button)sender);
}

private void RestButton(Button btn)
{
    btn.Image = Resources.question_mark_96;
    btn.Tag = "?";
    btn.BackColor = Color.Transparent;
}

private void RestartGame()
{

    RestButton(button1);
    RestButton(button2);
    RestButton(button3);
    RestButton(button4);
    RestButton(button5);
    RestButton(button6);
    RestButton(button7);
    RestButton(button8);
    RestButton(button9);

    PlayerTurn = enPlayer.Player1;
    lblTurn.Text = "Player 1";
    GameStatus.PlayCount = 0;
    GameStatus.GameOver = false;
    GameStatus.Winner = enWinner.GameInProgress;
    lblWinner.Text = "In Progress";

}

private void btnRestart_Click(object sender, EventArgs e)
{
    RestartGame();
}

private void Form1_Paint(object sender, PaintEventArgs e)
{

    Color white = Color.FromArgb(255, 255, 255, 255);
    Pen whitePen = new Pen(white);
    whitePen.Width = 15;
    //whitePen.DashStyle = System.Drawing.Drawing2D.DashStyle.Dash;
    whitePen.StartCap = System.Drawing.Drawing2D.LineCap.Round;
    whitePen.EndCap = System.Drawing.Drawing2D.LineCap.Round;

    //draw Horizontal lines
    e.Graphics.DrawLine(whitePen, 400, 300, 1050, 300);
    e.Graphics.DrawLine(whitePen, 400, 460, 1050, 460);

    //draw Vertical lines
    e.Graphics.DrawLine(whitePen, 610, 140, 610, 620);
    e.Graphics.DrawLine(whitePen, 840, 140, 840, 620);
}

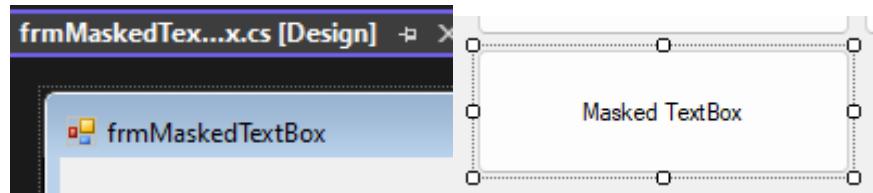
```

```
    }  
}  
}
```

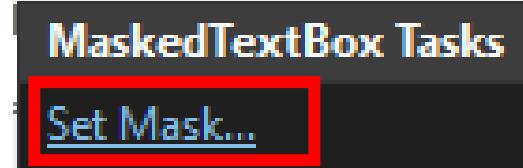
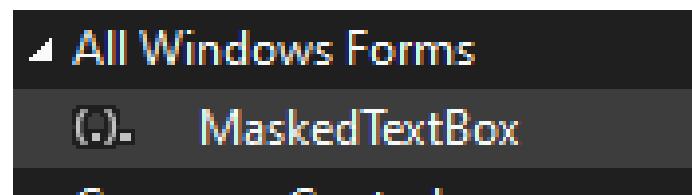
## MaskedTextBox

هوا text box عادي بس بيغير اليوزر انه يدخل الداتا ب format معينه

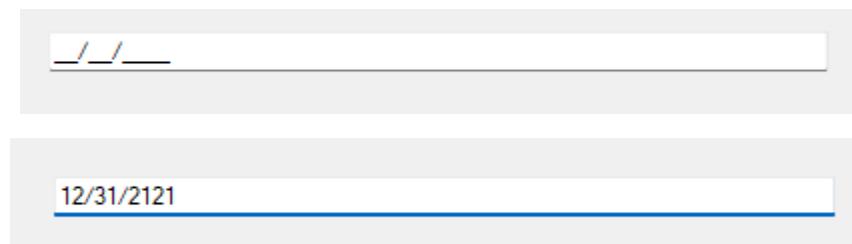
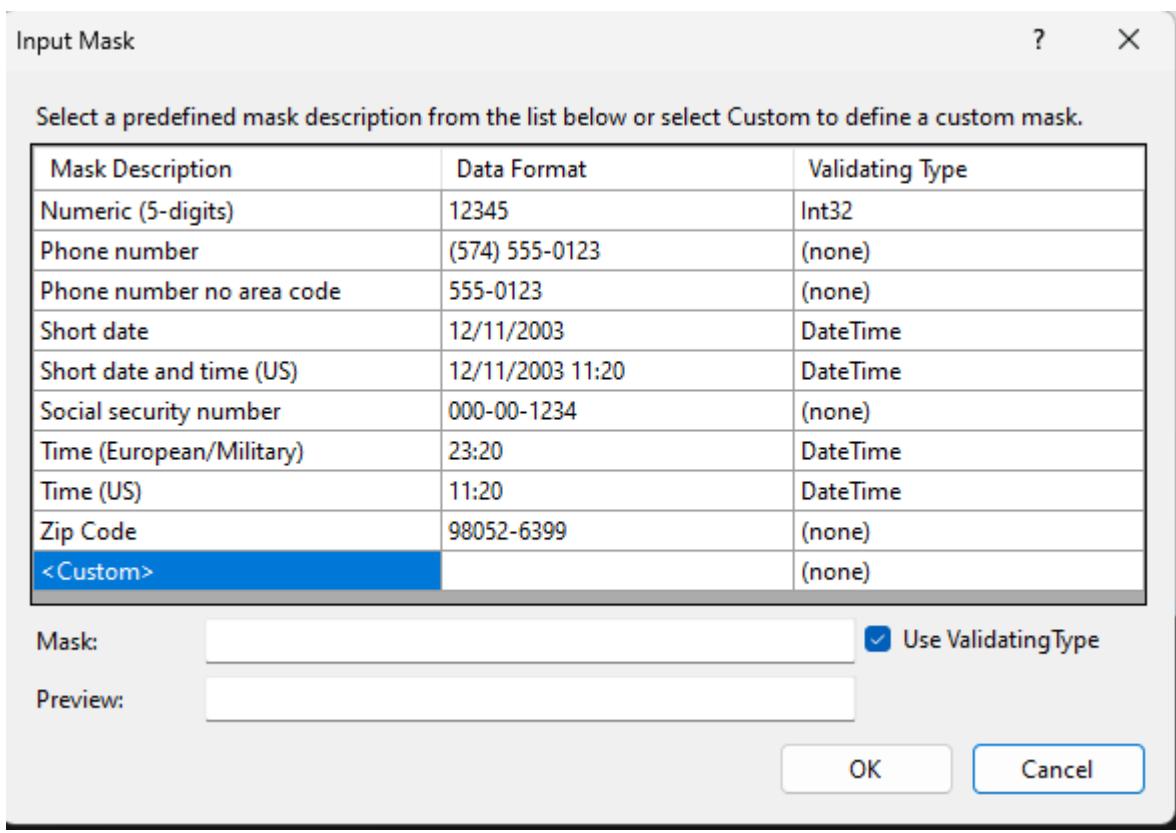
هنعمل فورم جديد



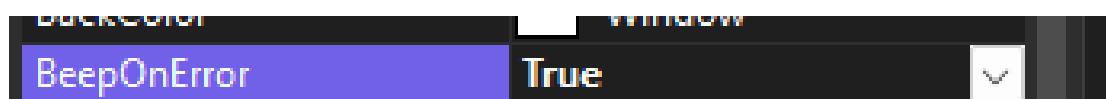
بعدين من ال toolbox هنختار masked textbox



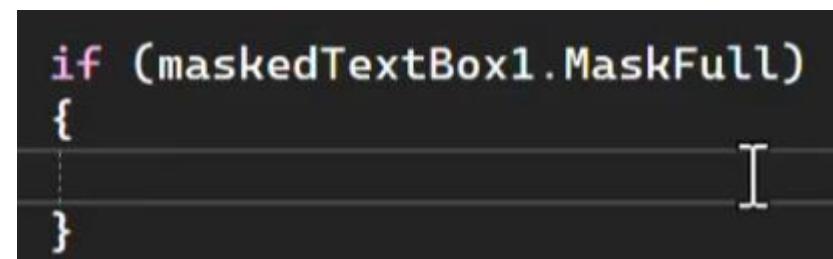
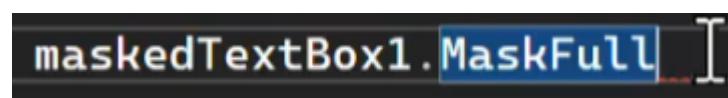
ومن هنا بتختار ال format اللي انت عايزه



لو دخلته ارقام بعدد معين مش هيقبل غير ب 5 ارقام  
 لو خليته تاريخ بصيغه معينه مش هيقبل غير بتاريخ بالصيغه دي وهكذا  
 فيه اوشن اسمه beep on error وده بيزلنك لما بتدخل حاجه غلط



فيه الكود ده بيطلعك bool بيقولك لو ال mask اتعبي ولا لا



طيب بيقولك كمان ينفع تتحكم بال mask ده براحتك عن طريق الخاصيه

Mask

00/00/0000

وهنا جاييلك الرومز اللي ممكن تحطها وكل رمز بيبين ايه

MaskedTextBox1.Mask = "00/00/0000";

Here is a list and description of masking characters.

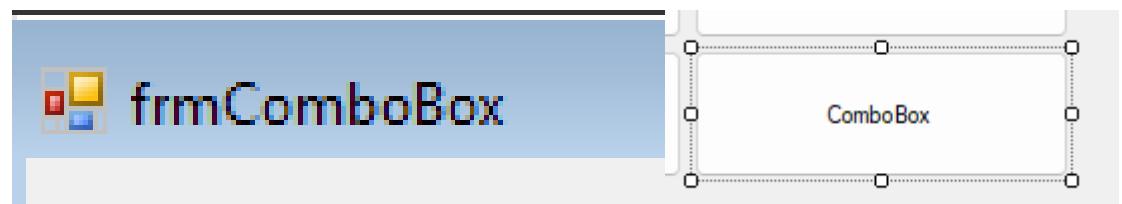
- 0 – Digit, required. Value between 0 and 9.
- 9 – Digit or space, optional.
- # - Digit or space, optional. If this position is blank in the mask, it will be rendered as a space in the Text property.
- L - Letter, required. Restricts input to the ASCII letters a-z and A-Z.
- ? - Letter, optional. Restricts input to the ASCII letters a-z and A-Z.
- & - Character, required.
- C - Character, optional. Any non-control character.
- A - Alphanumeric, required.
- a - Alphanumeric, optional.
- . - Decimal placeholder.
- , - Thousands placeholder.
- : - Time separator.
- / - Date separator.
- \$ - Currency symbol.
- < - Shift down. Converts all characters that follow to lowercase.
- > - Shift up. Converts all characters that follow to uppercase.
- | - Disable a previous shift up or shift down.
- \ - Escape. Escapes a mask character, turning it into a literal. "\\\" is the escape sequence for a backslash.

All other characters - Literals. All non-mask elements will appear as themselves within MaskedTextBox. Literals always occupy a static position in the mask at run time, and cannot be moved or deleted by the user.

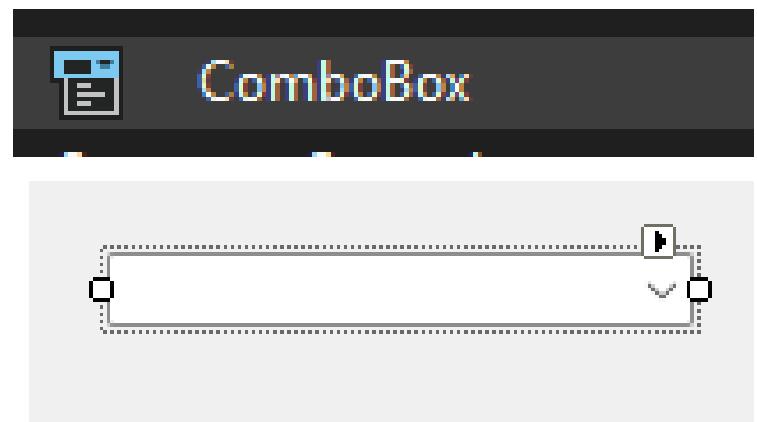
## ComboBox

ال drop down list او ال comboBox بيتتحالك انك تختار ما بين عدة خيارات وبيكون بديل عن ال radio button

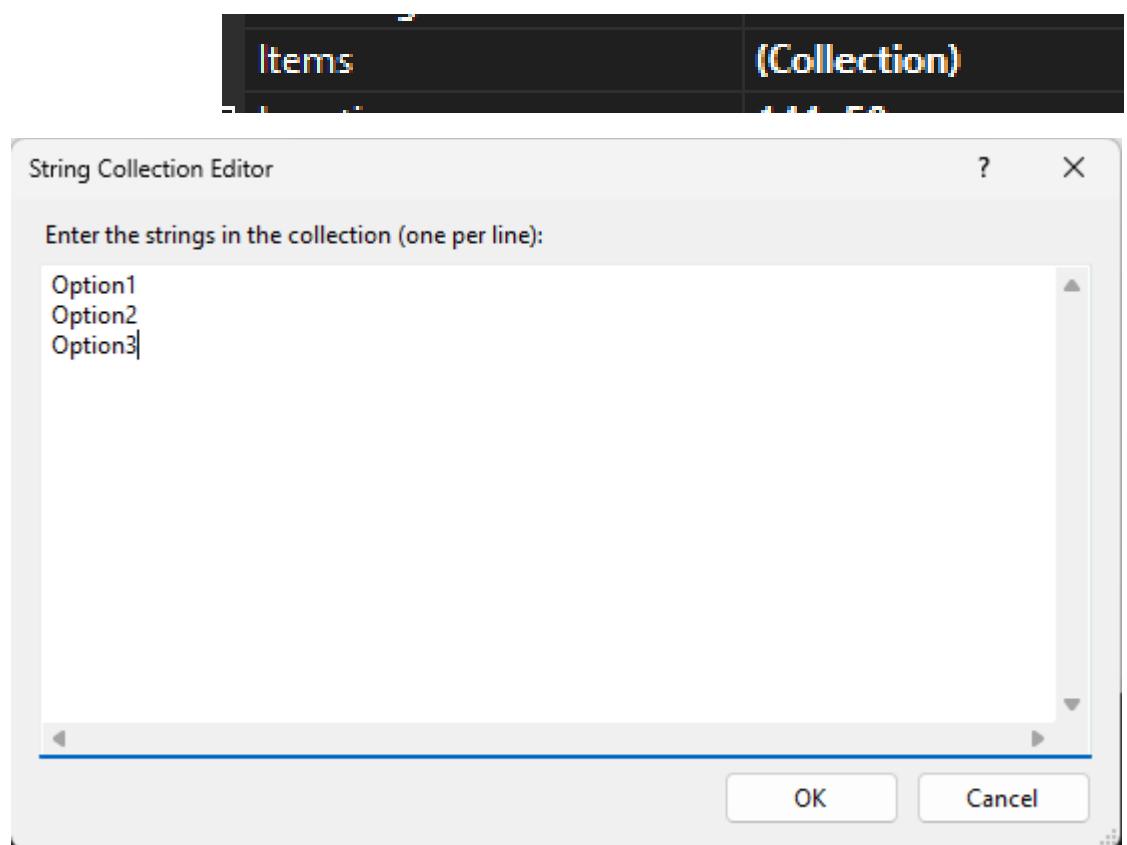
هنعمل فورم جديد

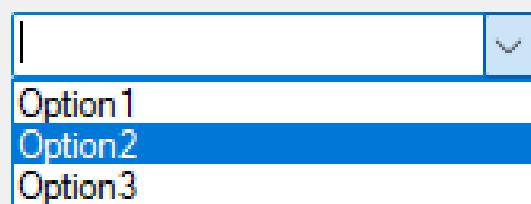
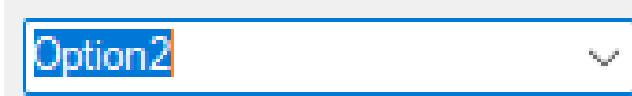


وبعدين هنضيف comboBox

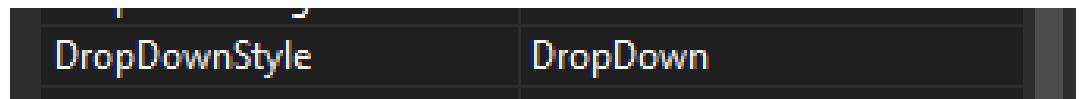


عشان اضيف فيه خيارات بروح لخاصية اسمها Items





وبتقدر تكتب فيه  
الخاصيه دي لو خيتها drop down list مش هيخلني فيه امكانيه انك تكتب اختار بس من الموجود



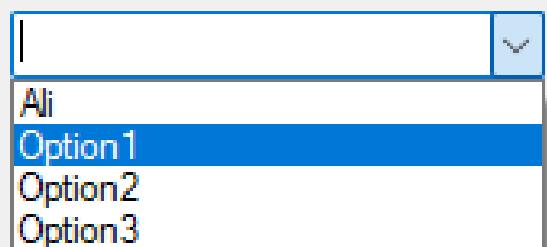
ودي لو عايز ترتبعهم



طيب دلوقتي عاوزين نضيف فيها عناصر بالكود

button1

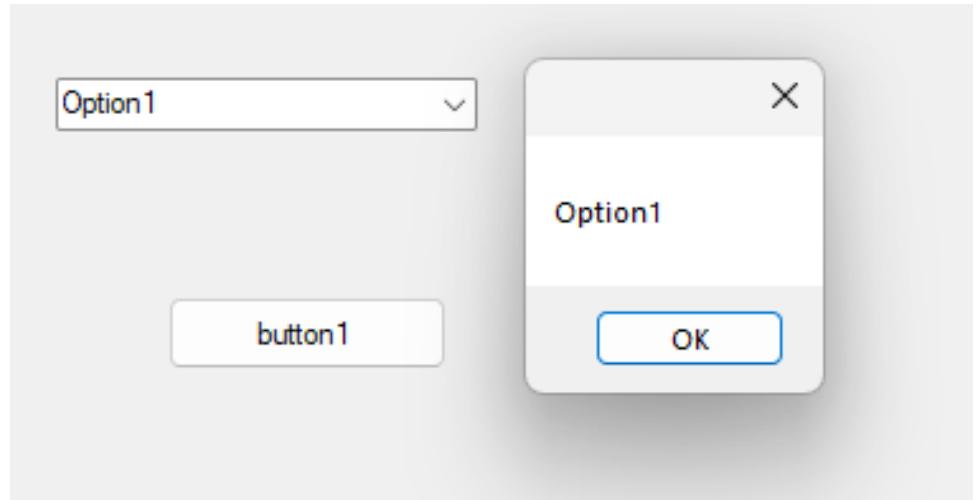
```
private void button1_Click(object sender, EventArgs e)
{
    comboBox1.Items.Add("Ali");
}
```



button1

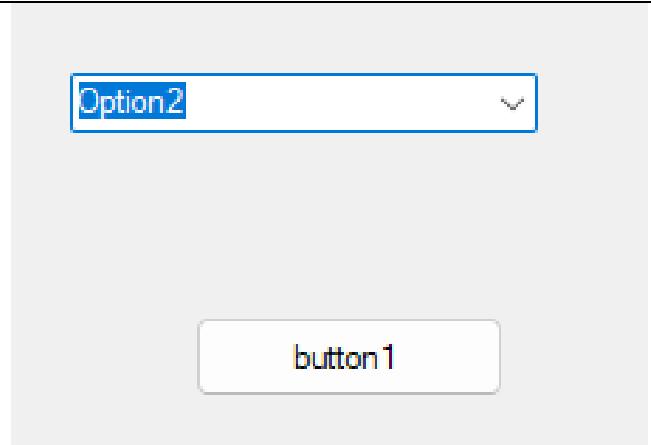
طيب انا دلوقتي عايز اعمل message box للعنصر اللي اخترته  
selected index changed event اسمه combo box Double click

```
private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    MessageBox.Show(comboBox1.Text);
}
```



طيب عايز دلوقتي اخلي ال option2 هيا المختاره  
هنغيرها في ال event load بتاع الفورم

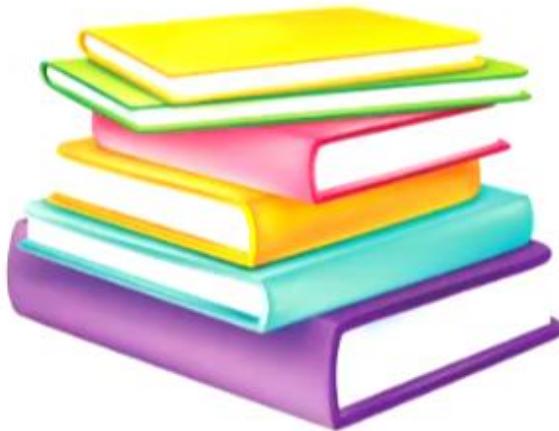
```
private void frmComboBox_Load(object sender, EventArgs e)
{
    comboBox1.SelectedIndex = 1;
}
```



### Combo Box Exercise

عاوزين نعمل التطبيق ده بس بال combo box لانه بيأخذ مساحه اقل من ال radio button

# BOOK



Change Picture:  

## **Combo Box Exercise (Solution)**

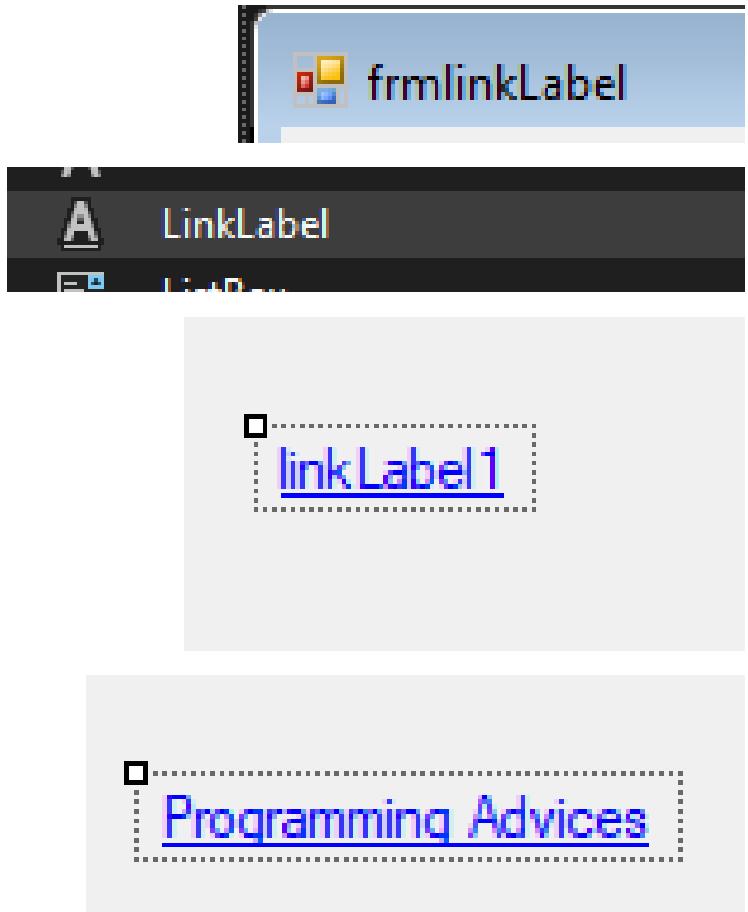
```
private void Form1_Load(object sender, EventArgs e)
{
    cmbPictures.SelectedIndex = 0;
}

private void cmbPictures_SelectedIndexChanged(object sender, EventArgs e)
{
    switch (cmbPictures.SelectedItem.ToString().ToLower())
    {
        case "boy":
            pictureBox1.Image = Resources.Boy;
            lblTitle.Text = "Boy";
            break;
        case "girl":
            pictureBox1.Image = Resources.Girl;
            lblTitle.Text = "Girl";
            break;
        case "book":
            pictureBox1.Image = Resources.Book ;
            lblTitle.Text = "Book";
            break;
        case "pen":
            pictureBox1.Image = Resources.Pen;
            lblTitle.Text = "Pen";
            break;
    }
}
```

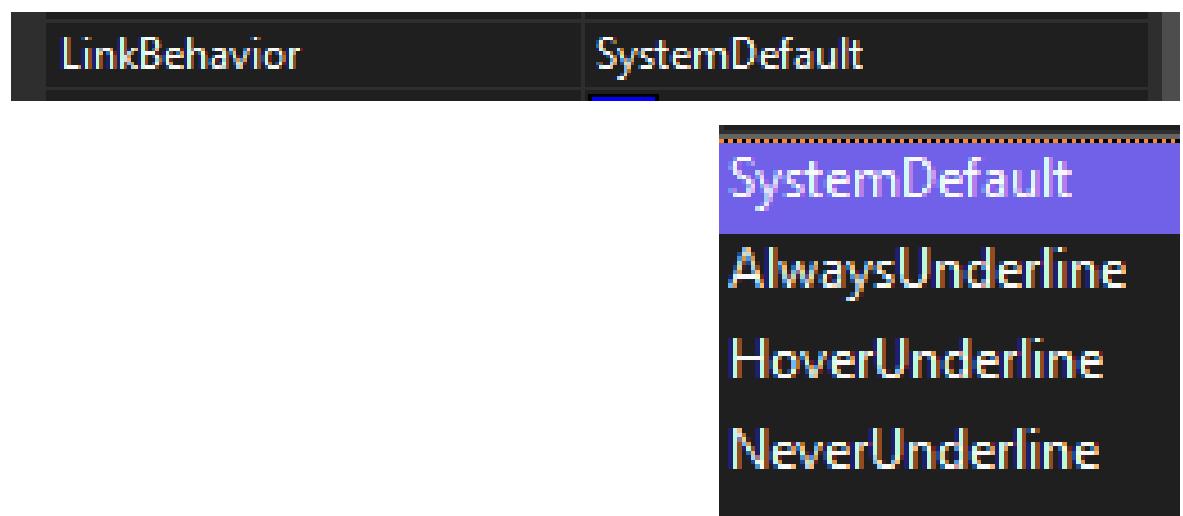
```
}
```

## LinkLabel

ال linklabel هو label عادي بس بيظهر في شكل link  
هنعمل form



الخاصية behavior بتخليلك ياما تحط خط تحتيه ياما تشيله ياما تحط الخط لما الماوس بيجي عليه



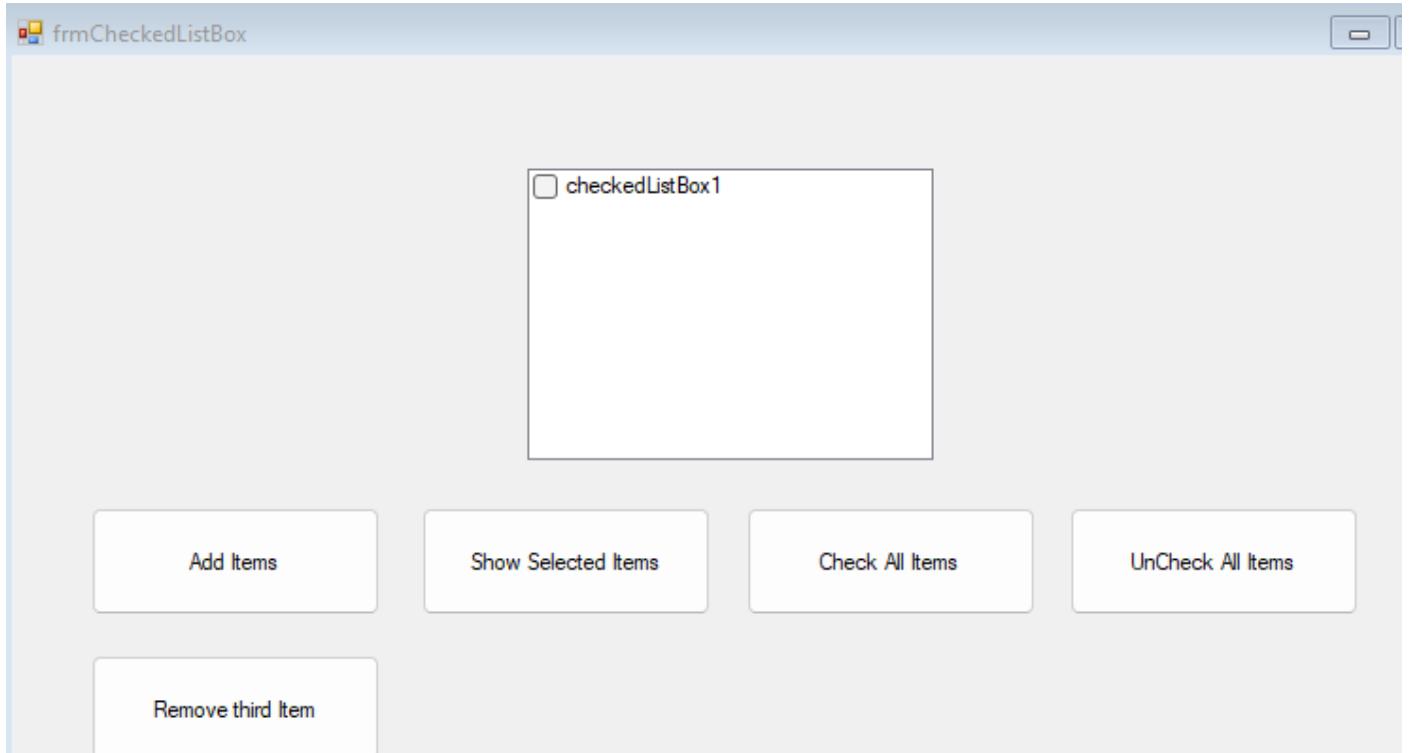
ال event بتاعه اسمه link clicked وبتحط فيها السطرين دول

```
private void.linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    linkLabel1.LinkVisited = true;
    System.Diagnostics.Process.Start("https://programmingadvices.com/");
```

```
}
```

## CheckedListBox

ال **check box** زى ال **group box** بس ده معمول لـ **check list box** وفيه بتقدر تضيف **check box** وتحذفه وتشوف مين اللي تم اختياره وتحدهم كلهم وتلغي تحديد عاللي انت عايزه يعني كأنك عامل من **array** **check boxes** هنعمل فورم جديد



ودي الاحداث بتاعت الزراير

```
private void button1_Click(object sender, EventArgs e)
{
    for (int i = 1; i <= 5; i++)
    {
        checkedListBox1.Items.Add("Item" + i);
    }
}

private void button2_Click(object sender, EventArgs e)
{
    for (int i = 0; i < checkedListBox1.CheckedItems.Count; i++)
    {
        MessageBox.Show(checkedListBox1.CheckedItems[i].ToString());
    }
}

private void button3_Click(object sender, EventArgs e)
{
    for (int i = 0; i < checkedListBox1.Items.Count; i++)
}
```

هنا بضيف 5 من ال  
check box

وهنا بي Shawf لو تم  
اختيار العنصر بيطلع  
رساله

```

{
    checkedListBox1.SetItemChecked(i,true);
}

}

private void button4_Click(object sender, EventArgs e)
{
    for (int i = 0; i < checkedListBox1.Items.Count; i++)
    {
        checkedListBox1.SetItemChecked(i, false);
    }
}

private void button5_Click(object sender, EventArgs e)
{
    checkedListBox1.Items.RemoveAt(2);

}

```

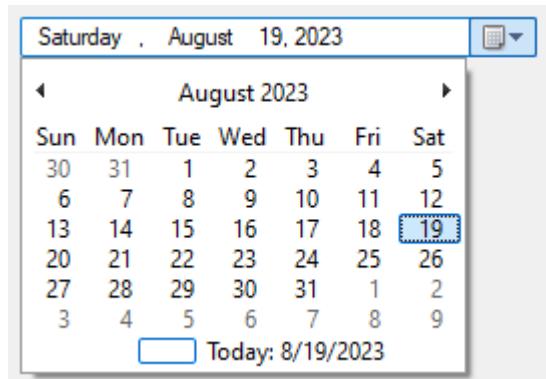
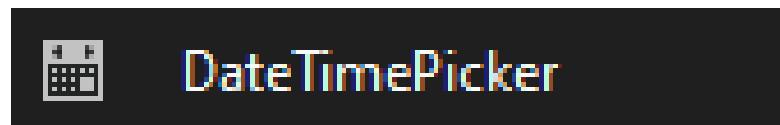
وهنا بيختار كل العناصر

وهنا بيلغي اختيار كل العناصر

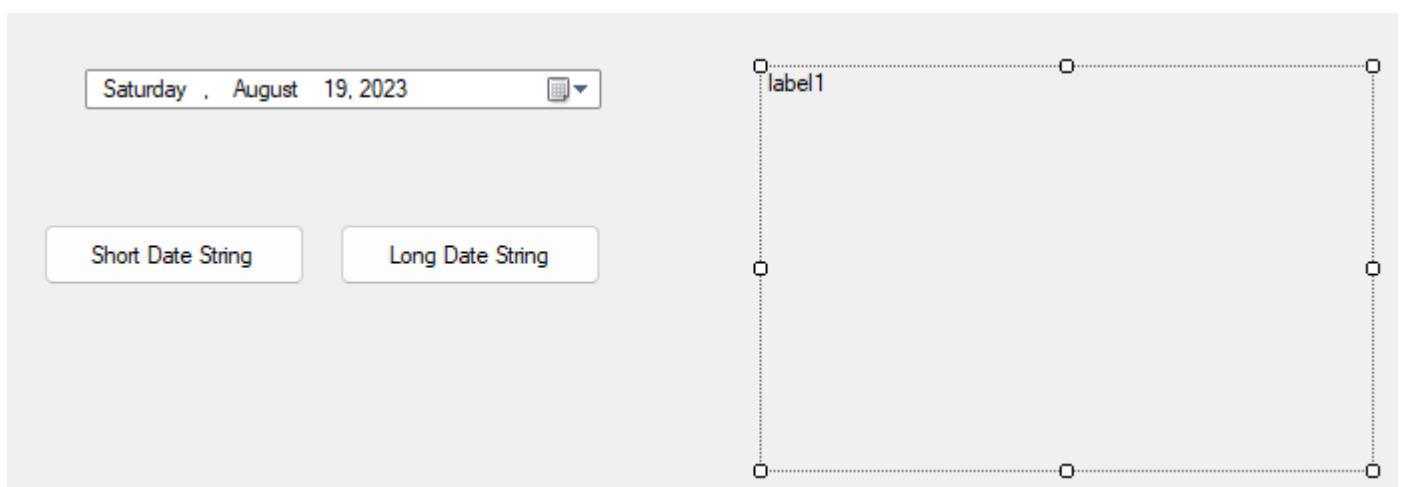
وهنا بيحذف عنصر معين

## Date Time Picker

عباره عن text box بيفتحلك مربع للتقويم بتختار منه التاريخ هنفتح فورم جديد



طيب عاوزين نعمل label يطبعنا التاريخ اللي تم اختياره باشكال مختلفه  
وعاوزين نعمل زرارين واحد يطلع رسالة بال long date وال الثاني بال short date



لو جينا في الخصائص بتاعت ال date time picker هنلاقي خاصيه اسمها format

Format	Long
GenerateMember	Long
ImeMode	Short
Location	Time
Locked	Custom
Margin	

لو اخترت long هيخليها زي ماهي ظاهره عندك كده

لو اخترت short هيظهر التاريخ بس

لو اخترت time هيظهر الوقت بس

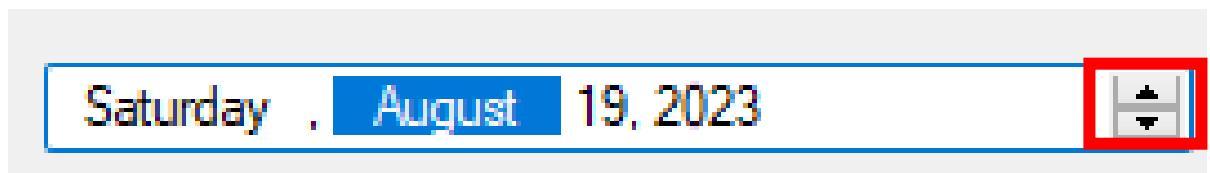
لو اخترت custom هتقدر تختار ال format اللي انت عايزه

الخاصيه check box بيظهر لك show check box ومانقدرش تختار التاريخ الا لما تعلم عليه



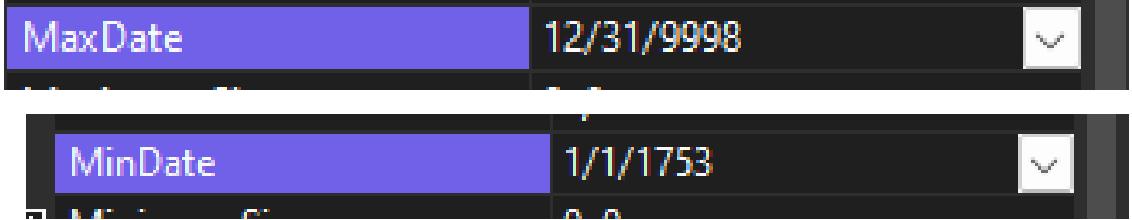
الخاصيه show up down لو عملتها true مش هيظهر شاشة التقويم لا هيخليك تختار الحاجه اللي عايز تغيرها في التاريخ وتدوس علي سهم فوق او سهم لتحت

ShowUpDown	False



فيه ال min date وال max date بتقدر من خلالهم تحدد ايه اقصي تاريخ يقدر يصله وايه اقل تاريخ يقدر يصله

كأنك مثلا عاوز تحدد انه يختار من 18 سنه او اكتر



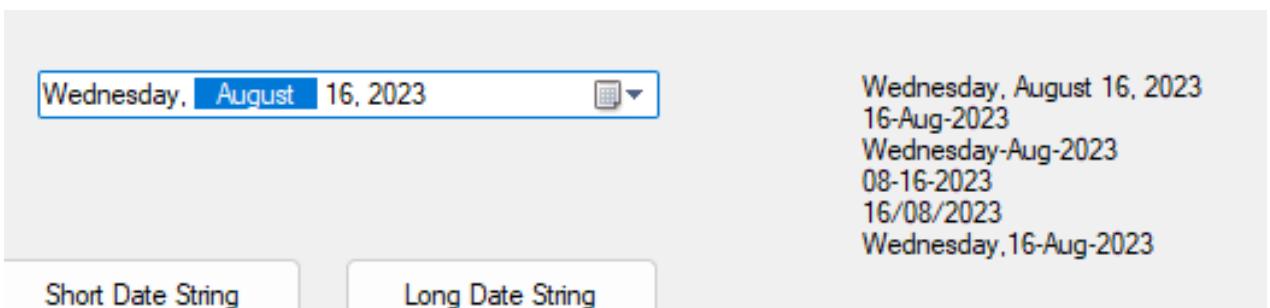
طيب تعالى بقى نشوف ازاي نخلي ال label يكتب التاريخ كل مانختار تاريخ معين  
ده ال date time picker event

```
private void dateTimePicker1_ValueChanged(object sender, EventArgs e)
{
    label1.Text = dateTimePicker1.Text + Environment.NewLine;

    label1.Text += dateTimePicker1.Value.ToString("dd-MMM-yyyy") + Environment.NewLine;
    label1.Text += dateTimePicker1.Value.ToString("dddd-MMM-yyyy") + Environment.NewLine;
    label1.Text += dateTimePicker1.Value.ToString("MM-dd-yyyy") + Environment.NewLine;
    label1.Text += dateTimePicker1.Value.ToString("dd/MM/yyyy") + Environment.NewLine;
    label1.Text += dateTimePicker1.Value.ToString("ddd,dd-MMM-yyyy") + Environment.NewLine;
}
```

الجزء ده بيقوله انزل سطر جديد بدل ماتقوله "\n"

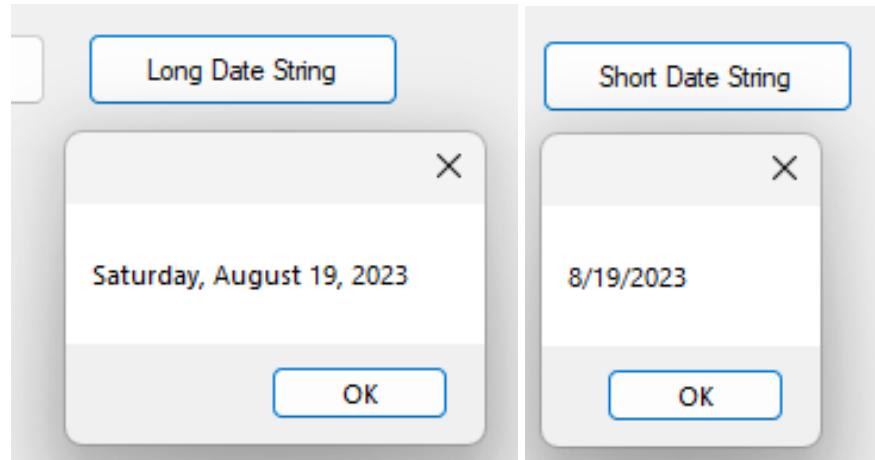
Environment.NewLine



فاضل كده الزرارين وكل اللي هنعمله اننا هنسخدم فيهم functions واحده اسمها short date وانت اقدر قلب شوف فيه تاني والثانية اسمها long date

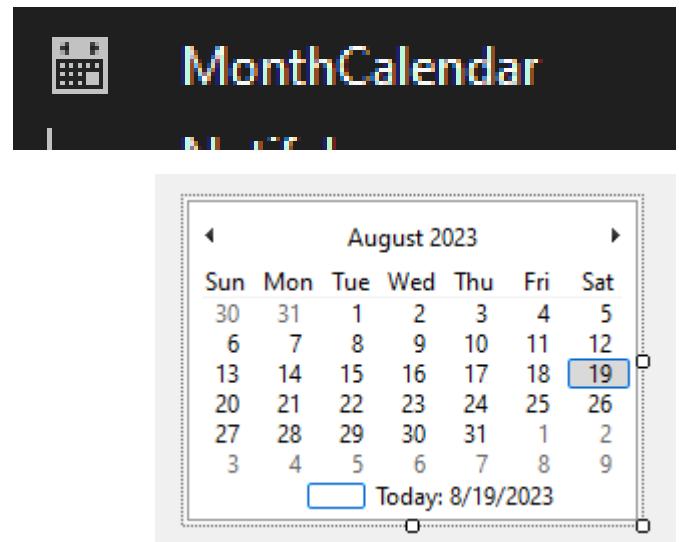
```
private void btnShortDateString_Click(object sender, EventArgs e)
{
    MessageBox.Show(dateTimePicker1.Value.ToShortDateString());
}

private void btnLongDateString_Click(object sender, EventArgs e)
{
    MessageBox.Show(dateTimePicker1.Value.ToString("yyyy-MM-dd"));
}
```

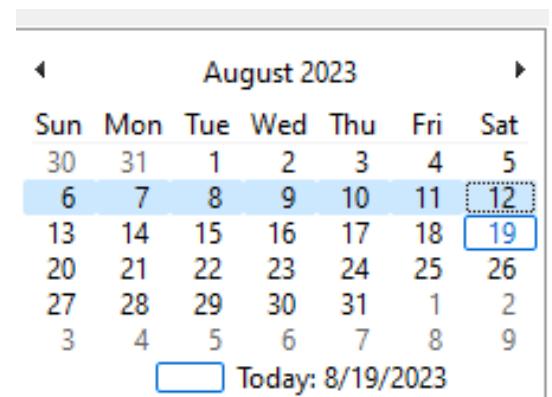


## **MonthCalandar**

دي ال month calander



وتقدر تعمل فيها نفس اللي بتعمله في ال dateTimePicker  
فيه خاصيه اسمها max selection count  
انت في ال calender دي بتقدر تختار اكتر من تاريخ  
الخاصيه دي بتخليك تختار اكتر من تاريخ



بتتفع لو بتعمل حجوزات الفنادق مثلا

ال سطر ده بيروح show today لو عملتها false

Today: 8/19/2023

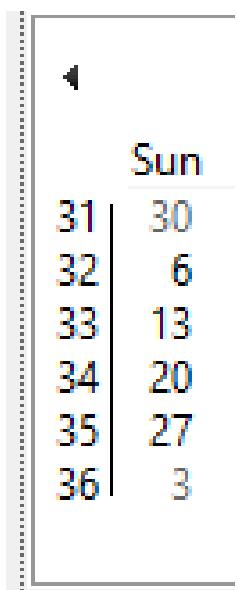
ShowToday	True
ShowTodayCircle	True

الخاصيه show today circle لو عملتها false بيشيل العلامه من علي تاريخ اليوم

ShowTodayCircle	True
ShowWeekNumbers	False

الخاصيه show week numbers بترقم الاسابيع

ShowWeekNumbers	False
Sun	31 30 32 6 33 13 34 20 35 27 36 3



عاوزين نضيف 3 ازرار واحد بيجيب اول تاريخ من التواريف اللي اخترناهم والثانى بيجيب اخر تاريخ والتالت بيجيب التواريف اللي اخترناهم كلهم

Show selected range

start date

end Date

```
private void button3_Click(object sender, EventArgs e)
{
    MessageBox.Show(monthCalendar1.SelectionRange.ToString());
}

private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show(monthCalendar1.SelectionRange.Start.ToString());
}
```

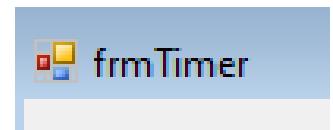
```

private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show(monthCalendar1.SelectionRange.End.ToString());
}

```

## Timer

ال timer من اسمه عبارة عن مؤقت بتدليله فتره معينه وكل مايعدى الفتره دي يعمل event اسمه tick هنعمل form جديد



وهنعمل label وزرارين



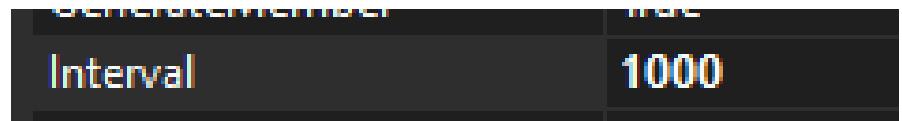
عشان تضيف timer بتعمل double click بيجيبيهولك تحت



الخاصيه enables بتشغله وتطفيه

Enabled	False
---------	-------

لوعملتها true بيروح يشوف ال interval قد ايه وكل ما بتخلص الفتره بيعملها event اسمه tick وال interval بتبقى بال milli second اللي هو ال 1000 منها بيعمل ثانية



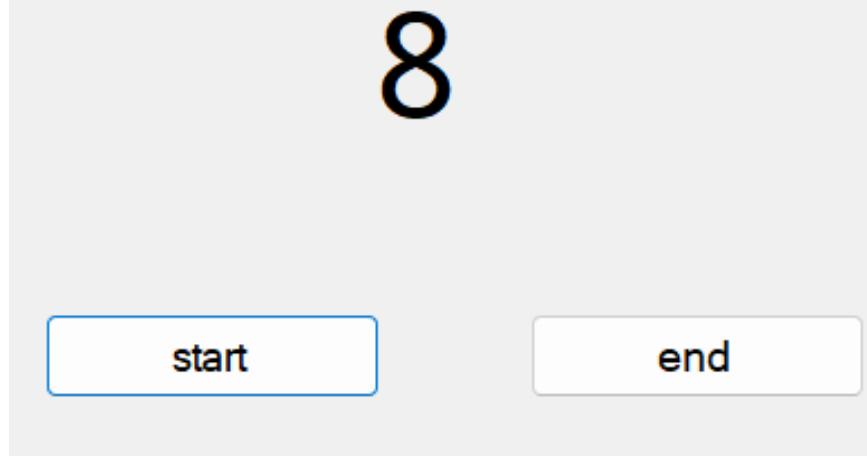
وكل اللي عملناه اننا عرفنا counter وكل مايشغل ال event يزود العداد ده ويطبعه في ال label

```
int Counter = 0;

private void button1_Click(object sender, EventArgs e)
{
    timer1.Enabled = true;
}

private void button3_Click(object sender, EventArgs e)
{
    timer1.Enabled = false;
}

private void timer1_Tick(object sender, EventArgs e)
{
    Counter++;
    label1.Text = Counter.ToString();
}
```



تقدر تعمل اكتر من timer في نفس الوقت

### NotifyIcon

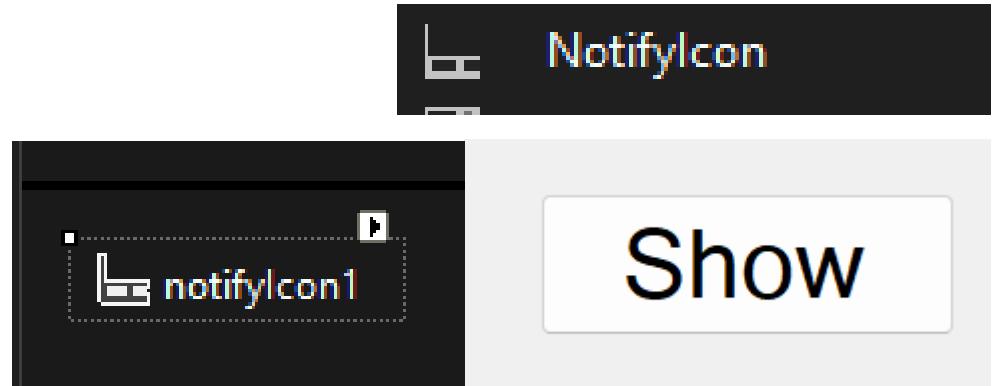
ال notify icon هو عباره عن اشعار

نهعمل فورم جديد



وهنحط فيه زرار و notify icon

ال notify icon بتظهر تحت برضه



عاوزين ندوس عالزرار يطلعنا الاشعار

تقدير تتحكم بمدة ظهورها ال C# مش بيسمحلك انك تخليها ظاهر اقل من 10 ثواني او اكتر من 30 ثانية

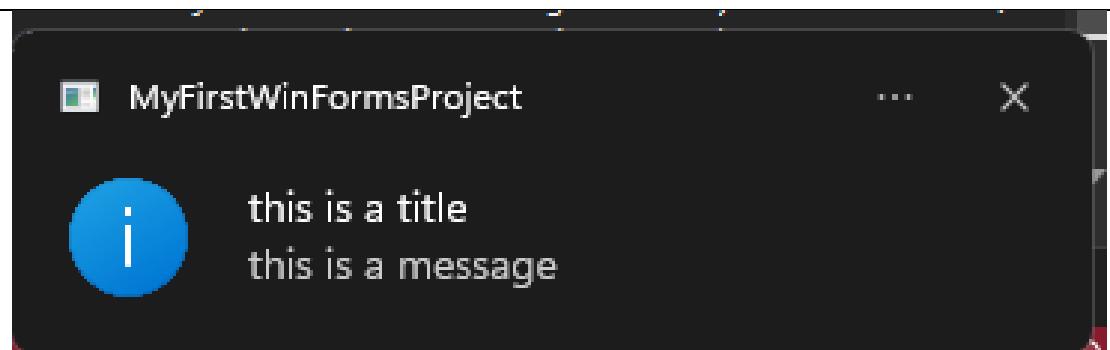
تقدير تستخدema لو عايز تطلع انه فيه ايميل جالك مثلا

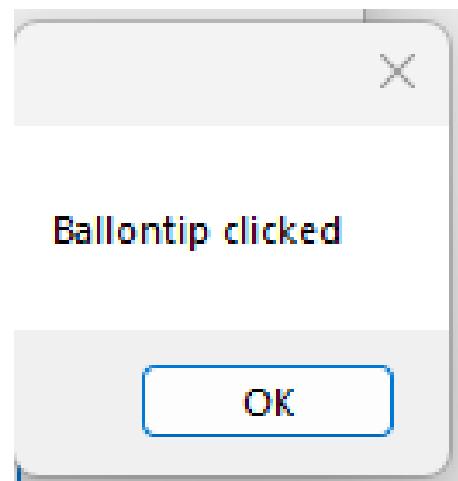
وفيها حدث on click تقدير تعمل فيه اللي انت عايزه

```
notifyIcon1.Icon = SystemIcons.Application;
notifyIcon1.BalloonTipIcon = ToolTipIcon.*;
notifyIcon1.BalloonTipTitle = "This is a title";
notifyIcon1.BalloonTipText = "This is a message";
notifyIcon1.ShowBalloonTip(1000);
```

```
private void button1_Click(object sender, EventArgs e)
{
    notifyIcon1.Icon = SystemIcons.Application;
    notifyIcon1.BalloonTipIcon = ToolTipIcon.Info;
    notifyIcon1.BalloonTipTitle = "this is a title";
    notifyIcon1.BalloonTipText = "this is a message";
    notifyIcon1.ShowBalloonTip(1000);
}

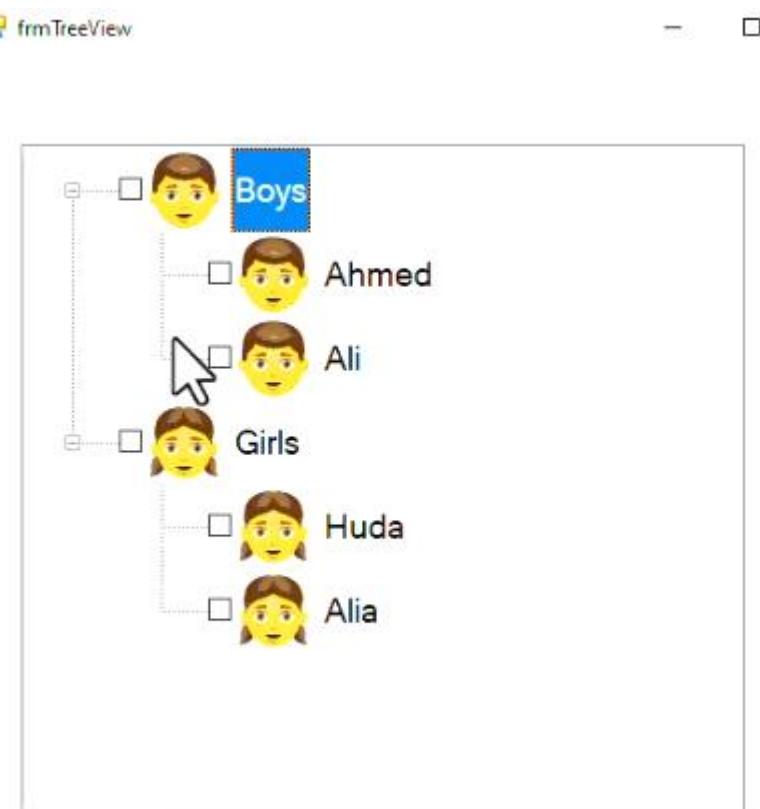
private void notifyIcon1_BalloonTipClicked(object sender, EventArgs e)
{
    MessageBox.Show("Ballontip clicked");
}
```





### **TreeView and ImageList**

ال image list هو عباره عن list بتقدر تحط فيها صور وتعين الحجم بتاعها وتقدر تربطها مع أي list  
ال tree view عباره عن شجره  
او زي لما تفتح ملفات كده  
كل عنصر من ال tree هو عباره عن node  
زي كده



وتقدر تختار أي عنصر فيها ولما تعمل double click على واحد فيهم بينفذ حدث معين  
تعالي نعمل زيه

هعمل فورم جديد

frmTreeViewAndImageList

وبعدن هنضيف image list



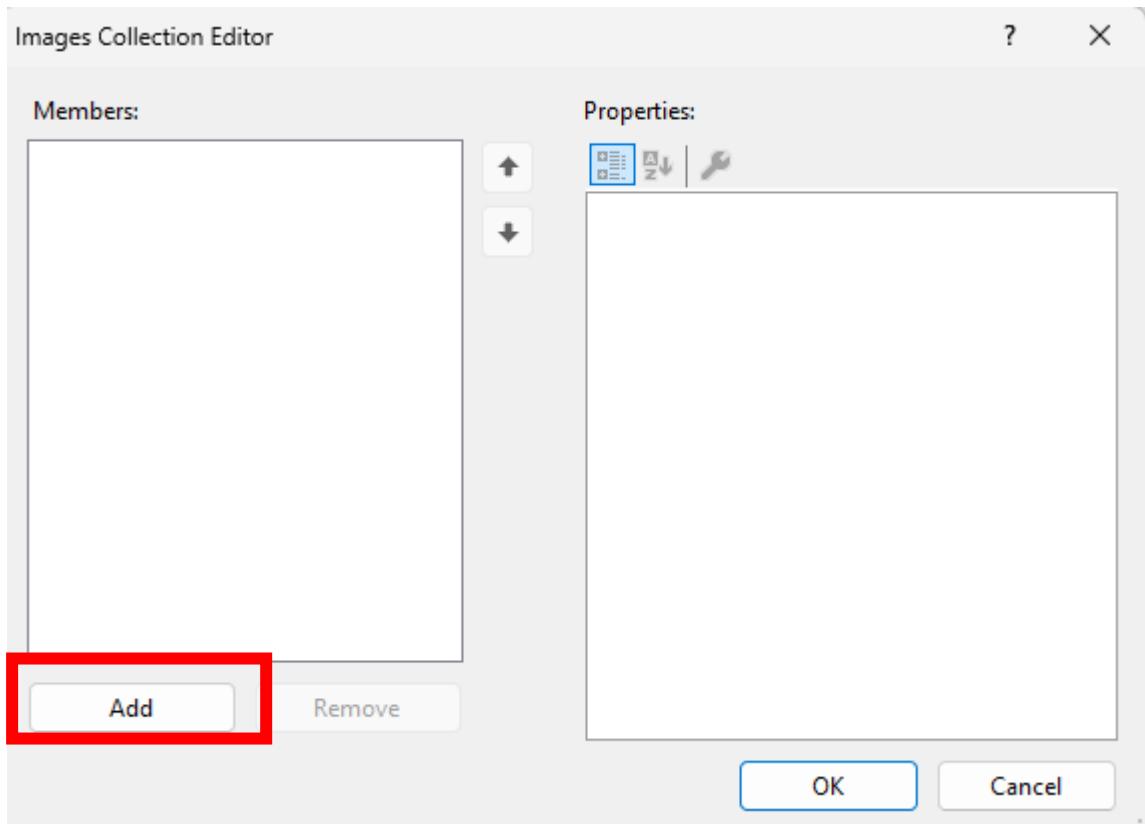
ImageList

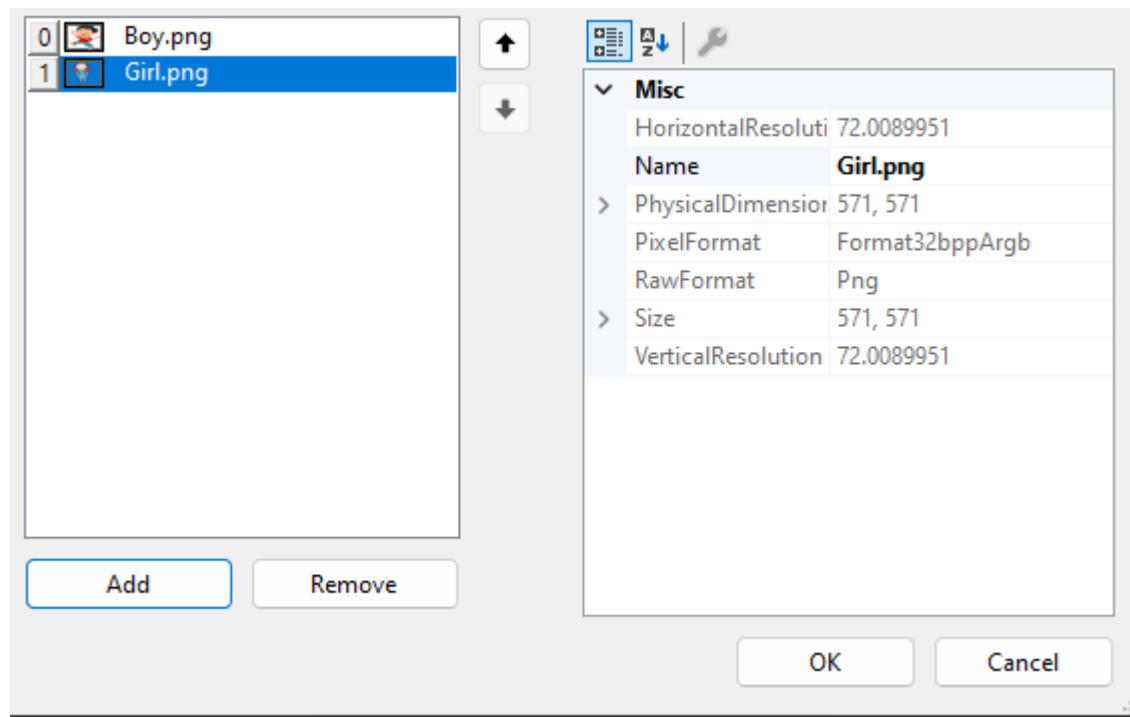
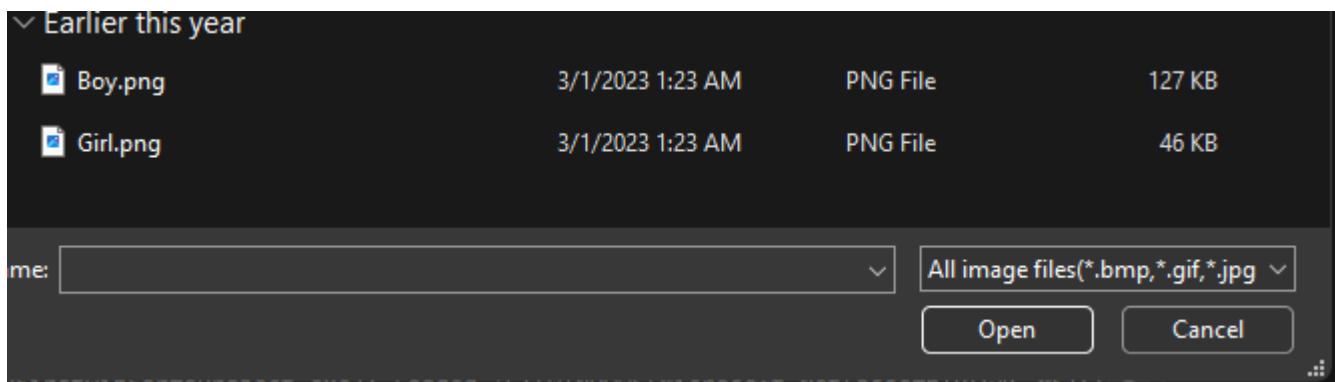


imageList1

بط فيها الصور من هنا

GenerateMember	True
Images	(Collection)





نقدر تتحكم في حجم الصور من هنا

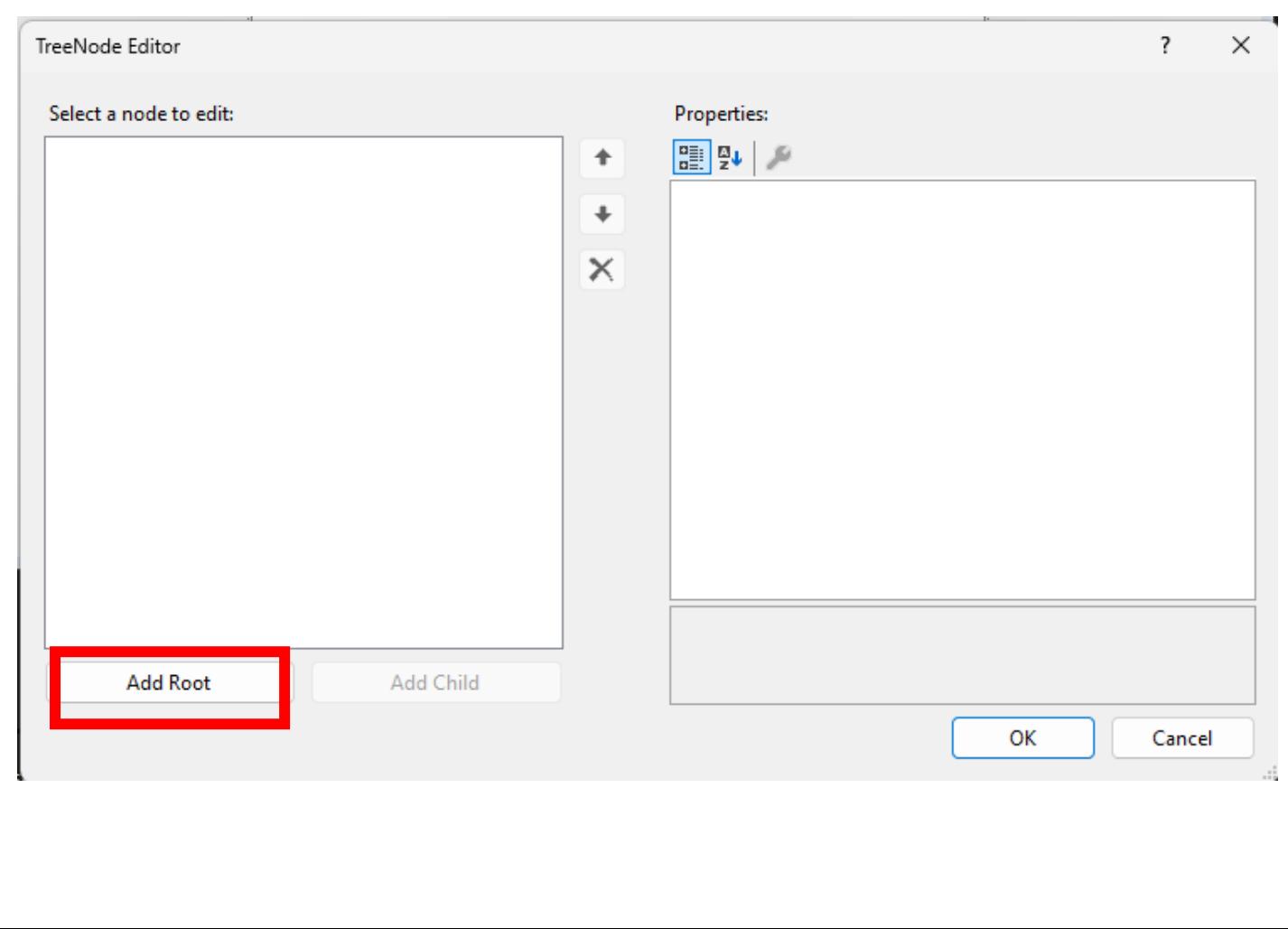
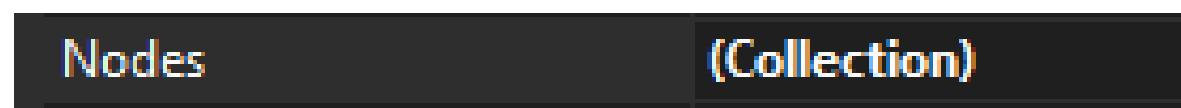
ImageSize 48, 48

تعالي بقى نحط ال tree view

TreeView



العناصر اللي فيه بتقى اسمها node  
بتقدر تضيفه من هنا



## TreeNode Editor

? X

Select a node to edit:

Node0
Node1

Add Root      Add Child

Node1 properties:

<input type="button" value="Up"/>	<input type="button" value="Down"/>	<input type="button" value="X"/>
<b>Appearance</b>		
BackColor	<input type="color"/>	
ForeColor	<input type="color"/>	
Name	Node1	
NodeFont	(none)	
Text	Node1	
ToolTipText		
<b>Behavior</b>		
Checked	False	
ContextMenu	(none)	
ContextMenuStrip	(none)	
ImageIndex	<input type="checkbox"/> (default)	
ImageKey	<input type="checkbox"/> (default)	

Text

The text displayed in the label of the tree node.

OK

Cancel

Node0

Node2

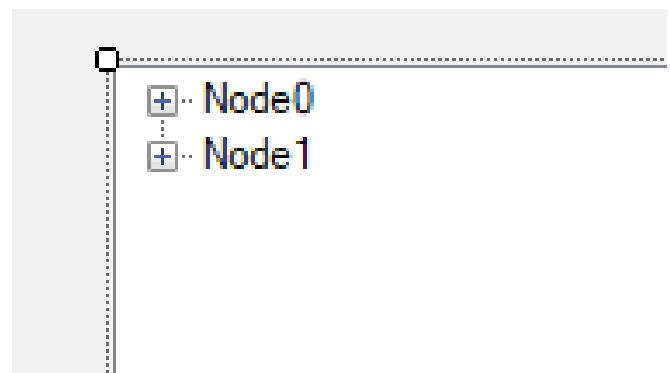
Node3

Node1

Node4

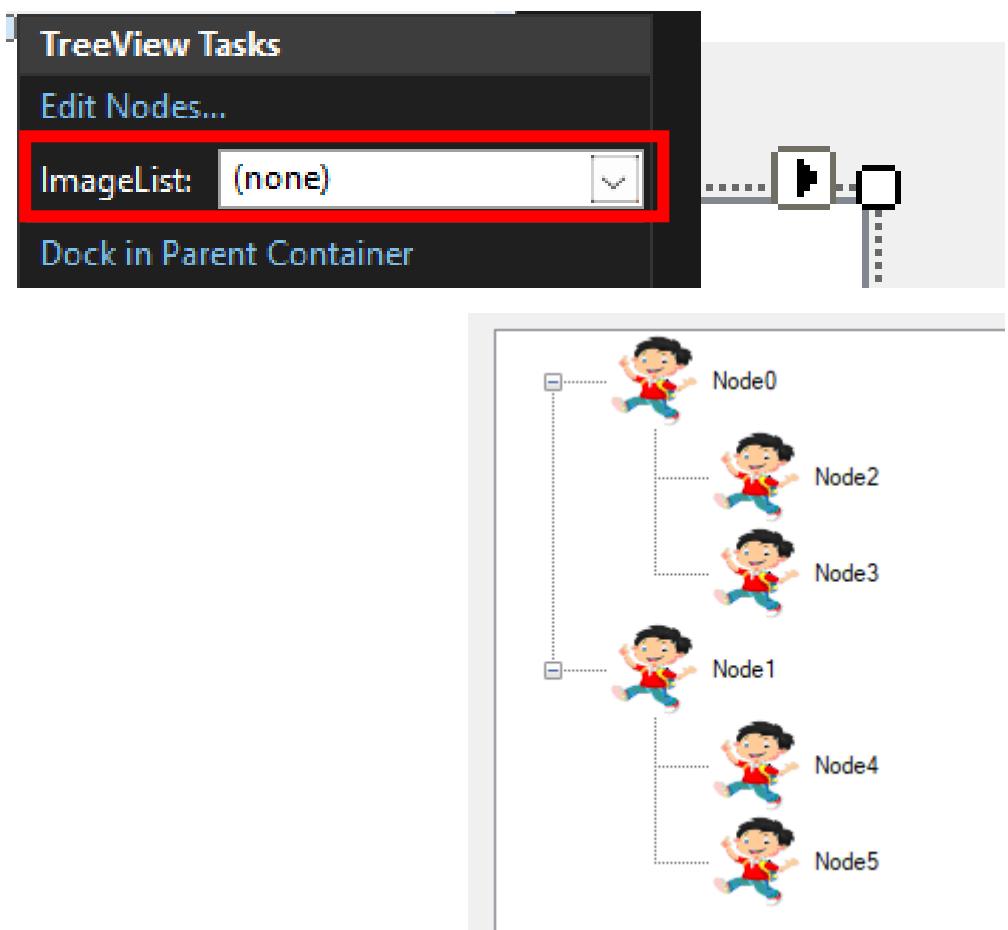
Node5

وبتقدر تضيفهم عن طريق الكود

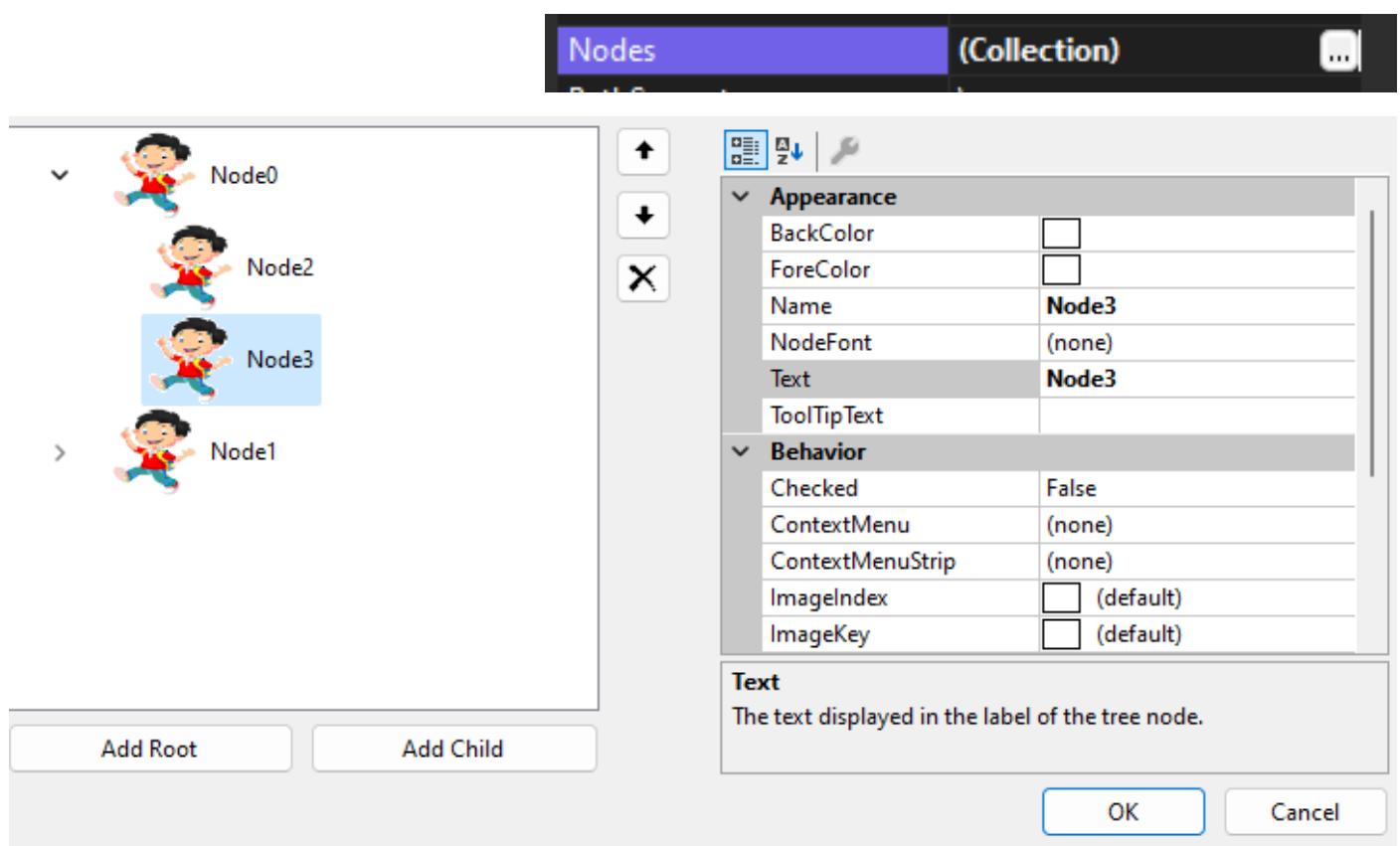


طيب دلوقتي عاوزين نضيف الصور

تقدر تضييفهم عن طريق السهم ده



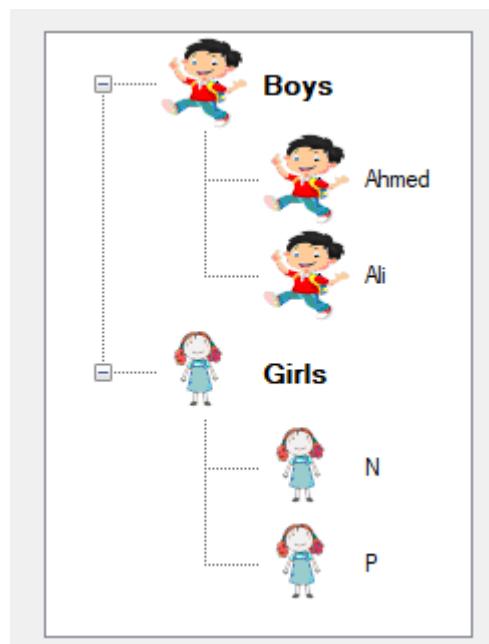
دلوقي عاوزين نغير الصور



SelectedImageIndex	<input type="button"/> (default)
SelectedImageKey	0
StateImageIndex	1
StateImageKey	<input checked="" type="button"/> (default)
▼ Data	<input type="button"/> (none)

▼ Behavior	
Checked	False
ContextMenu	(none)
ContextMenuStrip	(none)
ImageIndex	1
ImageKey	<input type="button"/> (default)
Index	1
SelectedImageIndex	1
SelectedImageKey	<input type="button"/> (default)
StateImageIndex	<input type="button"/> (none)
StateImageKey	<input type="button"/> (none)
▼ Data	

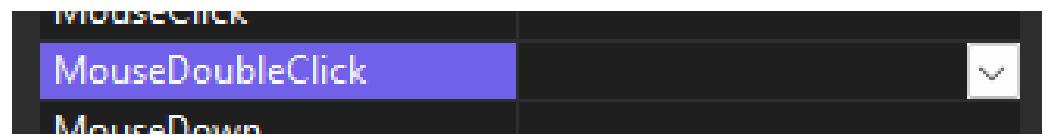
NodeFont	<input type="button"/> (none)
Text	<b>Boys</b>
ToolTipText	
NodeFont	<input type="button"/> (none) <input type="button"/>



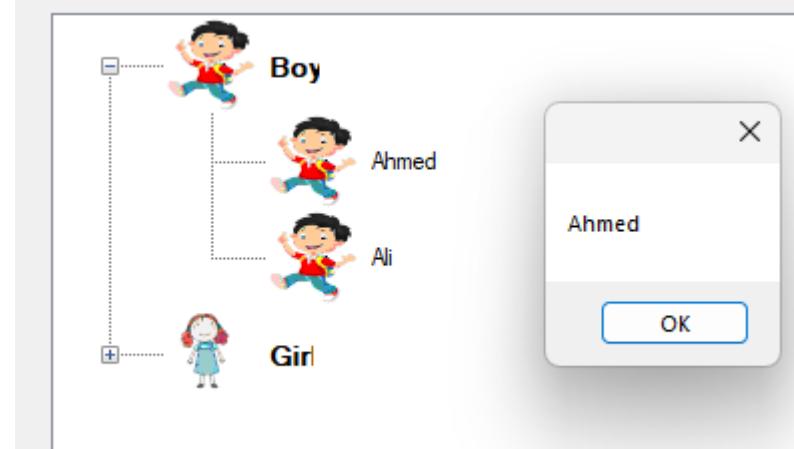
عسان تظهر ال check box

Checkboxes	True
------------	------

عاوزين لما نعمل double click يظهر اسم ال node



```
private void treeView1_MouseDoubleClick(object sender, MouseEventArgs e)
{
    MessageBox.Show(treeView1.SelectedNode.Text);
}
```



عاوزين لما نختار node رئيسيه يعمل check على كل ال nodes الفرعية  
اصل حاجه نستخدمها هنا هيا ال recursive function



```
private void treeView1_AfterCheck(object sender, TreeViewEventArgs e)
{
    CheckTreeViewNode(e.Node,e.Node.Checked);
}

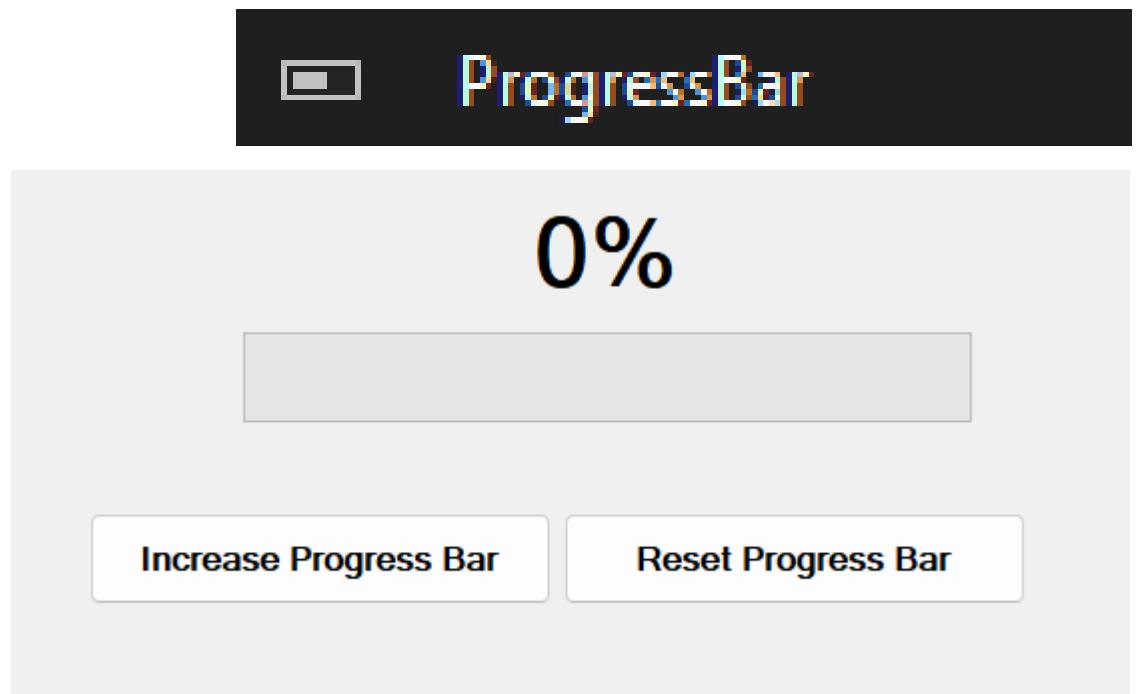
void CheckTreeViewNode(TreeNode Node,Boolean IsChecked) {

foreach (TreeNode item in Node.Nodes) {
    item.Checked = IsChecked;
    if (item.Nodes.Count>0) {
        this.CheckTreeViewNode(item,IsChecked);
    }
}
}
```

في ال for each لو عملتها من غير if هتشتغل عادي حتى لو حطيت أولاد للأولاد مش عارف حطها  
ليه

## ProgressBar

ال progress bar ده زي اللي بيظهر لك لما بتجي تحمل ملفات او حاجه  
بيكون ليه خصائص زي ال min value وال max value اللي هوا بيبدأ منين وينتهي فين  
تعالي نعمل فورم ونحط زرارين ونشوف عاوزين نعمل ايه



ودول اعلى واقل قيمه لـ progress bar  
الخاصيه step بتقولك عايزه يزيد بكم في كل مره  
وال value هيا قيمته اول مايبدأ

MarqueeAnimationSpeed	100
Maximum	100
Minimum	0
Step	10
Style	Blocks
TabIndex	0
Value	0

دلوقي عازين اول زرار يشغل ال progress bar

```
private void button1_Click(object sender, EventArgs e)
{
    progressBar1.Maximum = 100;
    progressBar1.Minimum = 0;
    progressBar1.Value = 0;

    for (byte i=1;i<=10;i++)
    {
        if (progressBar1.Value < progressBar1.Maximum)
        {
            Thread.Sleep(500);

            progressBar1.Value += 10;

            label1.Text = (byte)progressBar1.Value + "%";
            progressBar1.Refresh();
            label1.Refresh();
        }
    }
}
```

هنا بنحدد القيم وممكن تعاملها في الديزاين عادي

وهنا بيلف 10 مرات  
وبيقوله انه لو البار قيمته اقل من 100 هتعمل  
الاتي

هتنيم ال thread لمدة نصف ثانية (احنا شرحة  
يعني ايه thread )

زود القيمه بناتع البار ب 10  
غير ال text اللي في ال label

اعمل label refresh للبار ولل label

```

        }
        else { button1.Enabled = false; }

    }
}

```

عاوزين الزرار الثاني يفضي البار

```

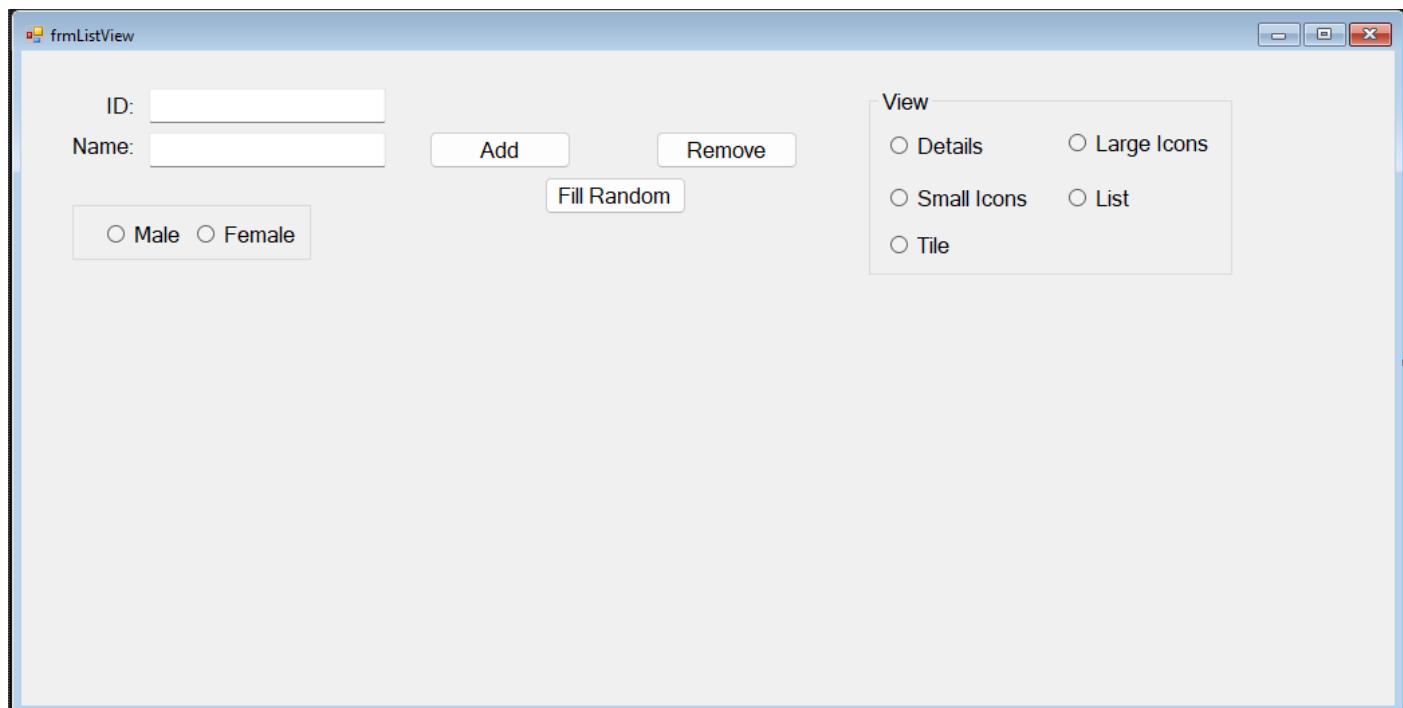
private void button3_Click(object sender, EventArgs e)
{
    progressBar1.Value = 0;
    label1.Text = "0%";
}

```

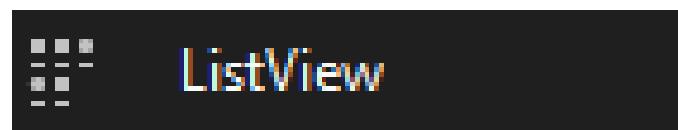
## ListView

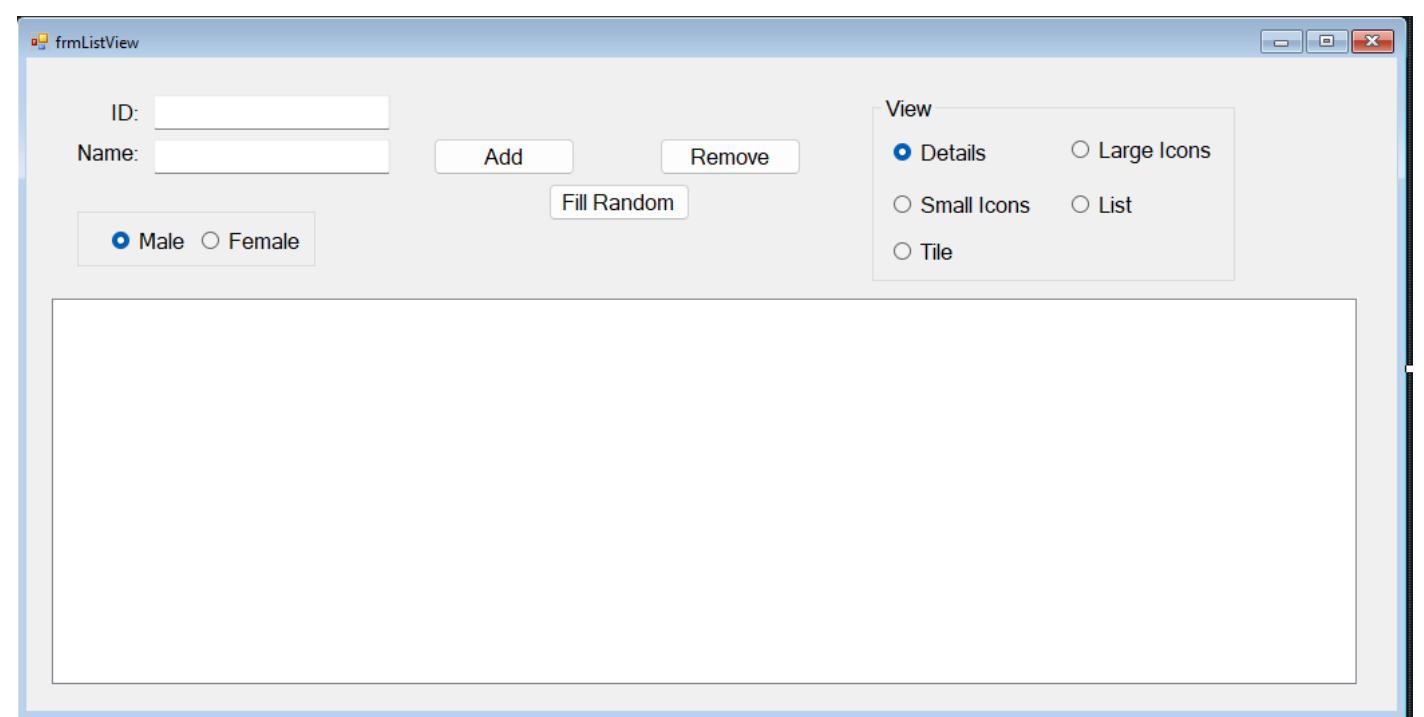
مثال عال LIST VIEW هوا الملفات اللي عندك وبيكون ليها اكثرا من شكل وبيتم تمثيلها ب واحد

هنعمل زي ده

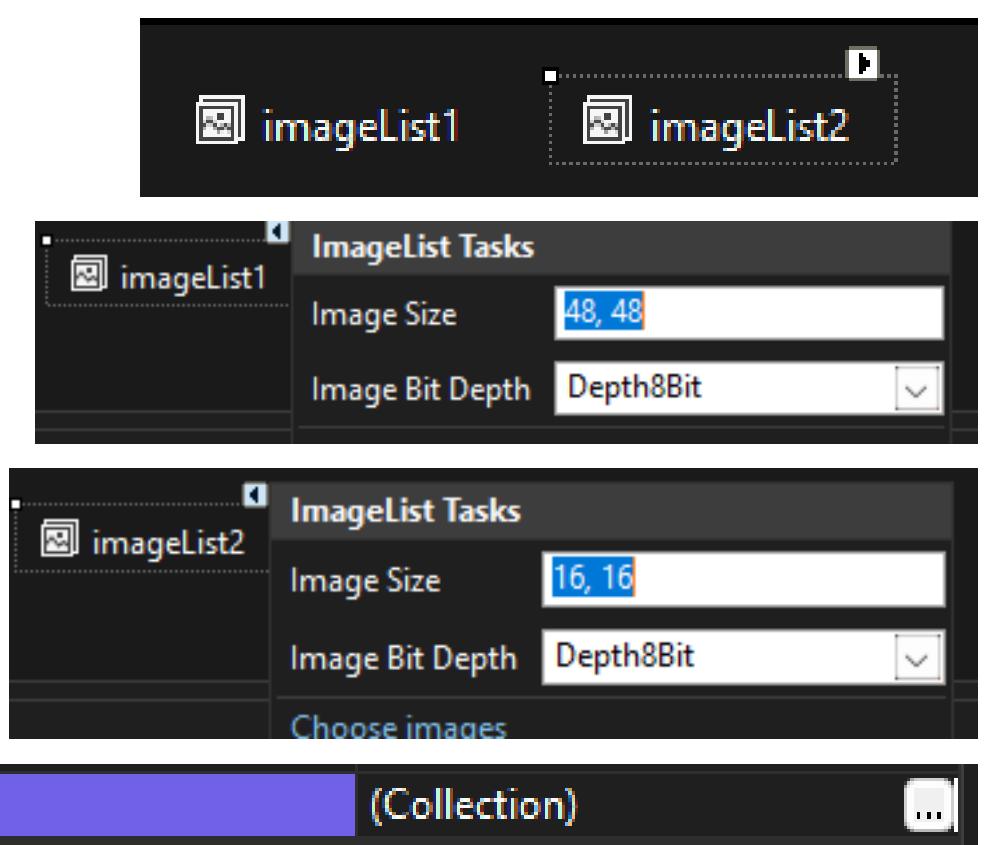


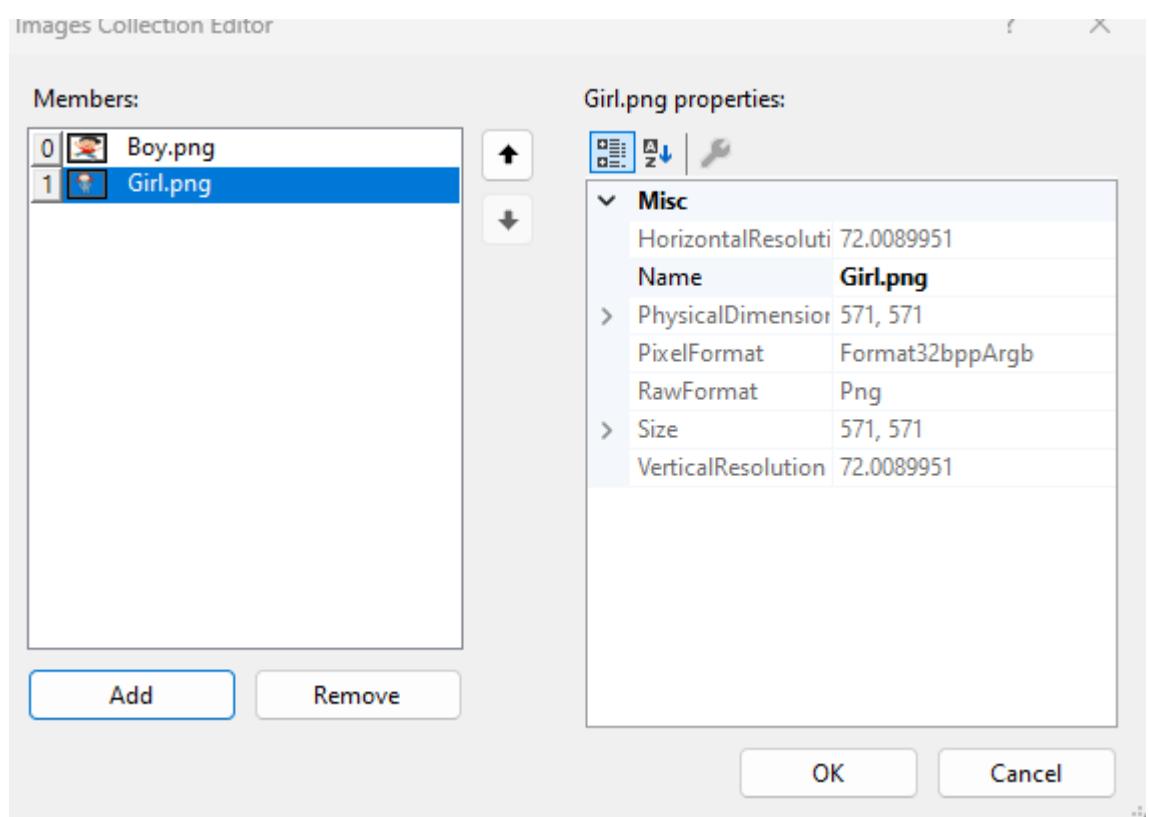
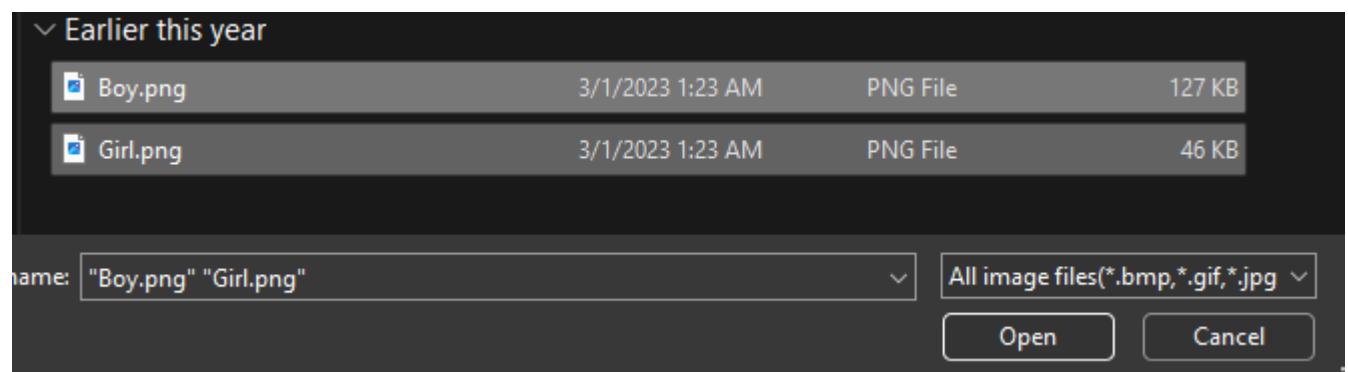
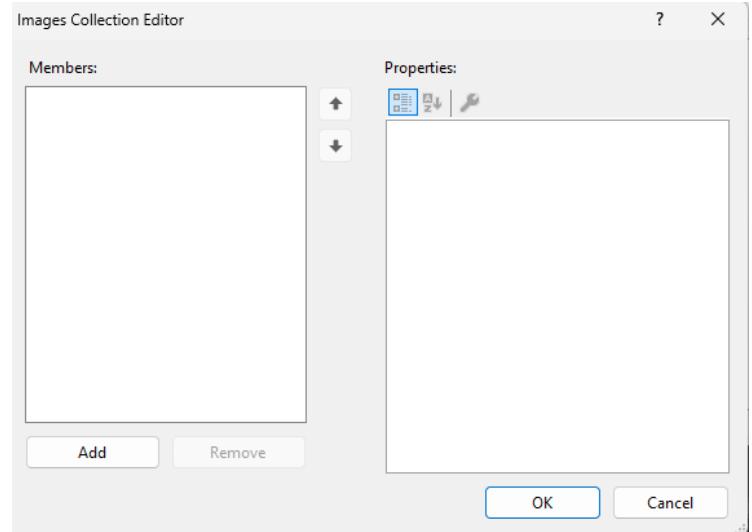
بعدين نضيف list view من هنا



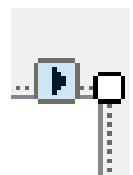


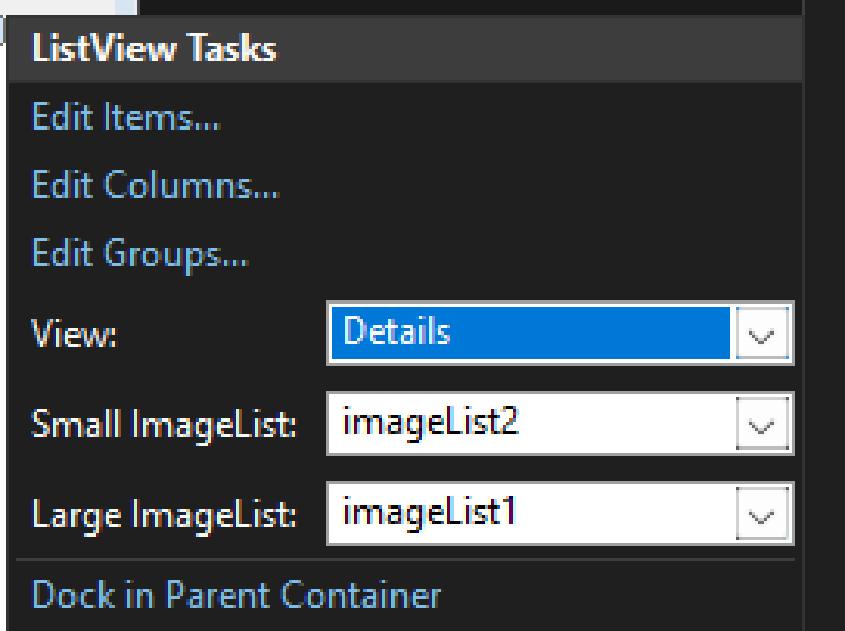
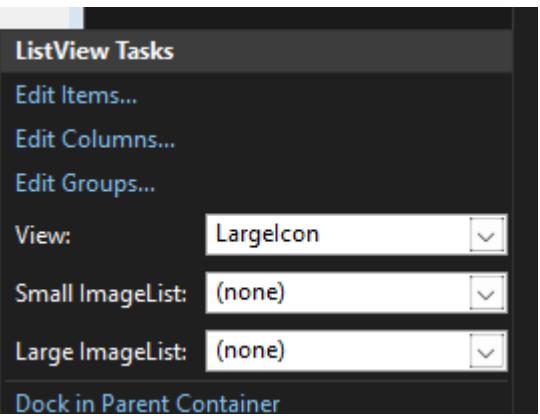
بعدين هنعمل اتنين الاتنين فيهم نفس الصور بس مختلفين في الحجم



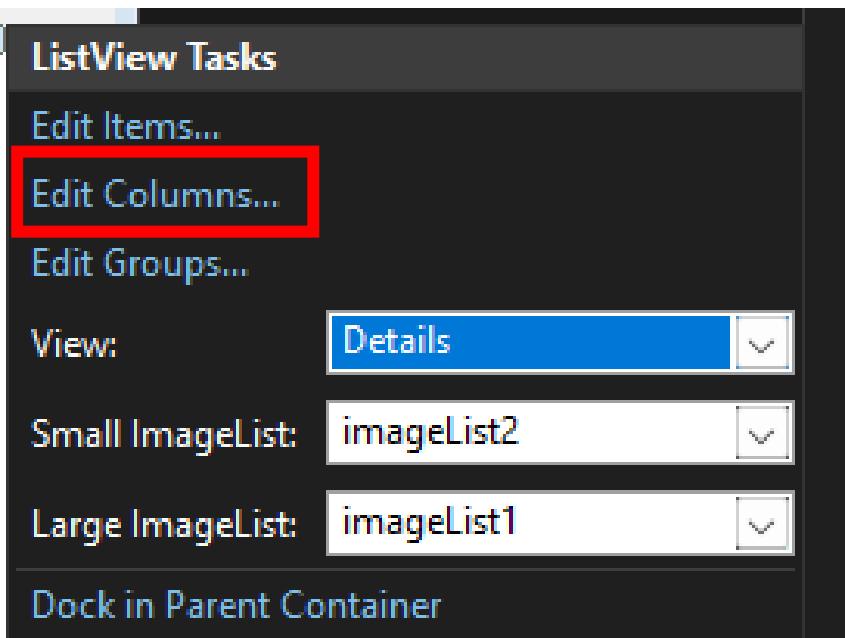


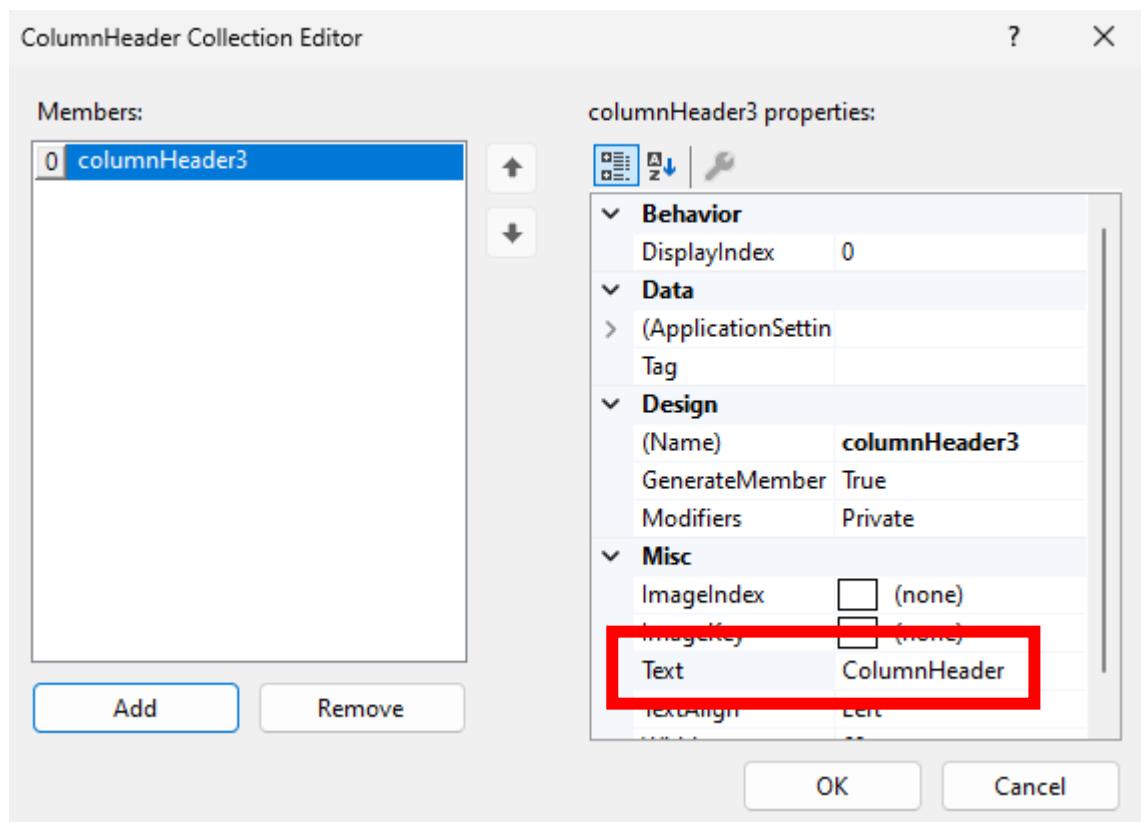
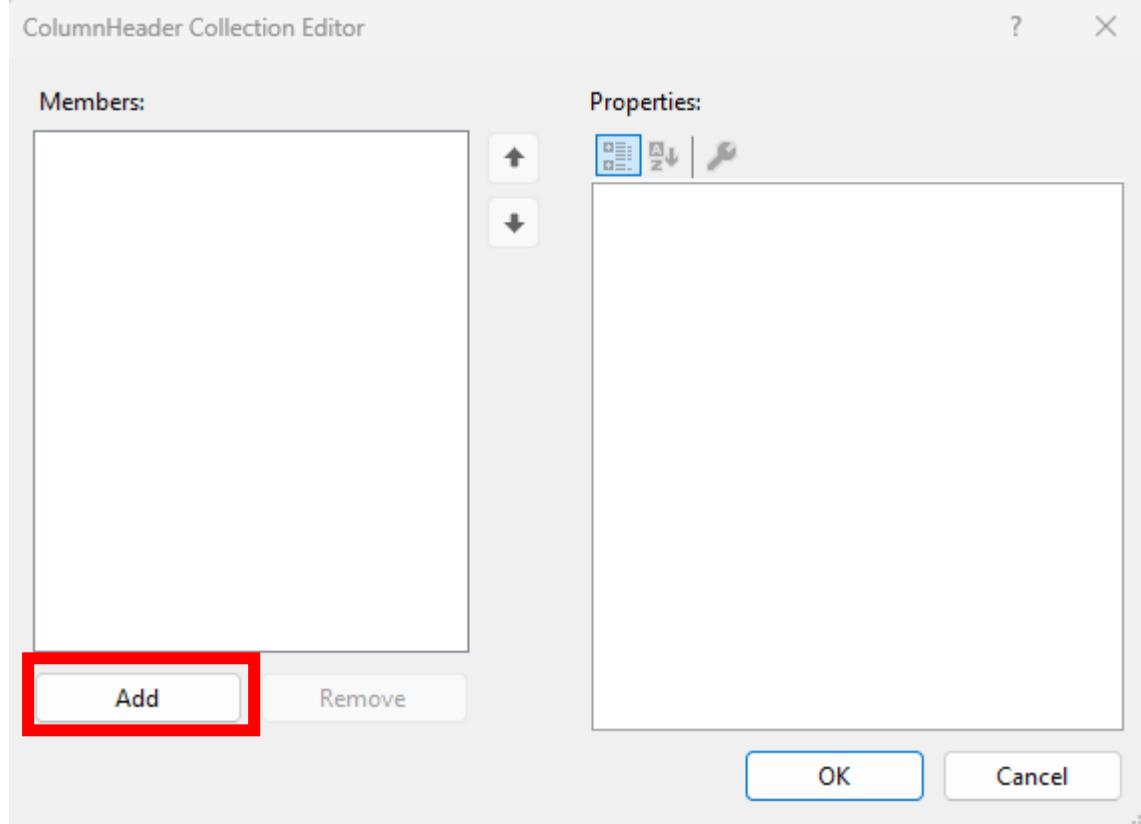
بعدين هنضيف الاتنين لـ list view image list





عشان الداتا تظهر معاك في ال list view لازم تحدد الاعمدة اللي هتظهر فيها الداتا  
لان ال list view هو اشبه بجدول فيه بيانات  
عشان تحدد الاعمده بندوس على edit Columns





## ColumnHeader Collection Editor

? X

## Members:

- |   |               |
|---|---------------|
| 0 | columnHeader3 |
| 1 | columnHeader4 |

## columnHeader4 properties:

DisplayIndex	1	
▼ Data		
> (ApplicationSetting)		
Tag		
▼ Design		
(Name)	columnHeader4	
GenerateMember	True	
Modifiers	Private	
▼ Misc		
ImageIndex	<input type="checkbox"/> (none)	
ImageKey	<input type="checkbox"/> (none)	
Text	Name	
TextAlign	Left	
Width	60	

Add

Remove

OK

Cancel

ID	Name

الخاصية دي بتخليلك لما تعمل **select** على عنصر معين بيختار الصف كله

**FullRowSelect** **True**

والخاصية دي بترسم شكل الجدول

**GridLines** **True**

ID	Name
1	Ali
2	Sayed
3	Mohamed
4	Hassan
5	Ahmed

نيجي بقى للاكواود

عاوزين لما نكتب id و name ونحدد ان كان female او male او نيجي ندوس على زرار ال add  
يضيف العنصر لل list  
فنيجي لزرار ال add

بس قبل مانبدا عايز اقولك انه ال list view يعني قاييمه من العناصر أيا  
كانت بقى العناصر دي ايه ا بتعبر عن ايه ال item ده بيكون كلاس اسمه

تاني حاجه خلينا متفقين عليها انه ال item ده أيا كان فهو بيعبّر عن record واحد من ال  
اللي في القاييمه

ال item ده ممكن تعمل جواه sub items زي ماكنا بنعمل في الشجره  
تمام ؟ تمام

طيب فاكر في مشروع البنك انك كنت بتعمل كلاس بحاله وبنقدر تربط فيه بس لمجرد انه يتخزن في  
ملف text وتسديعه في أي وقت؟

نفس الكلام هنا احنا كل اللي هنعمله انه في كل ضغطه على زرار ال add هنأخذ object من النوع  
list view item ونقعد نربط فيه وبعدين نخزنه في ال

طيب ازاي هنعمل ده ؟

تعالي نشوف الكود

```
private void btnAdd_Click(object sender, EventArgs e)
{
    if (String.IsNullOrEmpty(txtbxId.Text)
        || string.IsNullOrEmpty(txtbxName.Text)) {
        return;
    }
    ListViewItem item = new ListViewItem(txtbxId.Text.Trim());
```

ده الحدث بتاع زرار ال add

اول حاجه بنقوله له انه لو كان ال text box بتاع  
ال id او بتاع ال name فاضي اخرج وماماتعملش  
حاجه

ولو ما خرجش هنعمل object من ال

list view item ونديله الكلام اللي جوه ال  
بتاع ال id ويخرنـه بعد مايعمله trim  
يعني بعد مايشيل الفواصل الزرياده

```
if (rbMale.Checked)
{
    item.ImageIndex = 0;
}
else { item.ImageIndex = 1; }
```

بعدين بقوله لو اليوزر اختار الجنس ذكر هنحط  
صورة الولد ولو ختار اثـي هنحط صورة البنـت

```
item.SubItems.Add(txtbxName.Text.Trim());
```

طيب دلوقتي في حالتنا دي احنا عندنا subitem واحد وهو ال name فبقوـم واحد قيمته من ال item واضيفها لل text box

```
listView1.Items.Add(item);
```

وبعد كده بضيف ال item الجاهز لل list

```
txtbxId.Clear();
txtbxName.Clear();
txtbxId.Focus();
```

بعد ما يخلص بفضيـي ال text box وبخليـي ال id بيـجي عـال focus

كده خلاص ضيفنا عنصر لـ القايمـه

عاوزينـه نـحـفـ عـنـصر

```
private void btnRemove_Click(object sender, EventArgs e)
{
    if (listView1.Items.Count > 0) {
        listView1.Items.Remove(listView1.SelectedItems[0]);
    }
}
```

هنـجي لـ زـرار ال remove وبـقولـه يمسـح أول عنـصر من العـناـصـر الليـ اليـوزـر اختـارـها وبـما انهـ فيـ حـالـتـنـ اليـوزـر هـيـختـارـ عـنـصر واحدـ فـمـفيـشـ غـيرـهـ فـهـيـمسـحـهـ تـقدـرـ بـعـدـ كـدـهـ تـعـملـ loopـ يـمـشـيـ عـالـعـناـصـرـ كلـهاـ بـسـ الفـكـرـهـ هـنـاـ وـضـحتـ

زرار ال fill random عـاـوزـينـهـ يـضـيفـ 10ـ اـشـخـاصـ عـشـوـائـيـنـ

```
private void btnFillRandom_Click(object sender, EventArgs e)
{
    for (int i=1;i<=10; i++) {
        ListViewItem item = new ListViewItem(i.ToString());

        if (i%2==0) {
            item.ImageIndex = 0;
        }else { item.ImageIndex = 1; }

        item.SubItems.Add("person" + i);
        listView1.Items.Add (item);
    }
}
```

فـكـلـ الليـ بـعـملـهـ هوـاـ الليـ عـمـلـتـهـ فيـ زـرارـ ال for loop بـسـ المرـاديـ بـ add

طيب اخر حاجـهـ هـيـاـ انـناـ عـاـوزـينـهـ نـغـيـرـ ال viewـ لماـ نـدوـسـ عـلـيـ radio buttonـ منـ الليـ اـحـناـ عـامـلـيـنـهـ دـيـ مـافـيهـاشـ شـرحـ لـانـ ال list viewـ فـيهـ الـأـنـوـاعـ جـاهـزـهـ

```
private void rbDetails_CheckedChanged(object sender, EventArgs e)
```

```

{
    listView1.View = View.Details;
}

private void rbSmallIcons_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.SmallIcon;
}

private void rbTile_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.Tile;
}

private void rbLargeIcons_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.LargeIcon;
}

private void rbList_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.List;
}

```

وطبعا تقدر تعمل عمليات معينه لما اليوزر يختار عنصر من العناصر او لما يعمل **acion** معين على اي حاجه

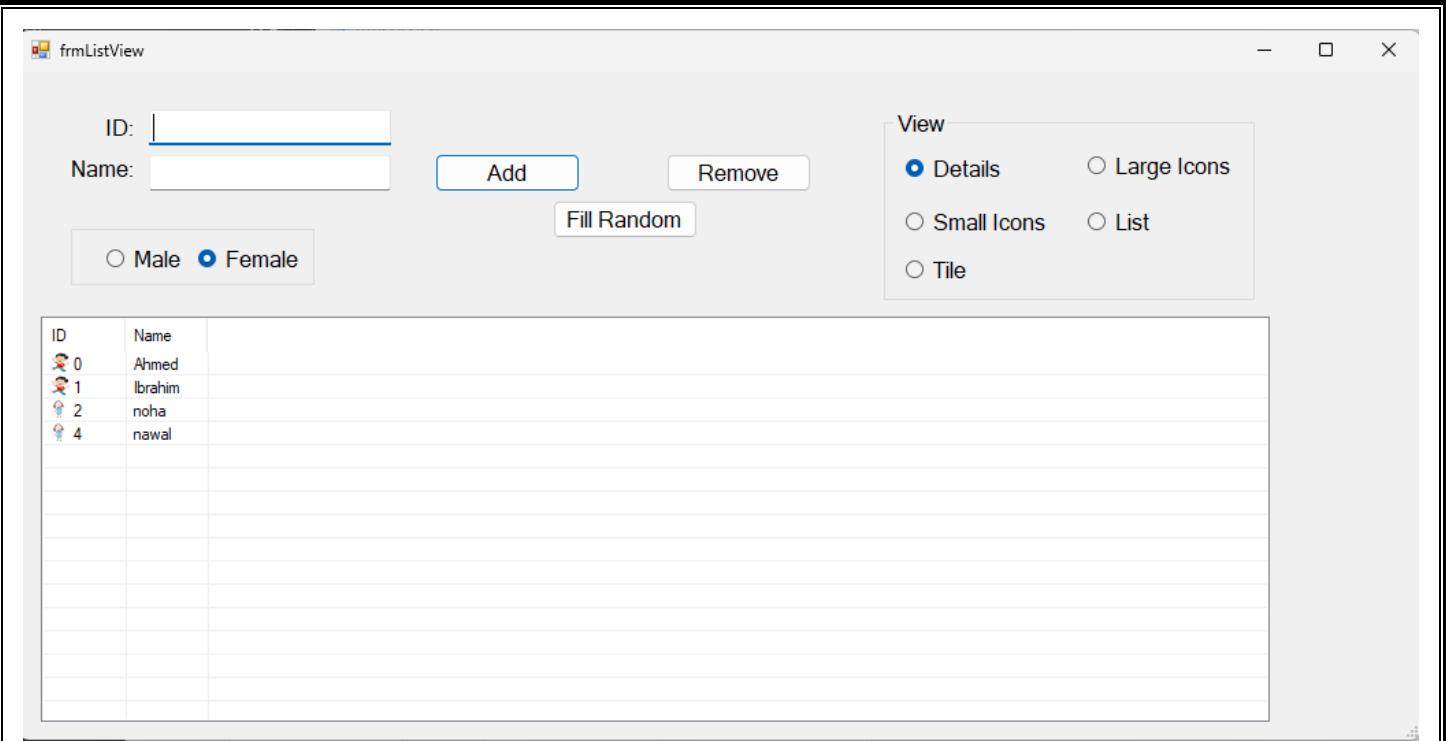
فيه بقى حاجه تانيه بعيد عن موضوع ال **list view** وهيا اني عايزة لما ادوس علي زرار ال **enter** في الكيورد يعملي **add** علي طول

بروح لخصائص الفورم بلاقي خاصيه اسمها **accept button** وبختار الزرار اللي عايزة البرنامج يدوس عليه لما ادوس علي زرار ال **enter**



فيه برضه **esc** وهيا لما ادوس علي زرار ال **cancel button**





وهي اكواد الازرار على بعضها

```
private void btnAdd_Click(object sender, EventArgs e)
{
    if (String.IsNullOrEmpty(txtbxId.Text)
        || string.IsNullOrEmpty(txtbxName.Text)) {

        return;
    }

    ListViewItem item = new ListViewItem(txtbxId.Text.Trim());

    if (rbMale.Checked)
    {
        item.ImageIndex = 0;
    }
    else { item.ImageIndex = 1; }

    item.SubItems.Add(txtbxName.Text.Trim());

    listView1.Items.Add(item);

    txtbxId.Clear();
    txtbxName.Clear();
    txtbxId.Focus();
}

private void btnRemove_Click(object sender, EventArgs e)
{
    if (listView1.Items.Count > 0) {
        listView1.Items.Remove(listView1.SelectedItems[0]);
    }
}

private void btnFillRandom_Click(object sender, EventArgs e)
{
    for (int i=1;i<=10; i++) {
        ListViewItem item = new ListViewItem(i.ToString());
```

```

if (i%2==0) {
    item.ImageIndex = 0;
} else { item.ImageIndex = 1; }

item.SubItems.Add("person" + i);
listView1.Items.Add (item);
}

private void rbDetails_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.Details;
}

private void rbSmallIcons_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.SmallIcon;
}

private void rbTile_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.Tile;
}

private void rbLargeIcons_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.LargeIcon;
}

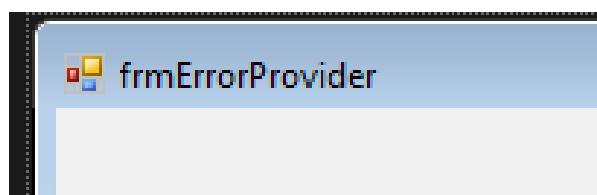
private void rbList_CheckedChanged(object sender, EventArgs e)
{
    listView1.View = View.List;
}

```

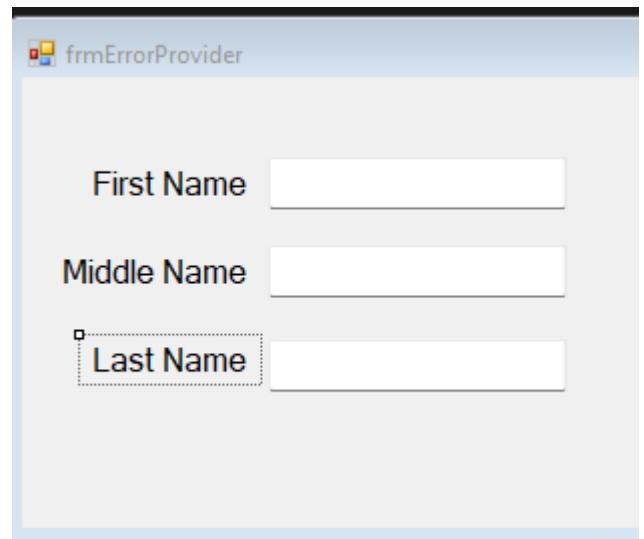
## ErrorProvider

ال message box ينفع في حته ال validation ويكون احسن من انك تستخدم

هنعمل فورم جديد



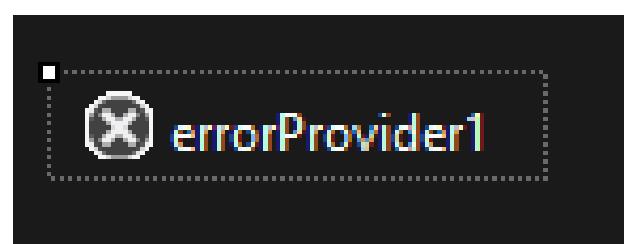
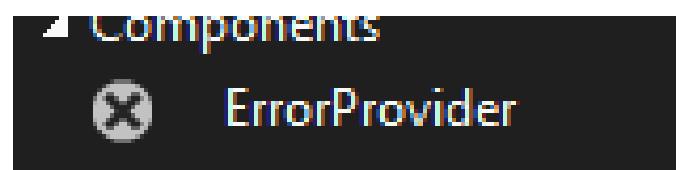
هنصيف فيه text box 3 و label 3



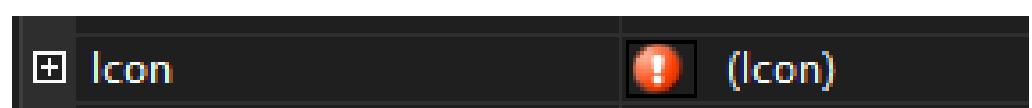
دلوقي انا عندي ال first name وال last name لازم اليوزر يدخلهم ولو انا واقف عال valid عايز البرنامج مايخليش اليوزر يخرج منه الا لو كان hint ولو دخل حاجه غلط تطلع له tool tip ولما يقف عليها بالماوس تطلع له



بضيف ال form error provider من هنا وبيكون بره ال



وبقدر اغير ال icon بتاعته من هنا

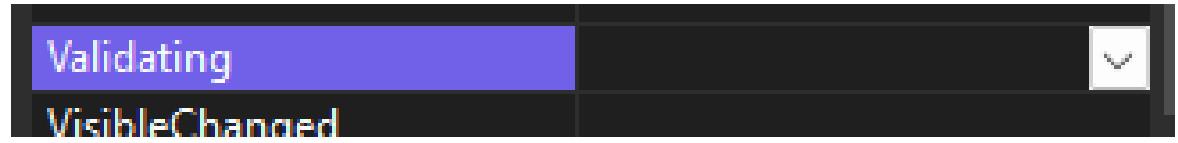


وده سرعة الوميض وبتقاس ال milli second



طيب دلوقي عايوذ لما اليوزر يسيب ال first name فاضي يطلع له ال error

هروح عال events بتاعت ال first name وبختار اسمه validating event



وبعدين نروح للكود بتاعه

```
private void txtbxFirstName_Validating(object sender, CancelEventArgs e)
{
    if (string.IsNullOrWhiteSpace(txtbxFirstName.Text))
    {
        e.Cancel = true;
        txtbxFirstName.Focus();
        errorProvider1.SetError(txtbxFirstName, "First Name Should Have A value");
    }
    else
    {
        e.Cancel = false;
        errorProvider1.SetError(txtbxFirstName, "");
    }
}
```

هنا بقوله انه لو ال FIRST NAME event بيشوف لو فيه معمول عليه بيبلغيه focus يكون عليه وبعدها يشغل ال provider ولو كده بيرجع يشغل كل حاجه تاني عادي وده الكود بتاع الزرارين

```
private void txtbxFirstName_Validating(object sender, CancelEventArgs e)
{
    if (string.IsNullOrWhiteSpace(txtbxFirstName.Text))
    {

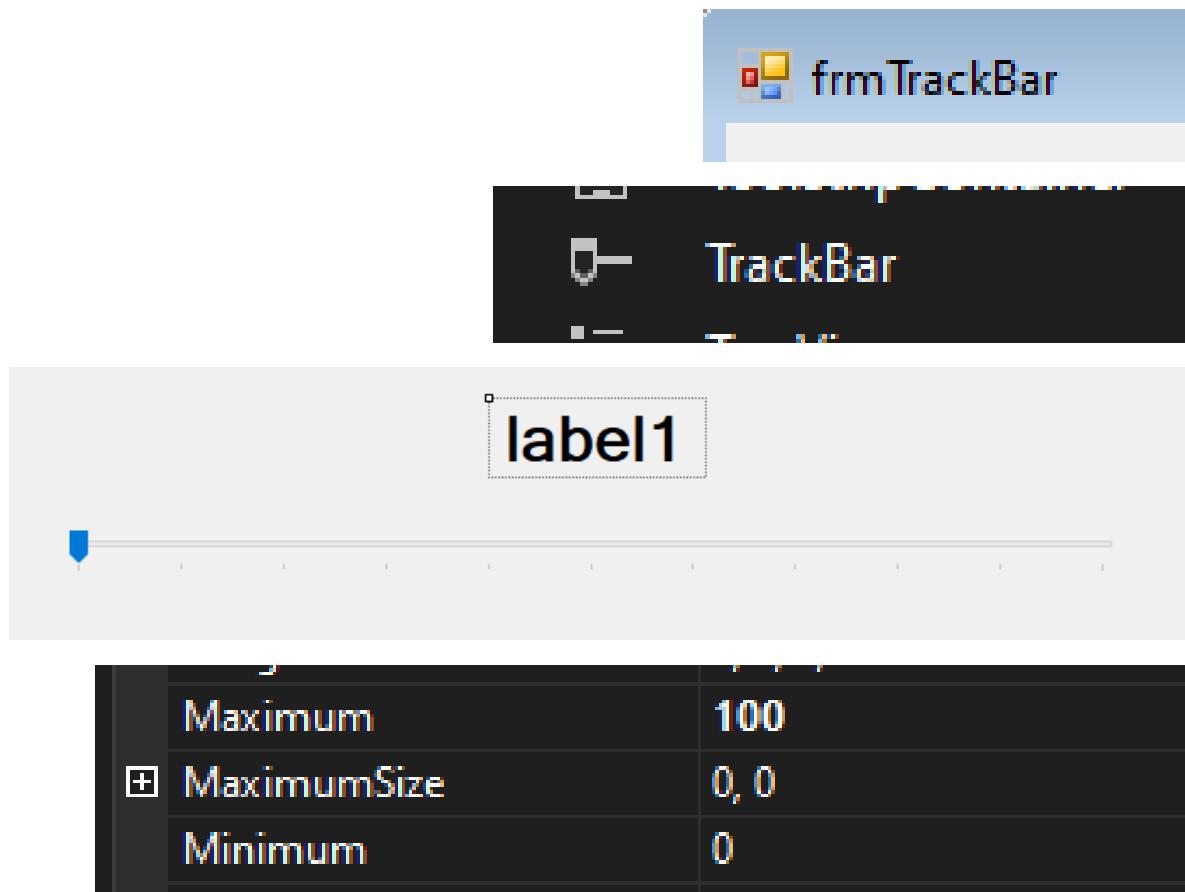
        e.Cancel = true;
        txtbxFirstName.Focus();
        errorProvider1.SetError(txtbxFirstName, "First Name Should Have A value");
    }
    else
    {
        e.Cancel = false;
        errorProvider1.SetError(txtbxFirstName, "");
    }
}

private void txtbxLastName_Validating(object sender, CancelEventArgs e)
{
    if (string.IsNullOrWhiteSpace(txtbxLastName.Text))
    {
        e.Cancel = true;
        txtbxLastName.Focus();
        errorProvider1.SetError(txtbxLastName, "First Name Should Have A value");
    }
    else
    {
        e.Cancel = false;
        errorProvider1.SetError(txtbxLastName, "");
    }
}
```

## TrackBar

ال track bar زي البتاع اللي بيוטي ويعلي الصوت او الشريط بتاع الفيديو

وليه progress bar زي ال maximum value و minimum value هوا وي ال progress bar بس انت اللي بتتحكم فيه ومش بيمشي غير لما انت تمشي  
وكالعادة تعالي نعمل فورم حديد بس خلي بالك لا يصدى



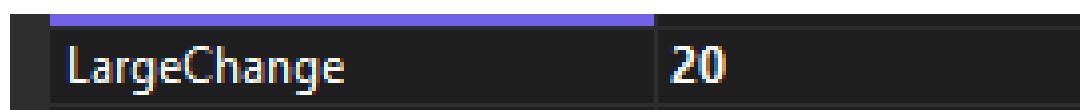
ال scroll event بتاعه اسمه

```
private void trackBar1_Scroll(object sender, EventArgs e)
{
    label1.Text=trackBar1.Value.ToString();
}
```

الخاصية small change لما ادوس على سهم يمين او شمال عاوزه يتغير بقد ايه

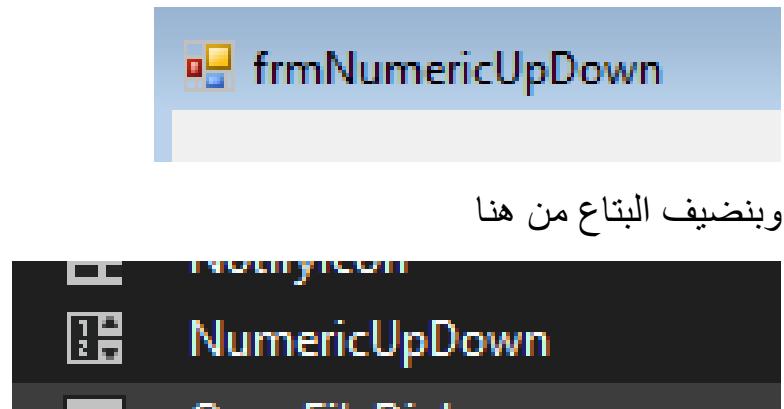


وال large change لما تدوس بالماوس على جانب الشريط



### NumericUpDown

ال numeric up down رقم وجنبه سهمين بيزود وينقص الرقم ده  
تعالي نعمل فورم



وبنضيف البتاع من هنا

Properties



NumericUpDown

NumericUpDown



فيه برضه increment و maximum و minimum عاوزه يزيد قد ايه كل ضغطة

Increment	1
Maximum	100
MaximumSize	0, 0
Minimum	0

وده ال event بتاعه

```
private void numericUpDown1_ValueChanged(object sender, EventArgs e)
{
    label1.Text = numericUpDown1.Value.ToString();
}
```

## TabControl

ال tab control بيفيدك انك تعمل اكتر من تبوب في نفس الفورم

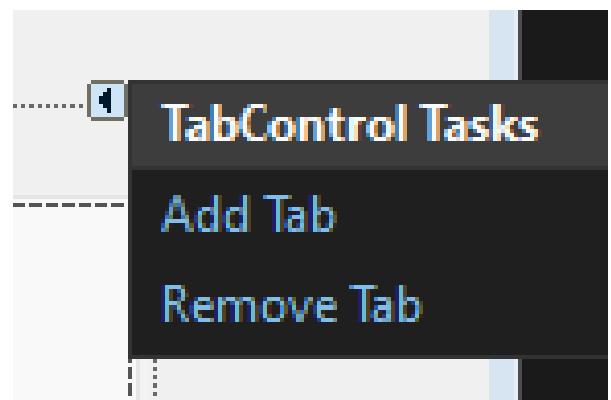
وتعالي نعمل فورم جديد



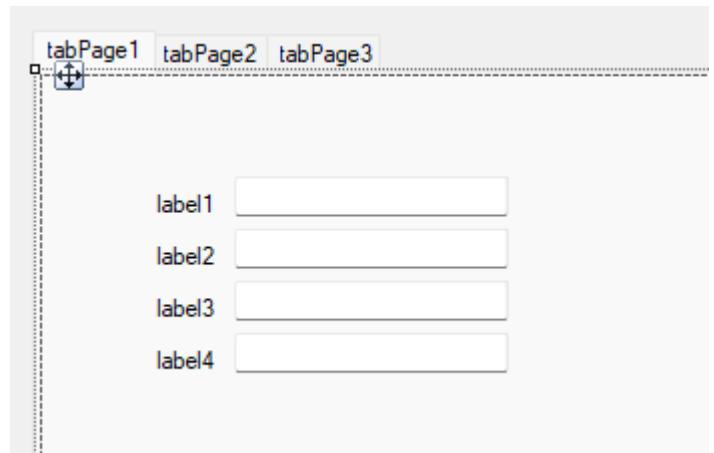
وبنضيف ال TAB CONTROL من هنا

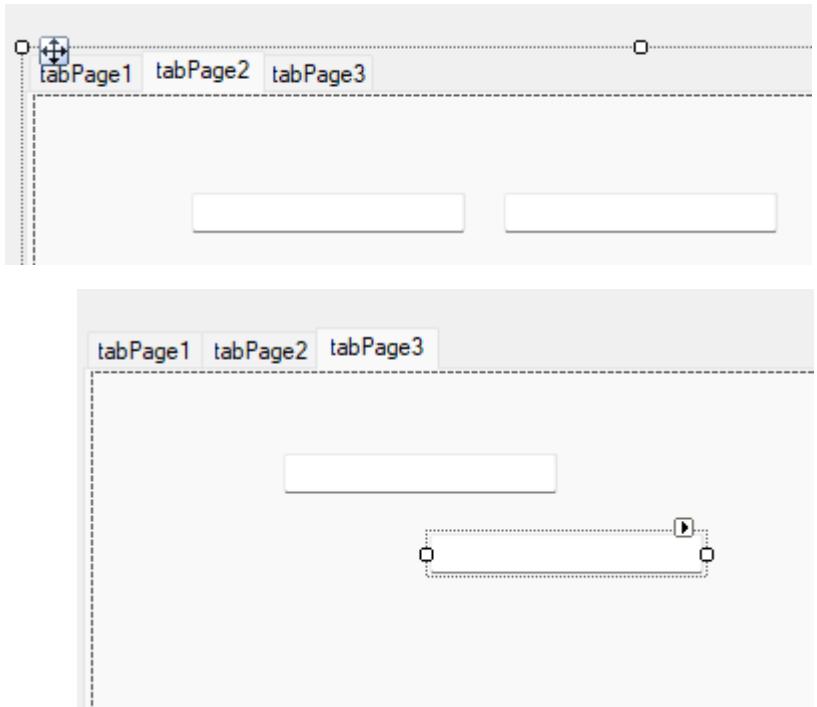


بتضيف tab جديد من هنا



بنحدد tab ونضيف فيه العناصر اللي عايزين نضيفها

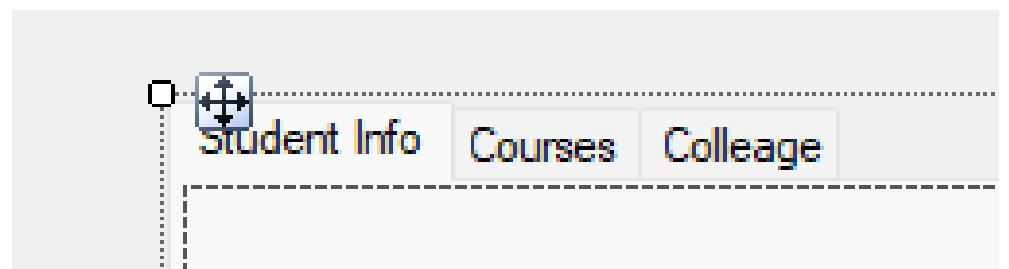
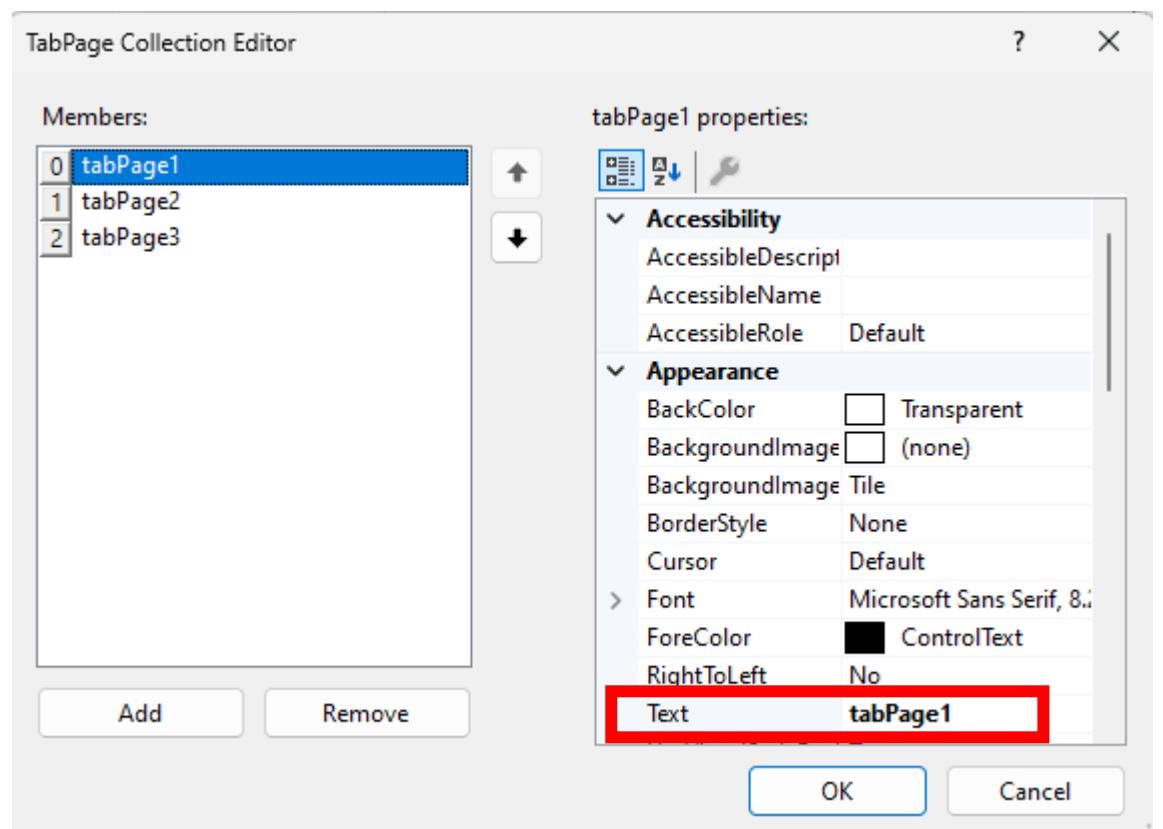




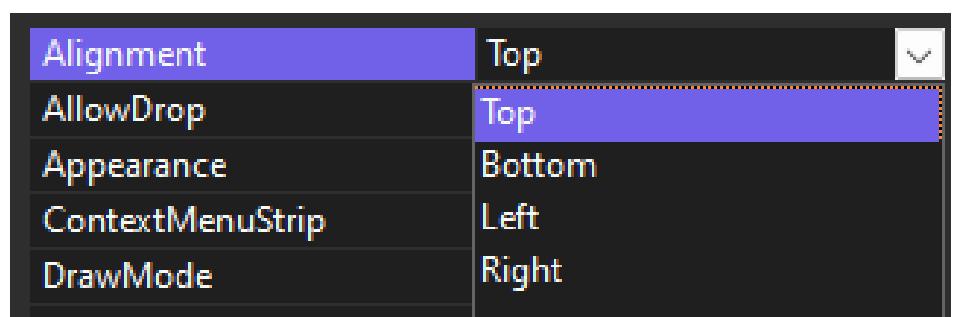
اقدر اغير اسمائهم من هنا



TabIndex	0
TabPages	(Collection)
TabCount	3



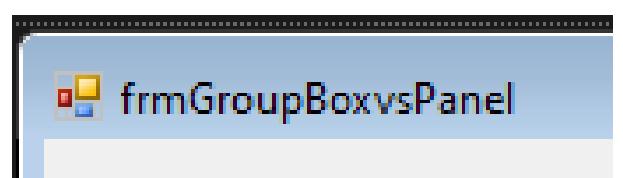
وتقدر بدل ماتخلיהם من فوق تخلיהם من الجنب او من تحت



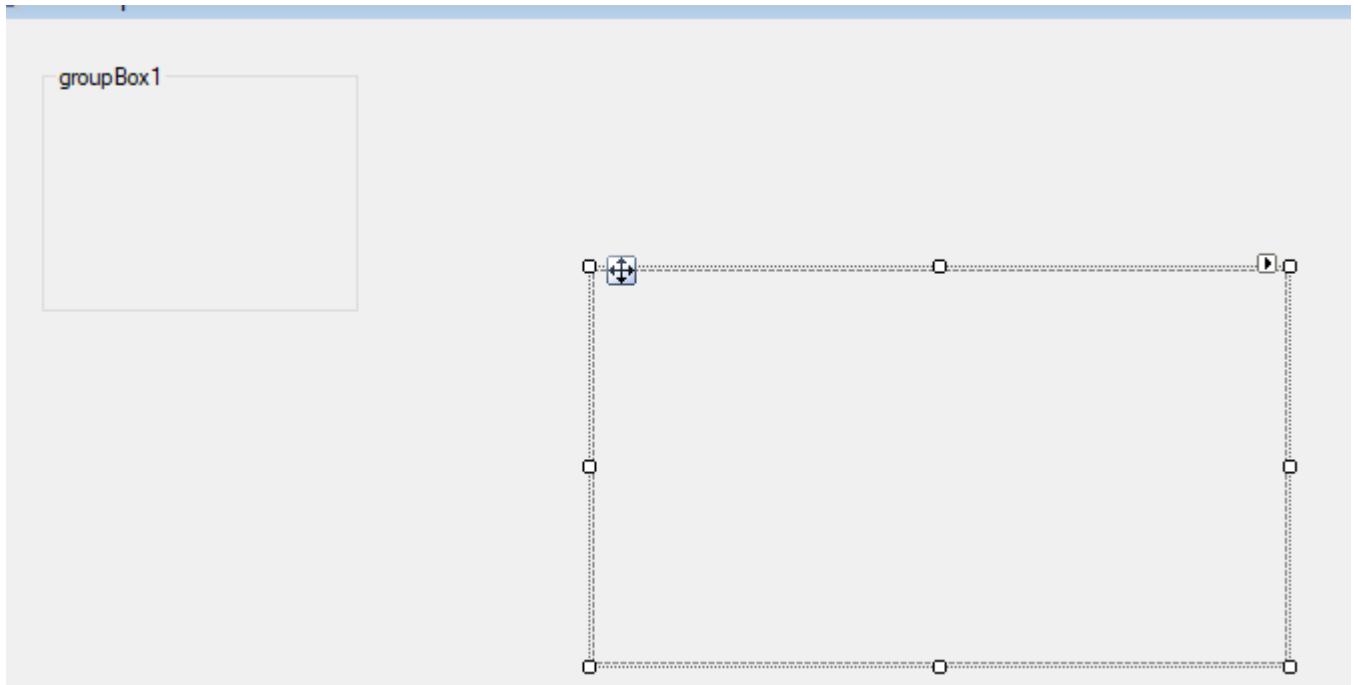
### GroupBox vs Panel

ال زيه زي ال container title group box بس مالوش scroll

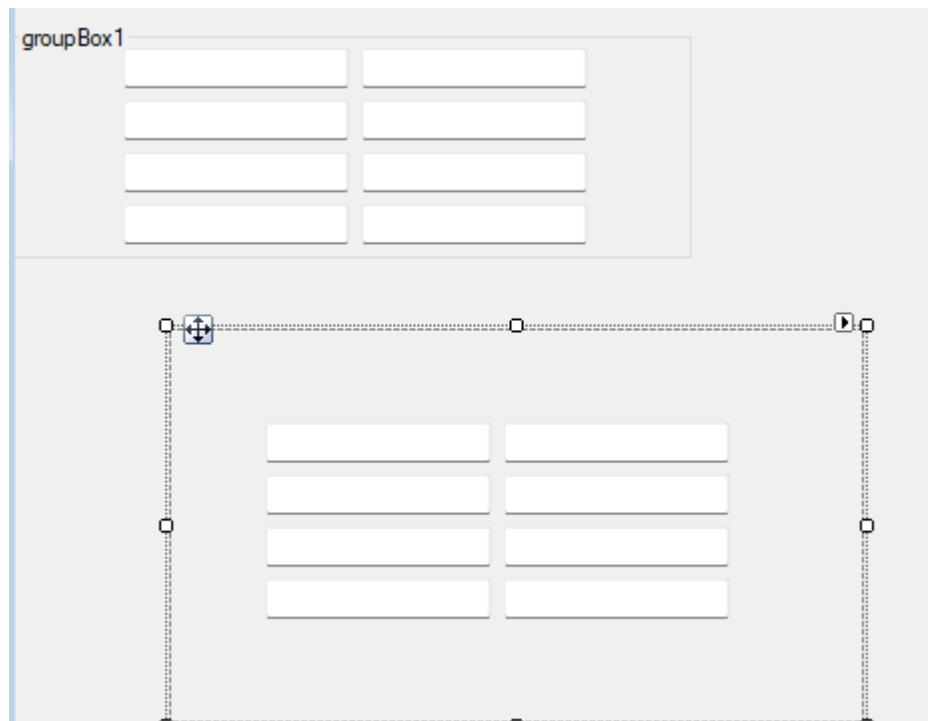
تعالعمول فورم



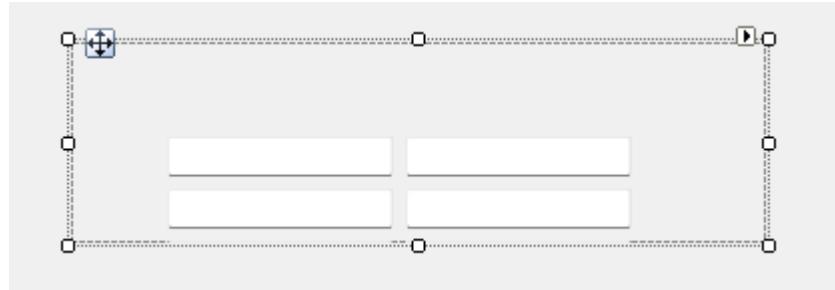
هنضيف panel و group box



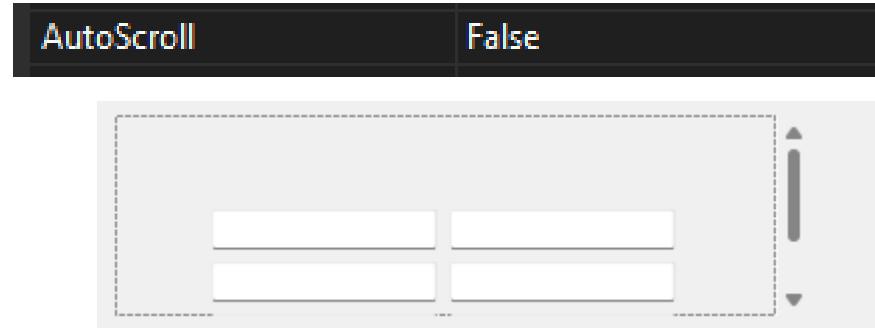
هنضيف فيهم أي عناصر



وبعدين صغينا ال panel عشان نشوف ال scroll



وبتفعل ال scroll من هنا



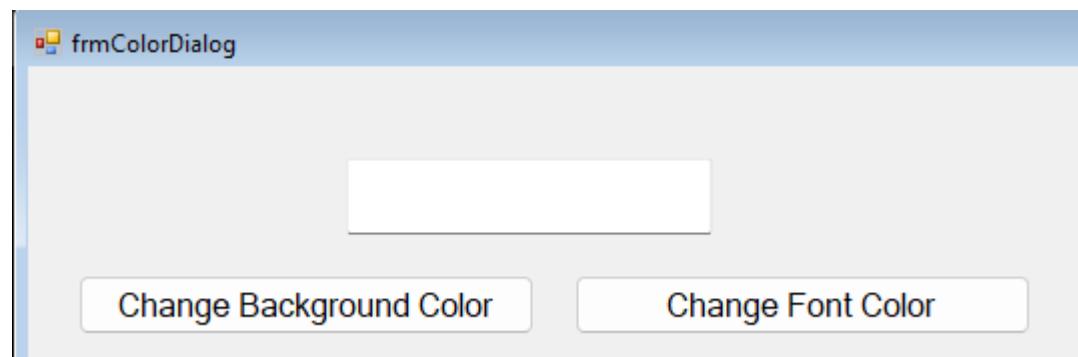
ال panel drag & drop بس ال group box مش بياخد منك

## Dialogs

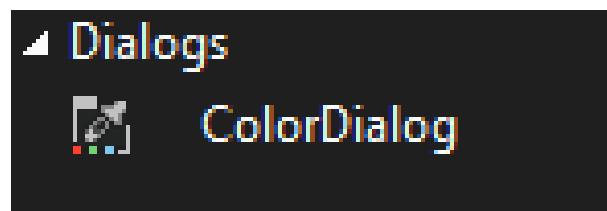
### ColorDialog

ال dialog زي برنامج بسيط بتقدر تعمل فيه حاجات معينه زي ال color dialog عباره عن واجهه بتختار منها لون معين

تعالي نعمل فورم



بنضيف ال color dialog من هنا وبيتحط بره الفورم

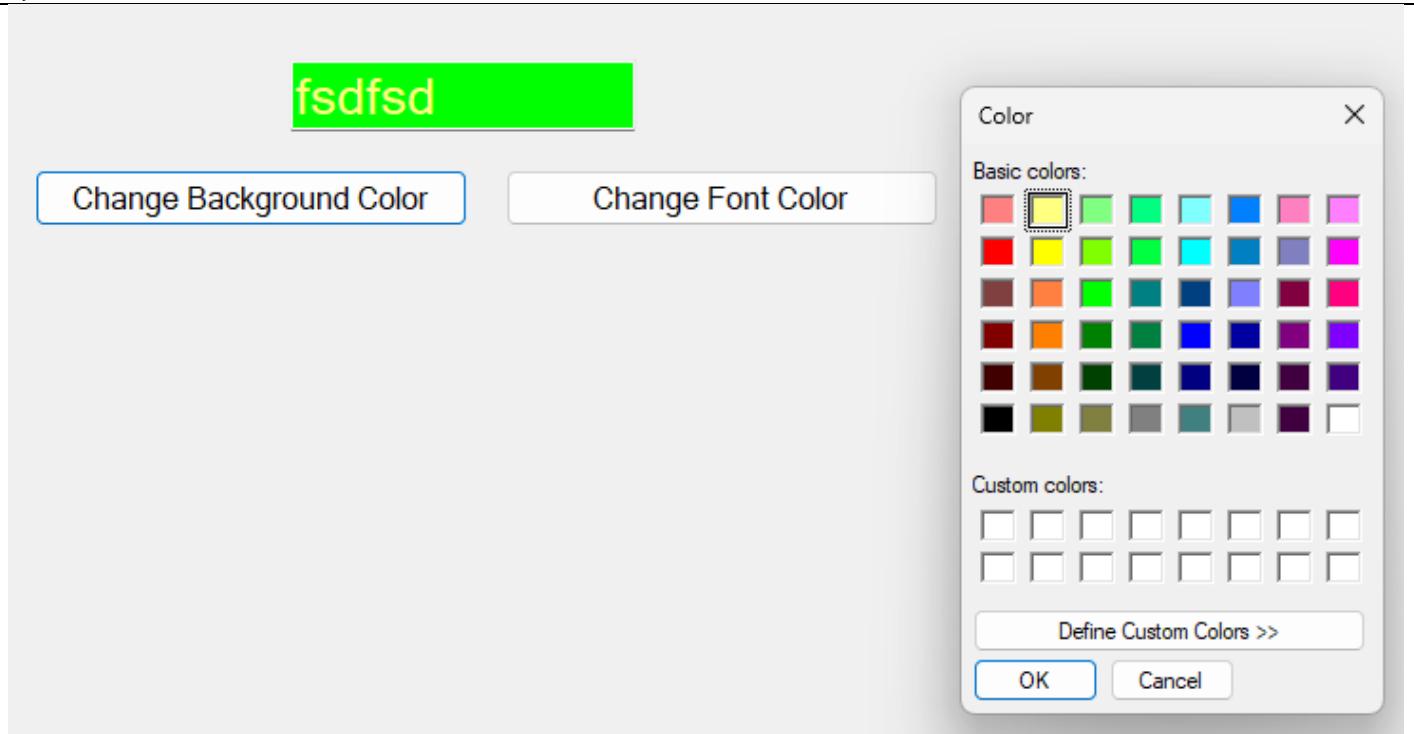




وده الكود بتاع الزرار

```
private void button1_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK) {
        textBox1.BackColor = colorDialog1.Color;
    }
}

private void button2_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK)
    {
        textBox1.ForeColor = colorDialog1.Color;
    }
}
```

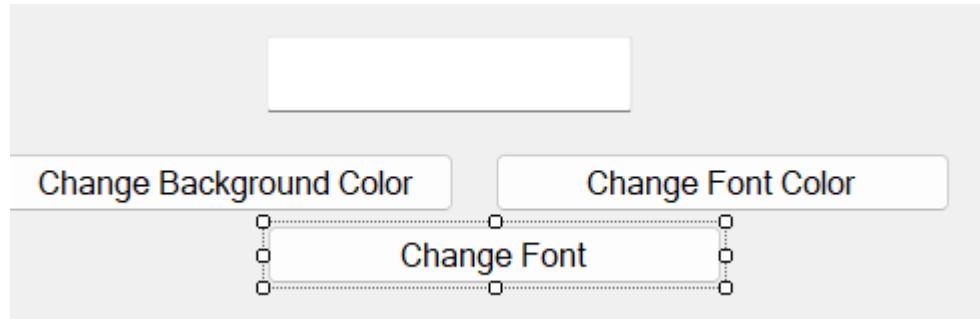


## FontDialog

ال font dialog ده اللي بيخليلك تتحكم في ال

مش هنعمل فورم جديد ⏪

هنعدل عالي عملناه قبل كده هنضيف فيه زرار



وهنضيف ال font dialog من هنا



```
private void button3_Click(object sender, EventArgs e)
{
    fontDialog1.ShowDialog = true;
    fontDialog1.ShowApply = true;
    fontDialog1.ShowEffects = true;
    fontDialog1.ShowHelp = true;

    fontDialog1.Font = textBox1.Font;

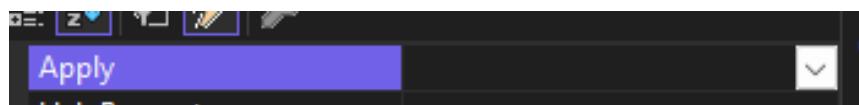
    if (fontDialog1.ShowDialog() == DialogResult.OK)
    {
        textBox1.Font = fontDialog1.Font;
    }
}
```

بظهر خانه اللون بتاع الخط  
بظهر زرار ال apply  
بظهر ال effects  
بظهر زرار ال help

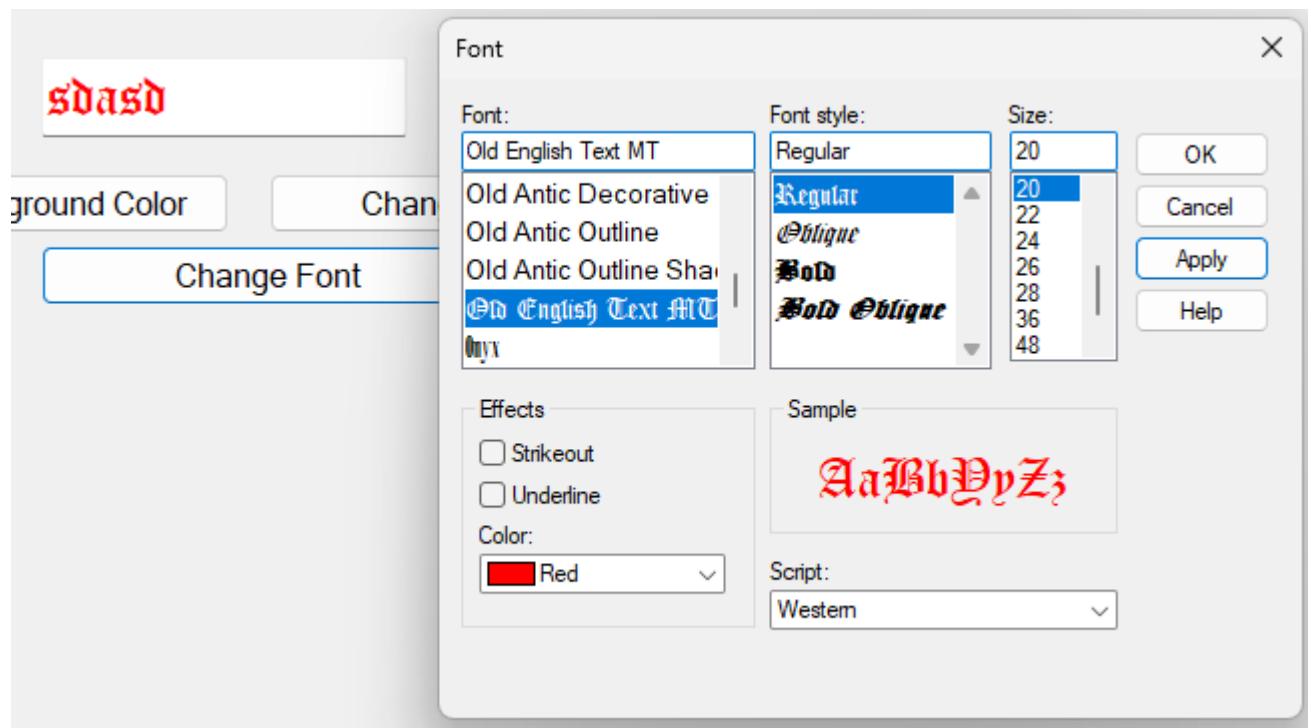
وهنا بخلي ال dialog يفتح علي نفس  
الخط اللي موجود في ال text box

وهنا بقوله يظهر ال dialog ويخللي ال  
textbox ياخذ الخط منه

فيه بقى زرار ال apply ممكن اعمل عليه event لوحده



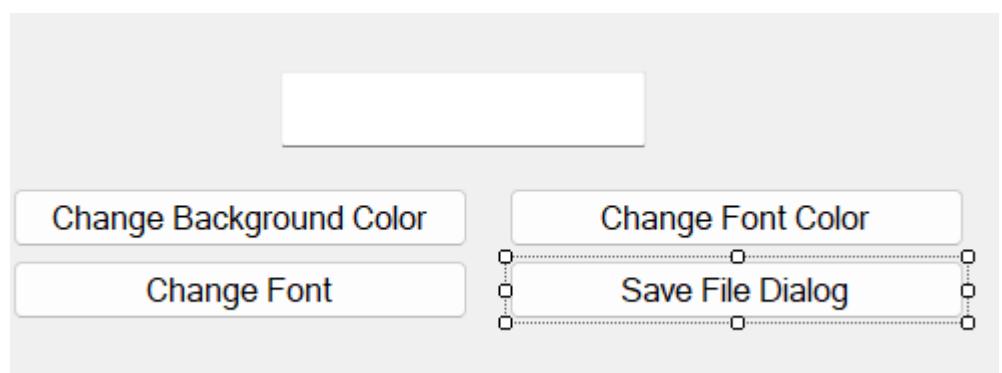
```
private void fontDialog1_Apply(object sender, EventArgs e)
{
    textBox1.Font = fontDialog1.Font;
    textBox1.ForeColor = fontDialog1.Color;
}
```



## FileSaveDialog

ال save as زى ال file save dialog بتاع الوورد والاكسيل كده  
هوا مش بيعمل حفظ لا هوا بيرجلك ال path بتاع الملف تاخده انت بقى تعمل عليه حفظ  
وبتقدر تتحكم بالصيغة اللي عايز تحفظ فيها الملف ولو الملف موجود او ملکش صلاحية انك تحفظ فيه  
بيطلعك رسالة

ومافيش فورم جديد



هنصيف ال dialog كده



وده ال event لو هخلية يطلعلي message box

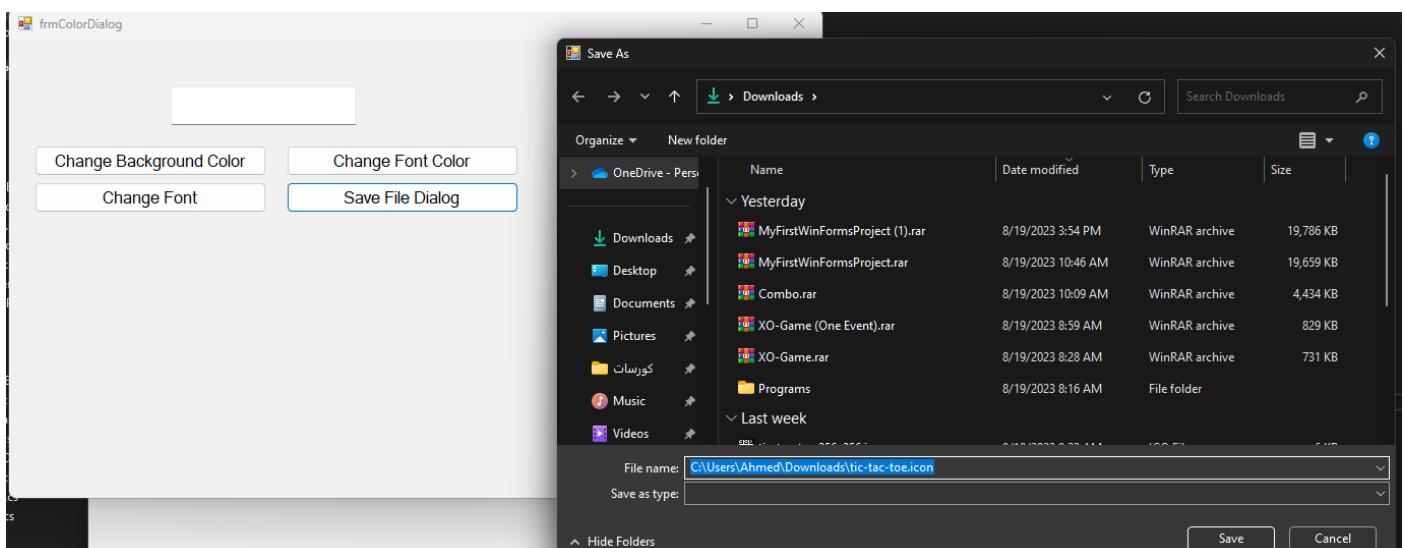
```

private void button4_Click(object sender, EventArgs e)
{
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
        MessageBox.Show(saveFileDialog1.FileName);
}

```

فيه بقى حاجات تانية

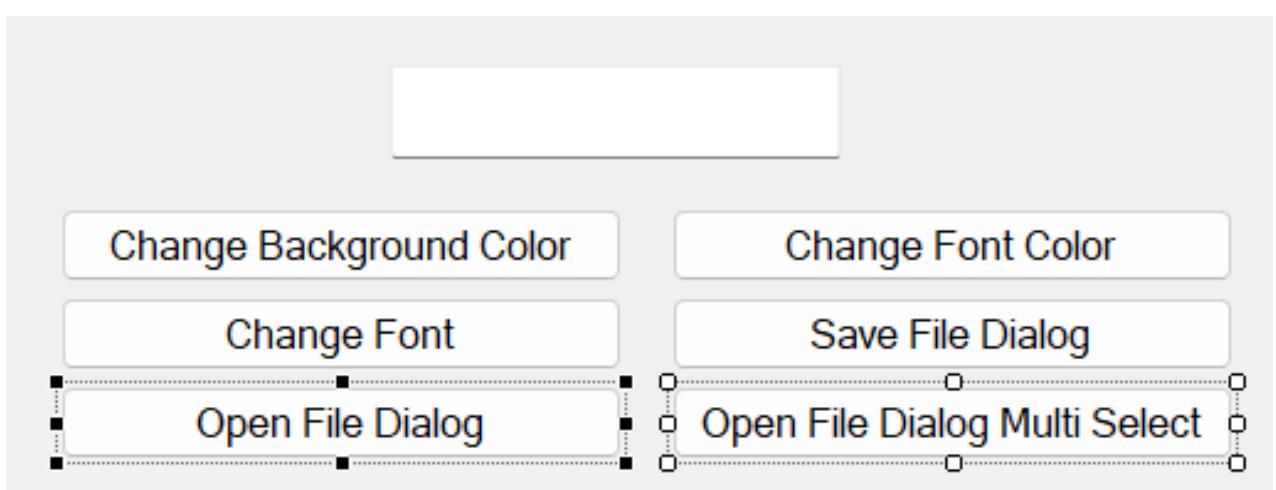
saveFileDialog1.InitialDirectory = @"c:\";	لو عايز ال dialog يفتح على مسار معين
saveFileDialog1.Title = "this is a title";	بغيير ال title بتاع ال dialog نفسه من فوق بدل مايكتب
saveFileDialog1.DefaultExt = "txt";	لو عايزه لما يفتح يحط من نفسه الصيغه اللي هيخفظ فيها الملف ولو مكتبتش الصيغ هوا بيحفظها من نفسه
saveFileDialog1.Filter = "txt files (*.txt) *.txt All files (*.*) *.*";	لو مش عايز تشوف الملفات كلها وعايز تشوف نوع معين زي ال text مثلا وبتظهر الفلاتر دي في الخانه اللي اسمها save type
saveFileDialog1.FilterIndex = 2;	لو عايز ال default filter يكون على نوع معين



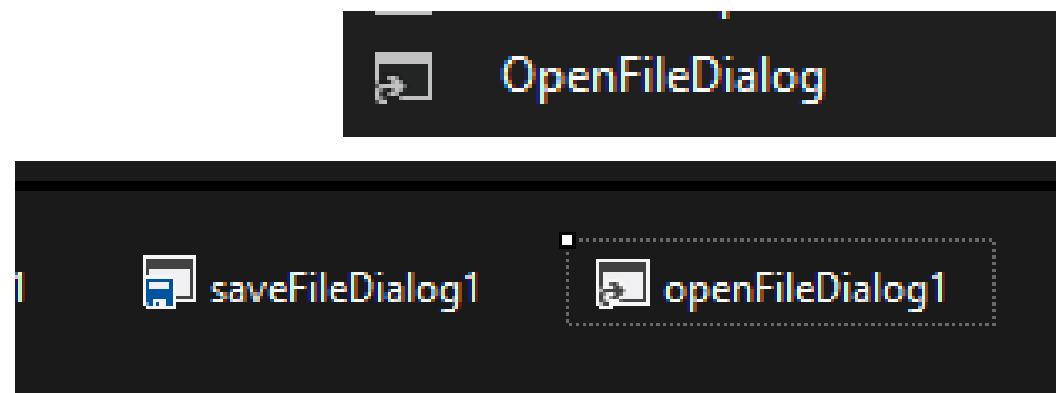
## FileOpenDialog

ال open file dialog لو عايز افتح ملف معين

وبيرجعلك المسار بتاع الملف

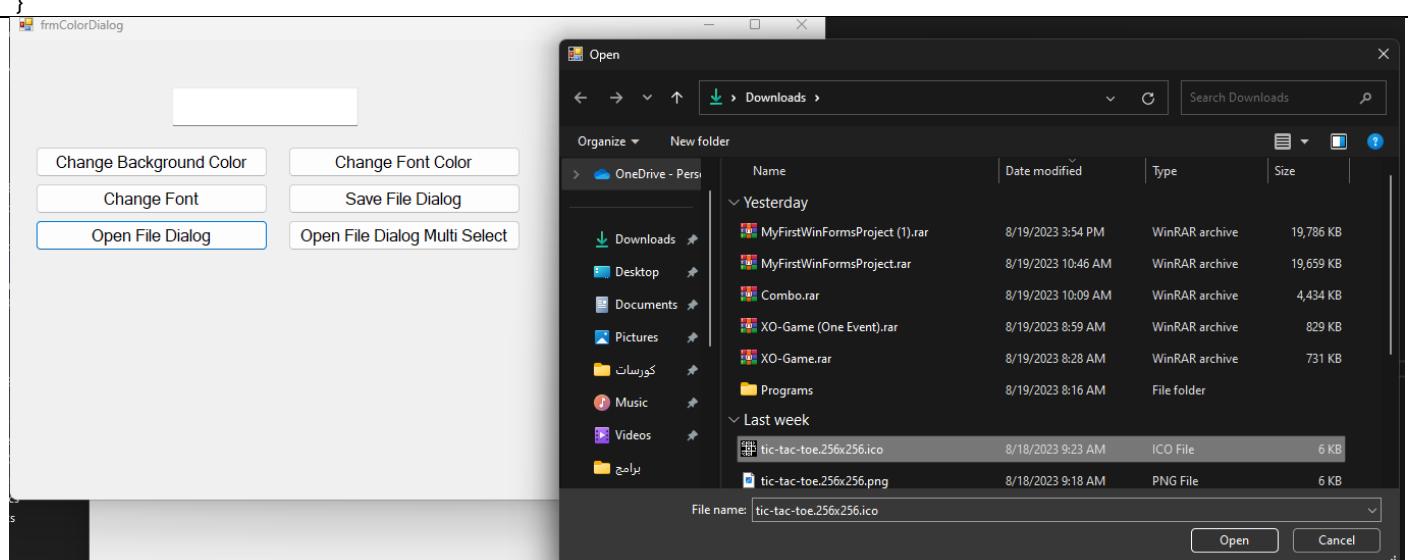


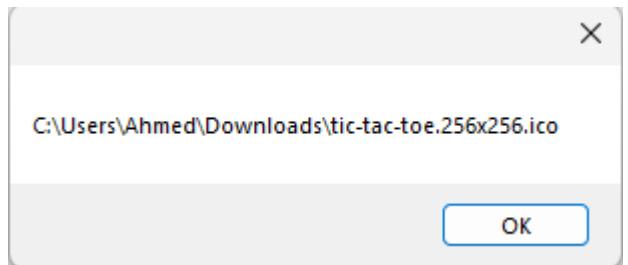
وبتضيف ال dialog من هنا



وال코드 بتاعه نفس الكود بتاع ال save file dialog

```
private void button5_Click(object sender, EventArgs e)
{
    /* openFileDialog1.InitialDirectory = @"c:\";
    openFileDialog1.Title = "this is a title";
    openFileDialog1.DefaultExt = "txt";
    openFileDialog1.Filter = "txt files(*.txt)|*.txt|All files(*.*)|*.*";
    openFileDialog1.FilterIndex = 2;*/
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        MessageBox.Show(openFileDialog1.FileName);
    }
}
```





عشان تقدر تختار اكتر من ملف

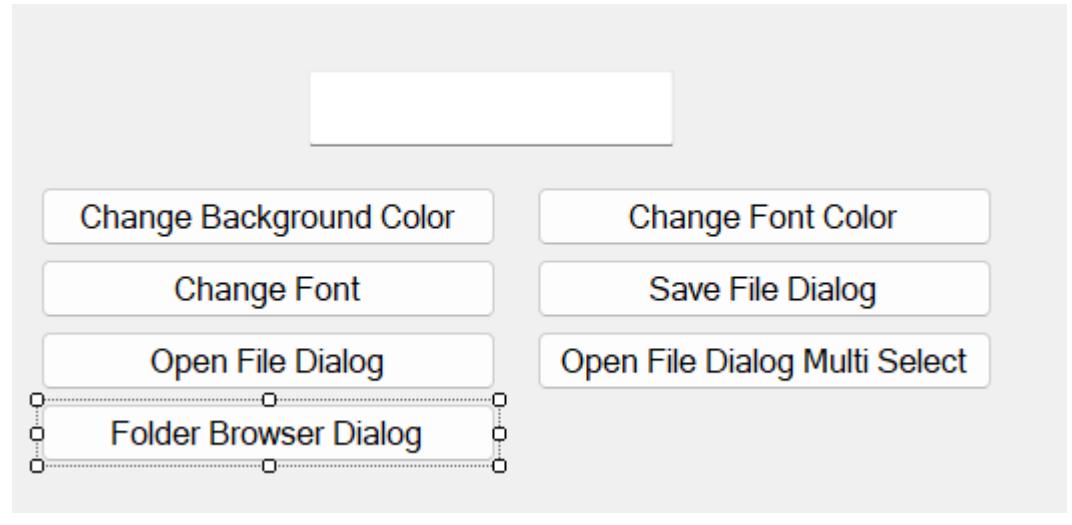
```
private void button6_Click(object sender, EventArgs e)
{
    openFileDialog1.Multiselect = true;

    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        foreach (string File in openFileDialog1.FileNames)
        {
            MessageBox.Show(File);
        }
    }
}
```

ما اعتدش الموضوع محتاج شرح

## FolderBrowserDialog

ال folder browser dialog بيسحلك انك تتصفح الملفات اللي عندك وعمول لـ folders وبقدر منه تضيف ملفات وتأخذ المسار بتاعهم



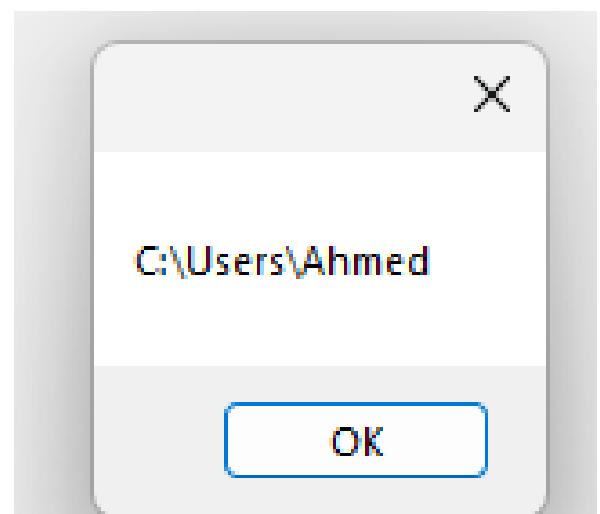
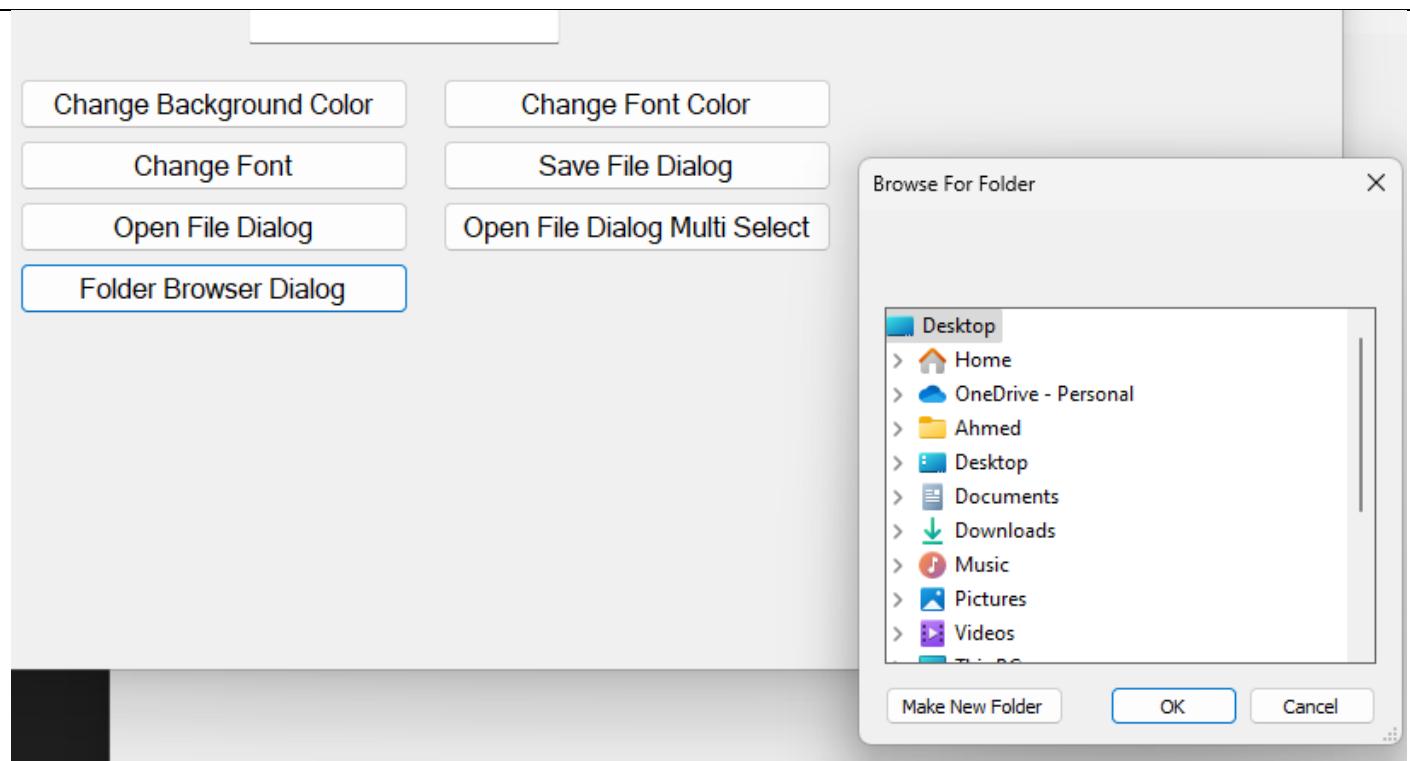
```
private void button7_Click(object sender, EventArgs e)
{
```

```

folderBrowserDialog1.ShowNewFolderButton = true;

if (folderBrowserDialog1.ShowDialog() == DialogResult.OK)
{
    MessageBox.Show(folderBrowserDialog1.SelectedPath);
}
}

```



## **Multiple Documents Interface (MDI) Container**

### **MDI Container**

ال mdi اختصار ل multiple documents interface

ده شغال منظم لـ forms بدل ما هيأ مرمي في الشارع

ال visual studio نفسه هو عباره عن mdi form بتفله بيغفل كل حاجه موجوده جواه

تعالي نعمل مشروع جديد

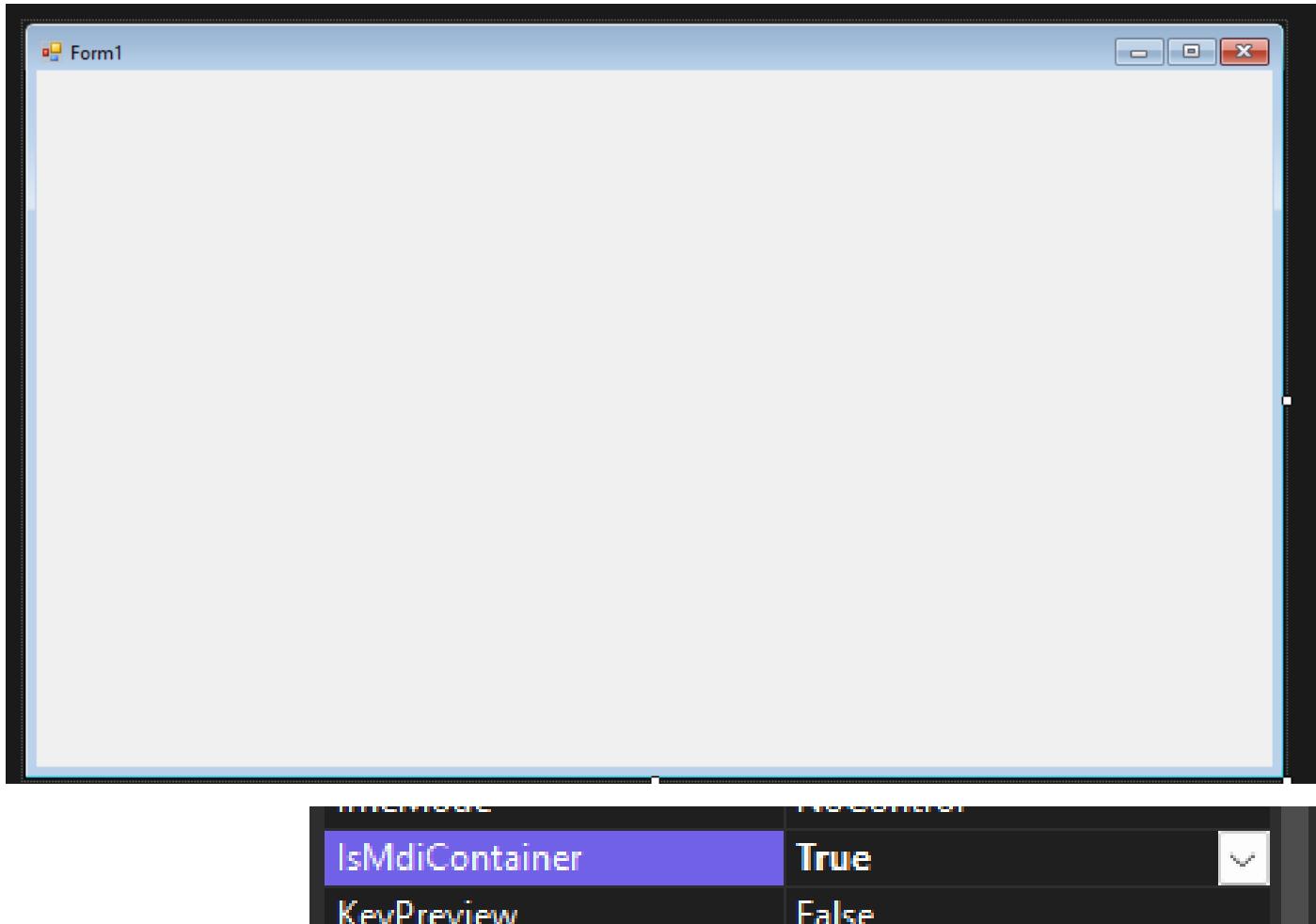
Project name

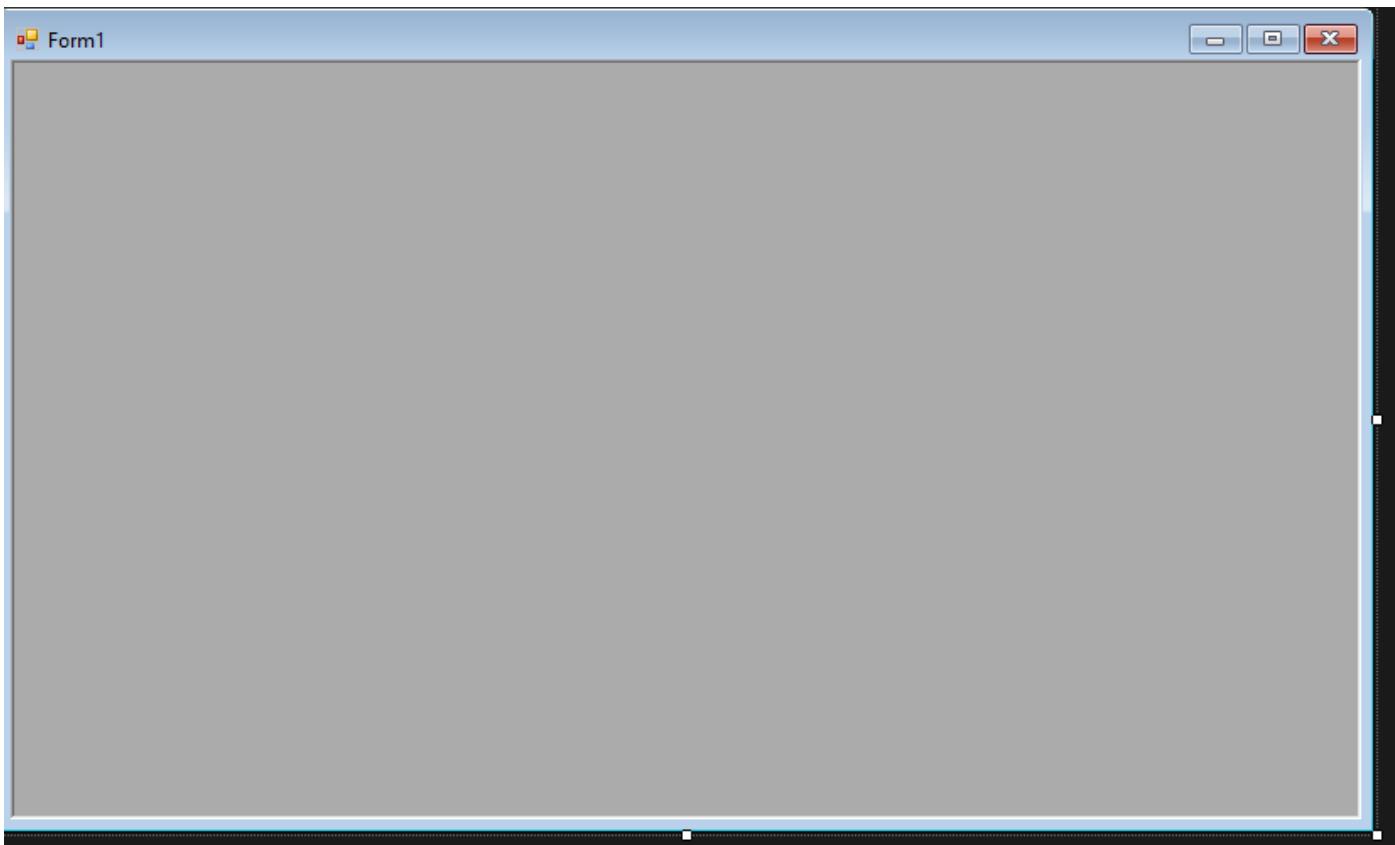
MDI Sample

Location

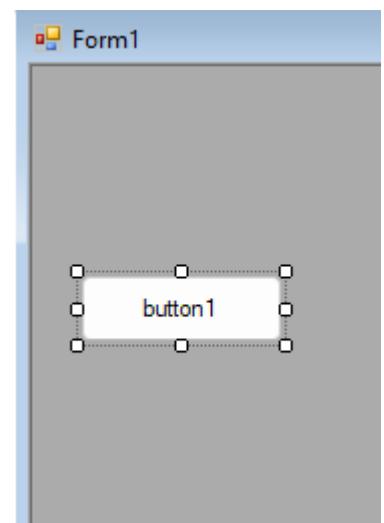
عملنا فورم عادي

الفورم ده بقى احنا ممكن نقلبه ل mdi container

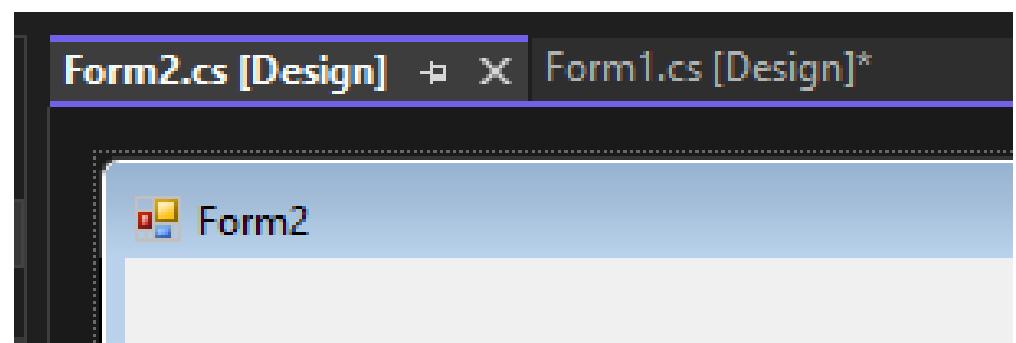




كده الفورم اتقلب بقى mdi form تقدر تضيف فيه قوائم و forms تانيه  
تعالي نحط فيه زرار و عادة مش بيتحط فيهم زرار بيتحط فيهم قوائم

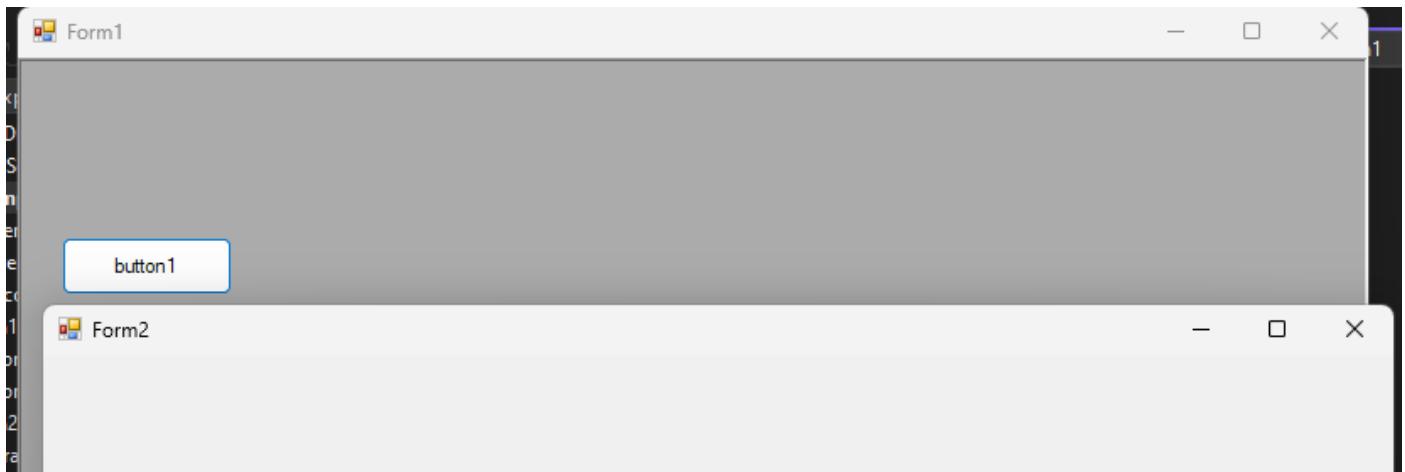


طيب تعالي نعمل form تاني ونخلي الزرار يفتحه



```
private void button1_Click(object sender, EventArgs e)
{
    Form2 form2 = new Form2();
    form2.Show();
}
```

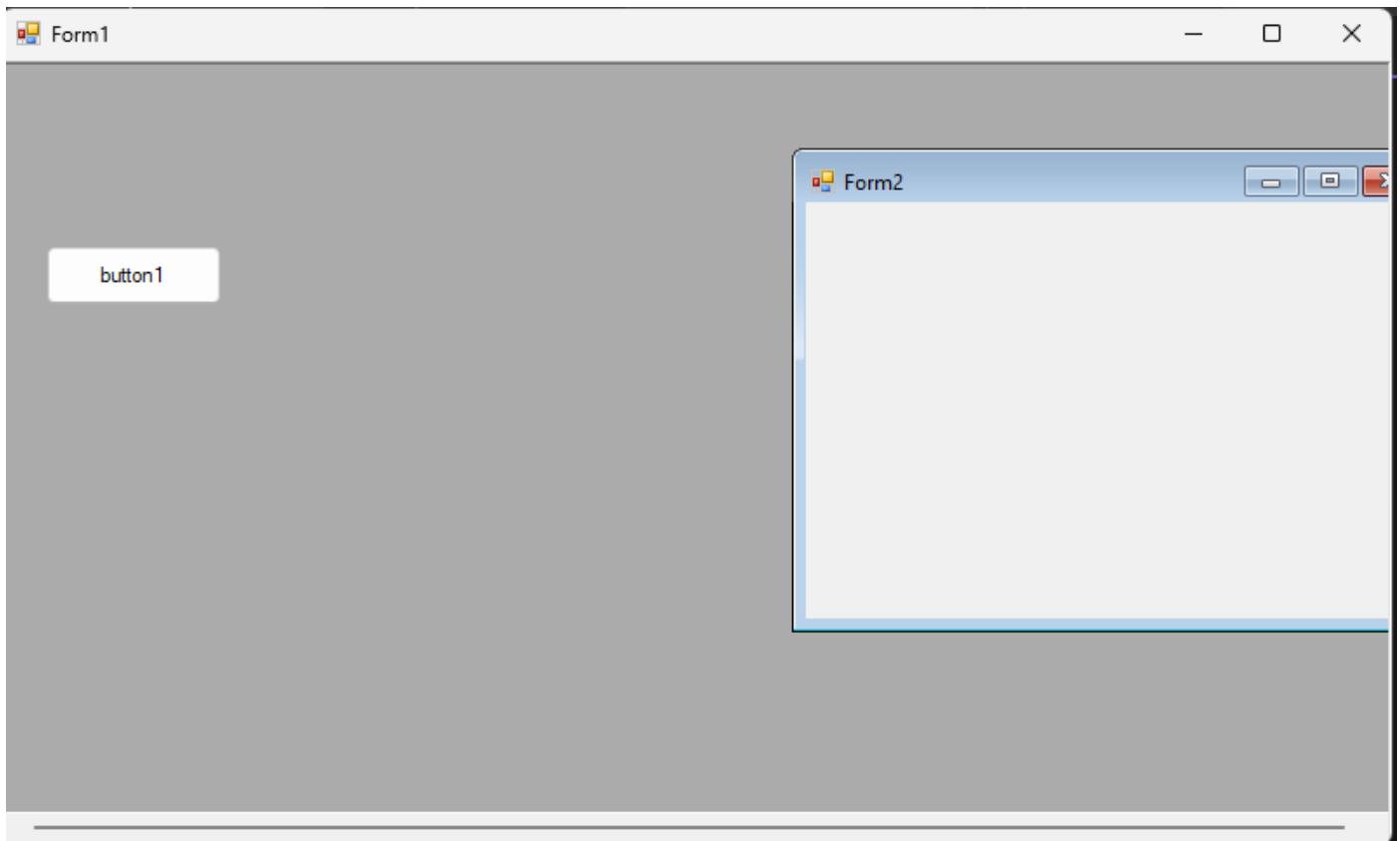
لو جينا فتحنا هيفتح عادي



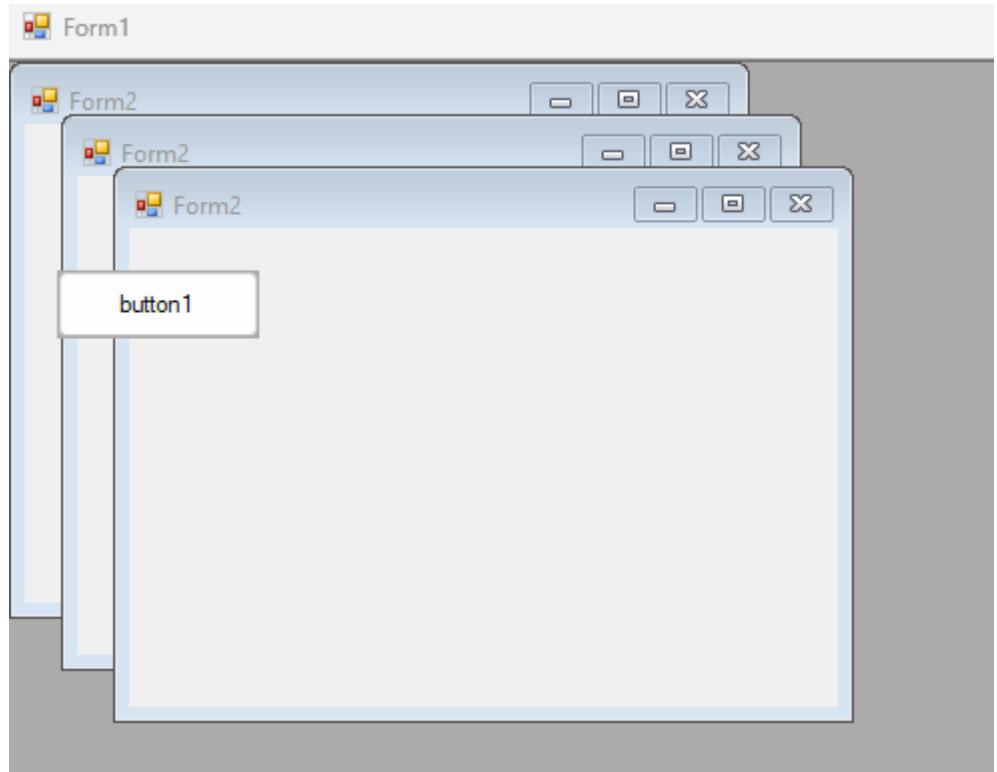
طيب المفروض لما تصغر الفورم الأساسي يقفل كل ال forms الثانيه وعشان نعمل كده لازم نروح لل  
ونعرف ال MDI parent form2

```
private void button1_Click(object sender, EventArgs e)
{
    Form2 form2 = new Form2();
    form2.MdiParent = this;
    form2.Show();
}
```

بده اصبح الفورم الثاني داخل اطار الفورم الأول ولو الفورم الاب انقل الابن هيتقلب



وتقدر تفتح نفس الفورم اكتر من مره



لو مش عايز افتح الفورم اكتر من مره بعرفه بره الزرار زي كده

```
Form2 form2 = new Form2();
private void button1_Click(object sender, EventArgs e)
{
    form2.MdiParent = this;
    form2.Show();
}
```

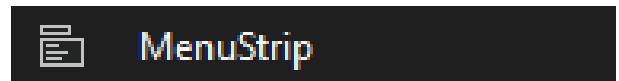
## Menus

## MenuStrip

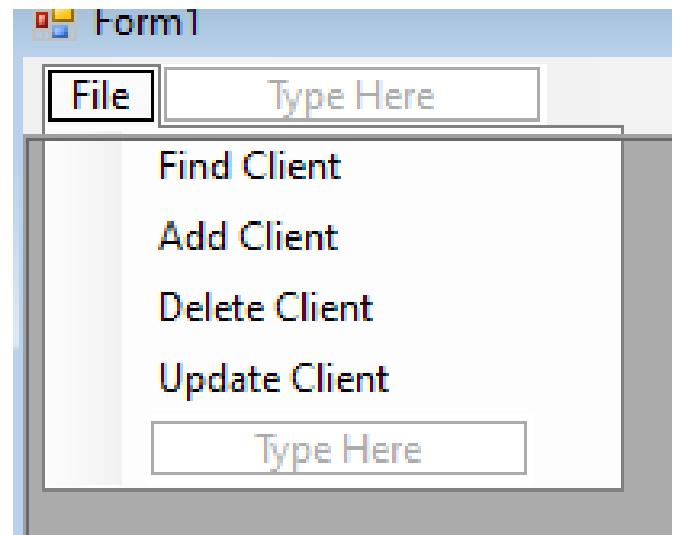
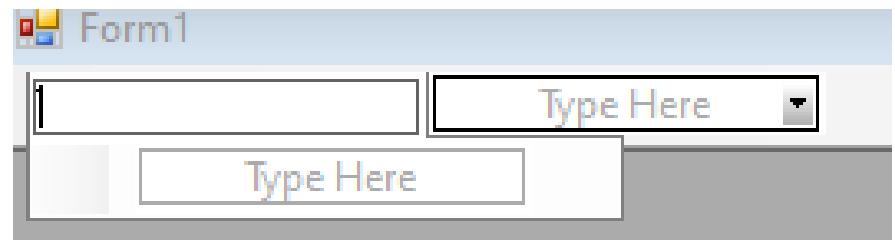
هنعمل menu بتقدر تطلع منه shortcut icon وتقدر تحططها sub menus

عندنا ال mdi container اللي كنا عاملينه اللي هو ده

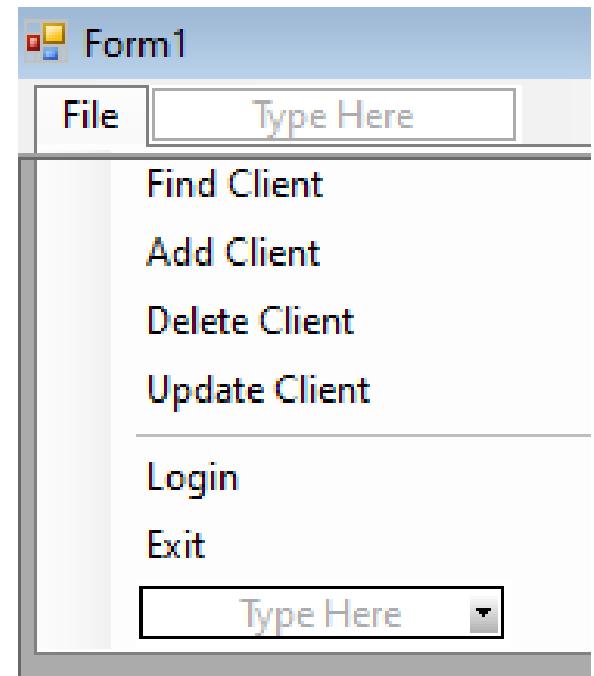
بنضيف ال menu strip من هنا

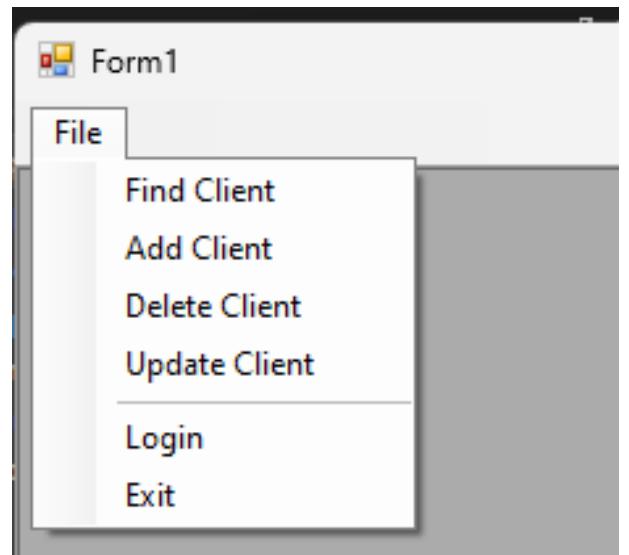


بس وتقعد تكتب اسم القائمه وال حاجات اللي تحتها وبختار يالما تضيف في الجنب او تحت



عایزین نصیف ال login بس عایزین نحط separator قبلها  
یدوب تكتب - (ناقص) وبعدها دوس enter



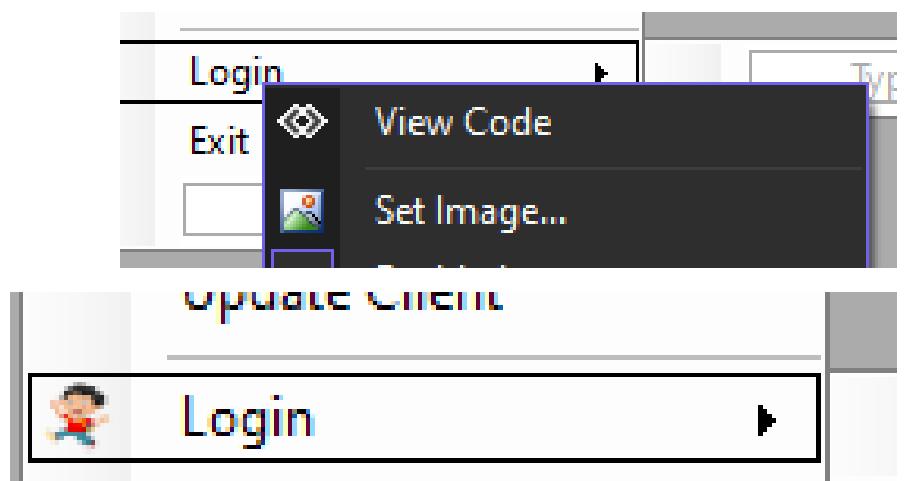


كل item في القائمه ليه on click تقدر تعمل فورم لكل واحده وبعدين تظهرها بس هنا عشان نخلص  
هنعمل message box

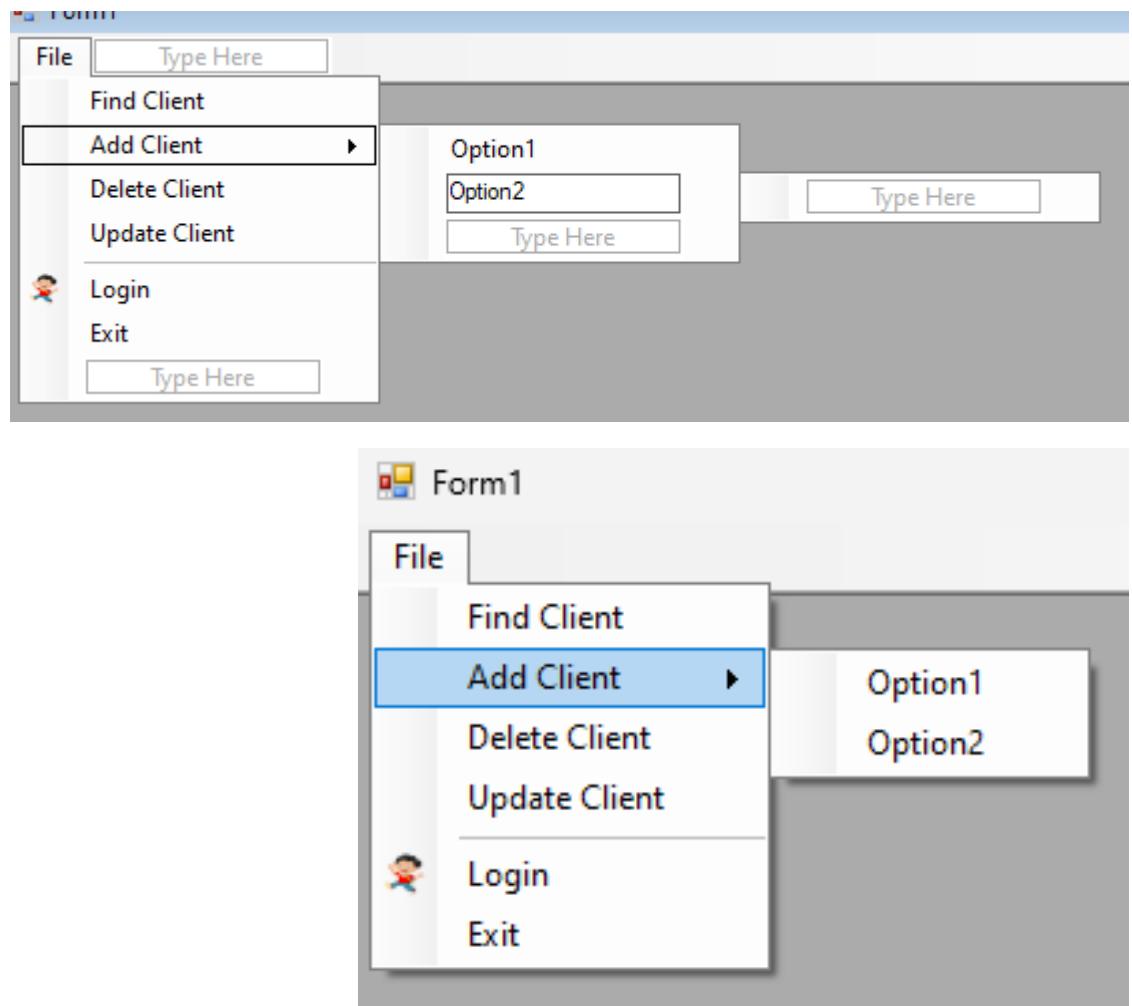
```
private void findClientToolStripMenuItem_Click(object sender, EventArgs e)
{
    MessageBox.Show("Find Client is Here");
}

private void addClientToolStripMenuItem_Click(object sender, EventArgs e)
{
    MessageBox.Show("Add New Client is Here");
}
```

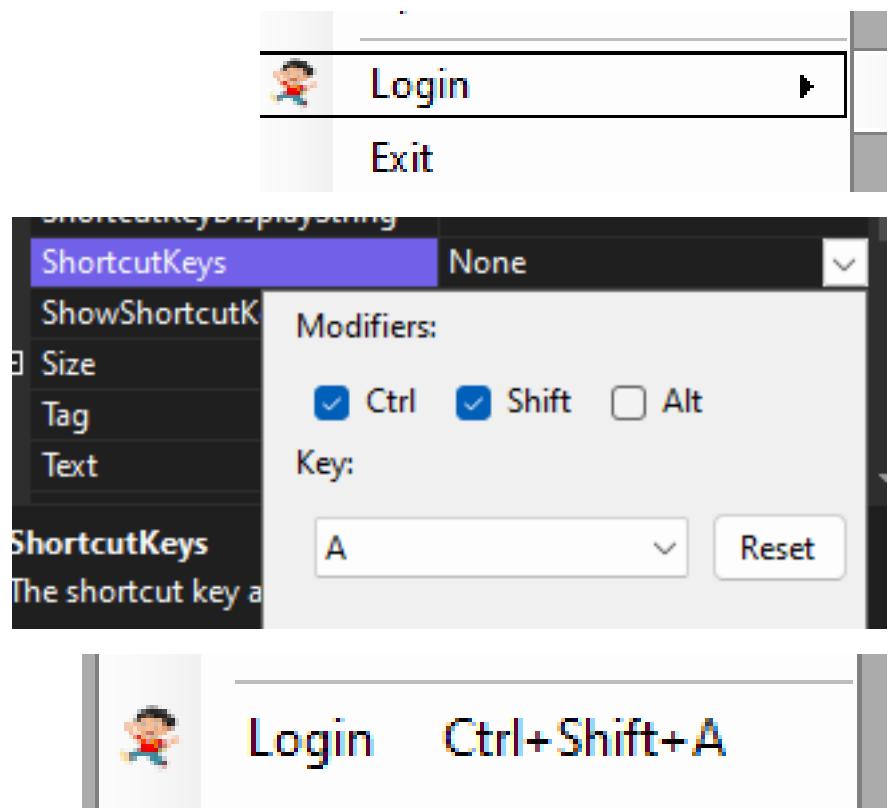
تقدر تحط icon لاي عنصر فيهم يدوب كليك يمين وتخترar set image



تقدر تضيف sub menu وكله بالماوس



لو عايز اعمل short cut بحدد العنصر وبروح لـ properties



فيه برضه لو دوست علي زرار ال alt وبيجيلاك خط تحت حرف معين لو ضغت عليه بيفتحهالك  
عشان تعملها بنتيجي قبل الحرف وبتكتب & زي كده

File

Type Here

&Find Client

File

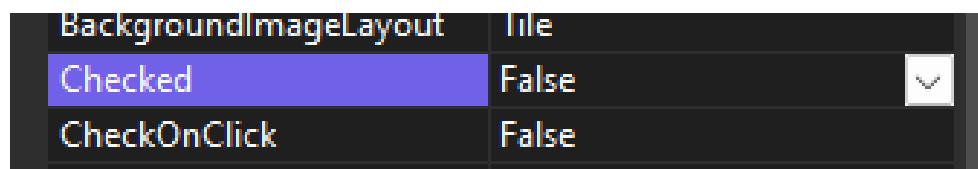
Type Here

Find Client ▾

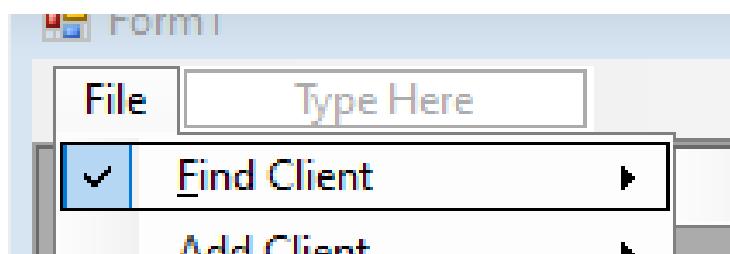
Add Client ▾

فيه خصائص تانية زي انك تحط drop down list بس اقعد قلب فيها وشوف

دي بتخليلك تعمل item على ال check



وال check on click بتخليلك تعمل لما المايوزر يدوس عليها



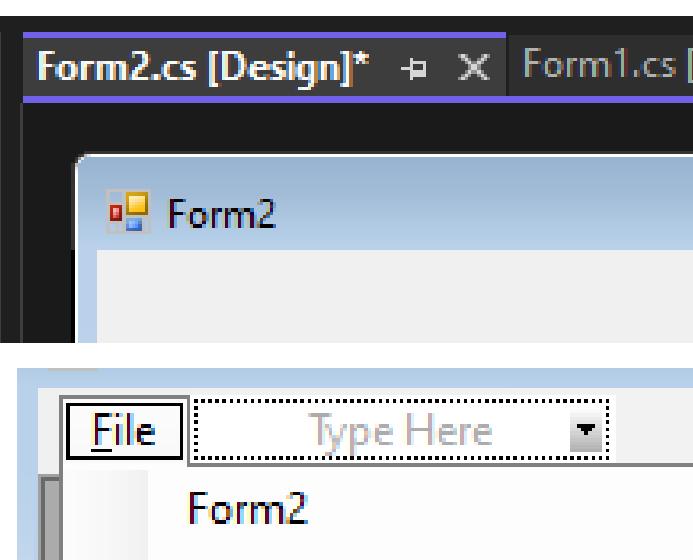
وممكن حسب ال صلاحيات ماتظهرش العناصر للمايوزر

## Context Menu

دي القائم اللي بتظهر لما تدوس كليك يمين شبه لما تيجي تفتح ملف كده

وبتظهر حسب السياق اللي انت فيه

تعال نعمل فورم ونخليله يفتح من القائمه بتاعت المشروع اللي فات

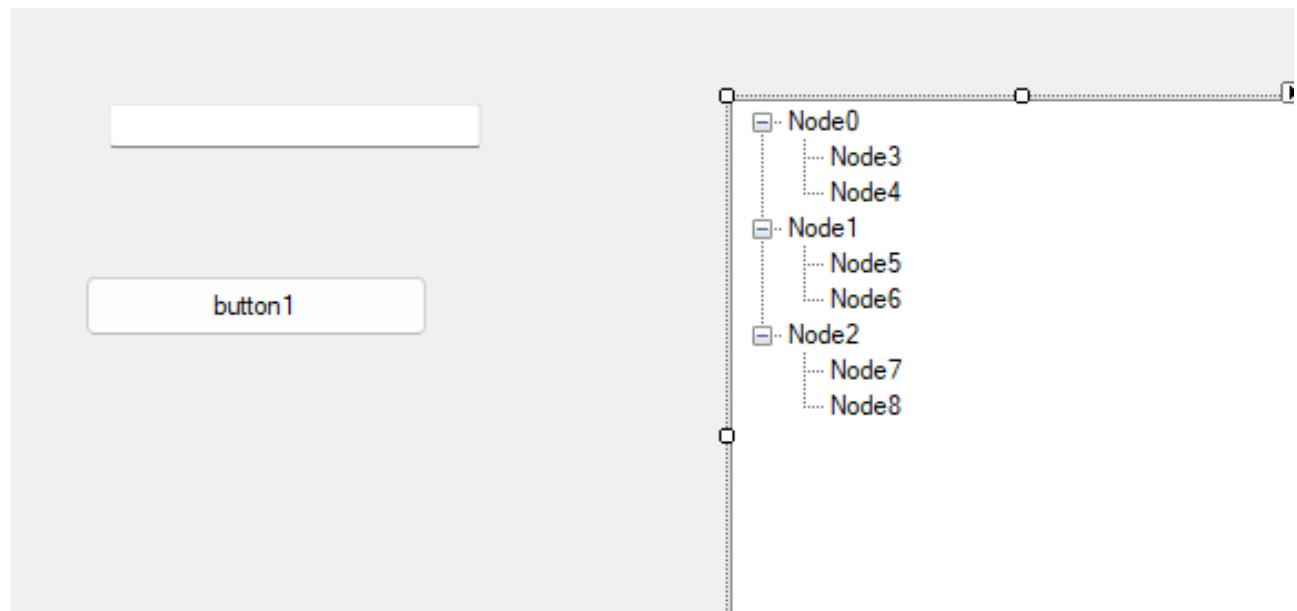
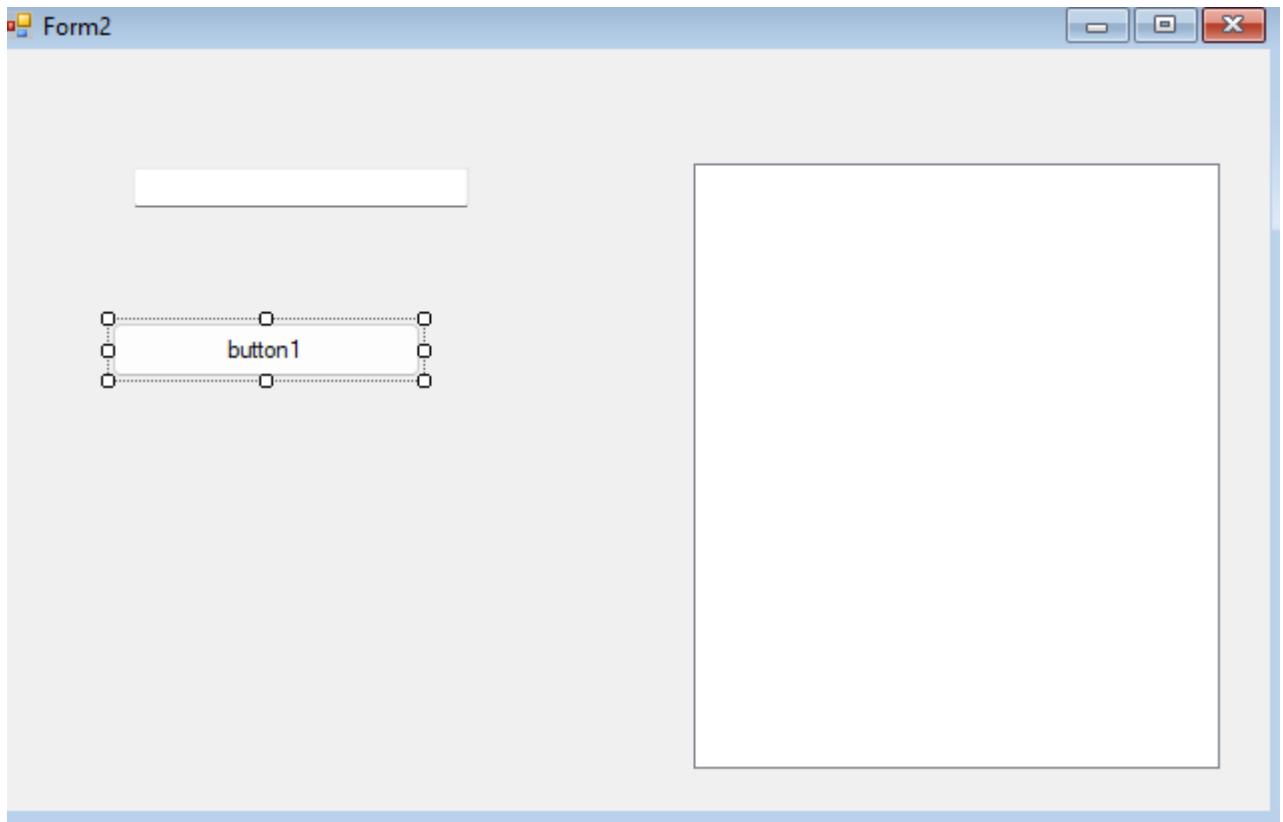


```
private void toolStripMenuItem2_Click(object sender, EventArgs e)
{
    Form frm2 = new Form2();

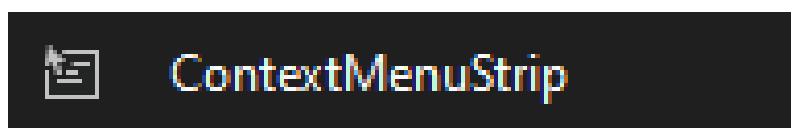
    frm2.MdiParent = this;

    frm2.Show();
}
```

تعالي نحط فيه tree و text box و button

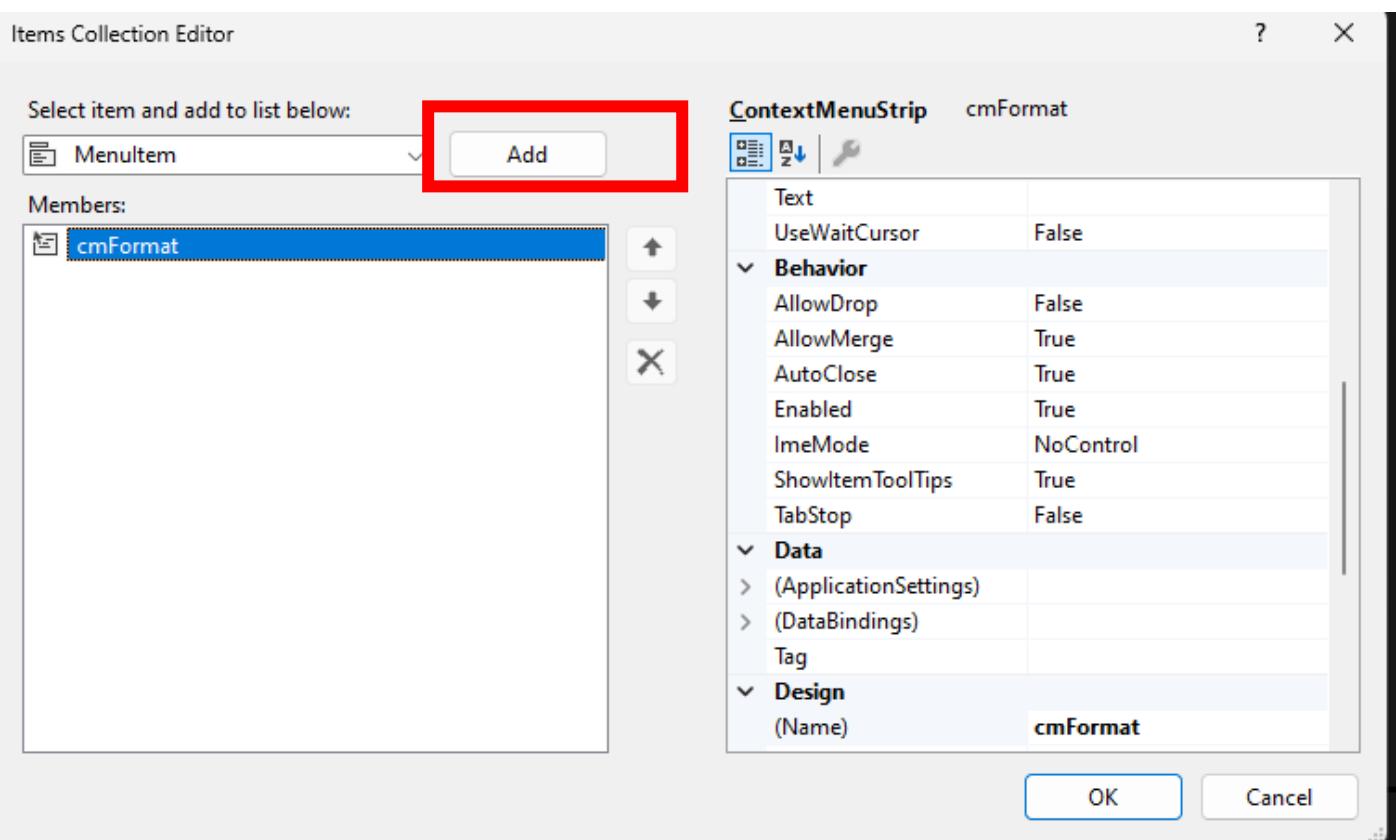
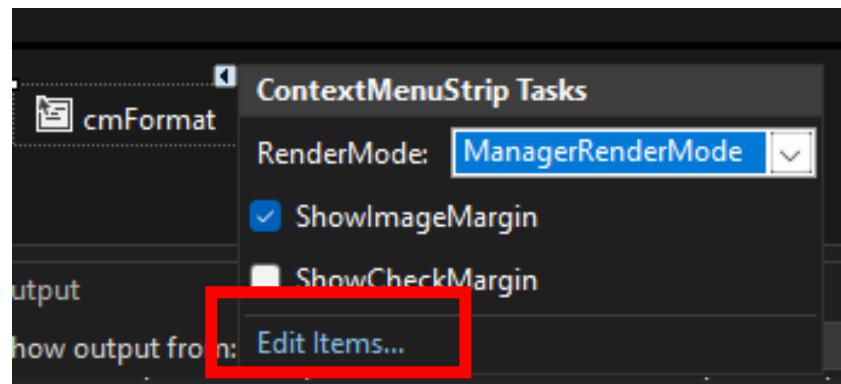


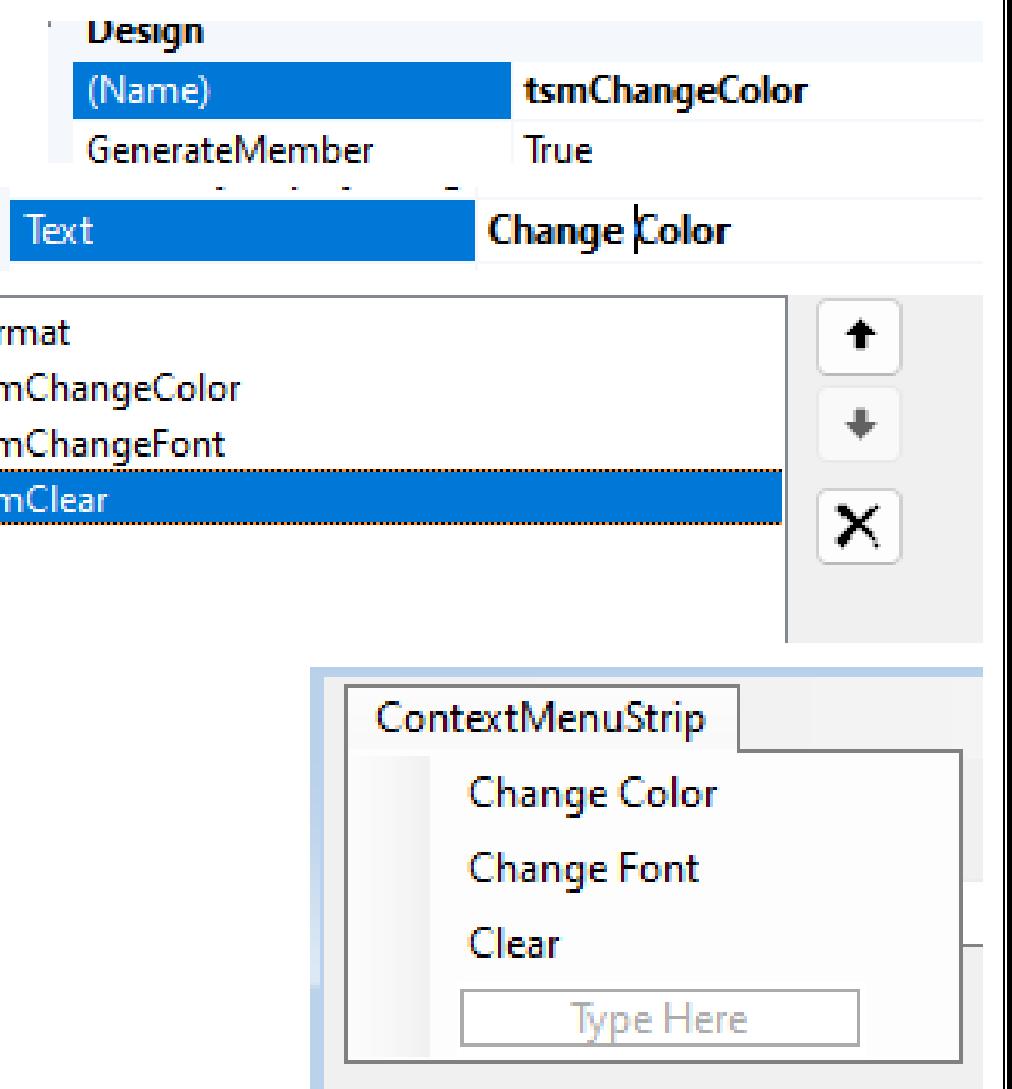
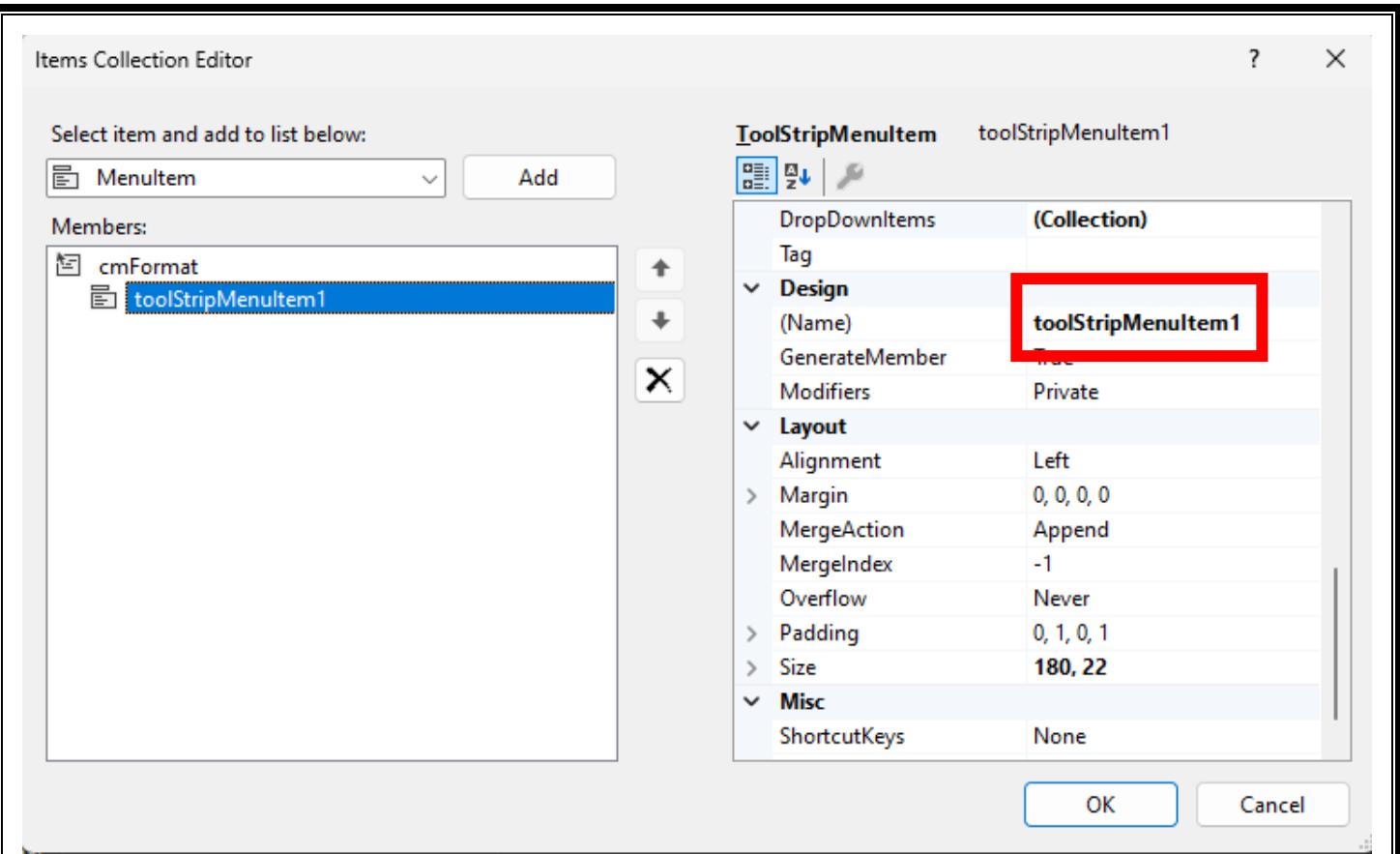
بنضيف من هنا context menu





طيب عايزيين نضيف فيها عناصر





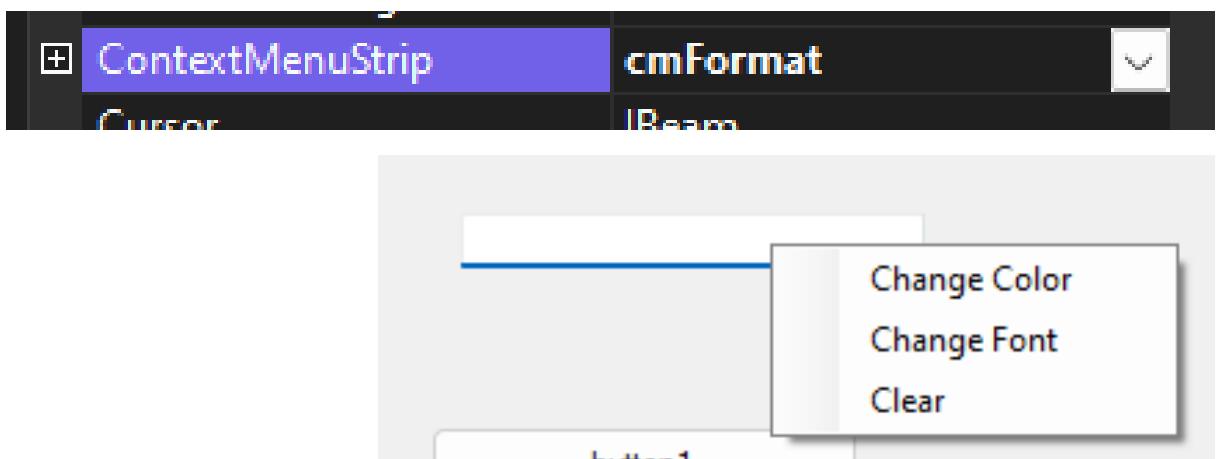
دبل كليك على أي عنصر بيوديك عالكوند بتاعها

```
private void tsmChangeColor_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK)
        textBox1.BackColor = colorDialog1.Color;
}

private void tsmSetFont_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
    {
        textBox1.Font = fontDialog1.Font;
    }
}

private void tsmClear_Click(object sender, EventArgs e)
{
    textBox1.Text = "";
}
```

عشان نربط القايمه بعنصر معين زي ال text box بنروح على اعدادات العنصر ده ونختار ال من هنا menu



لو جيت عمل text box تاني وربطت نفس القايمه دي بييه وجيت غير اللون مثلا اللي هيتغير بس هو اول text box

فانت عندك حل من اتنين يالما تخلی الكود يغير اللون بتاع الاتنين text box مره واحده يالما تعمله قايمه خاصه بييه

### **End Of Course Message**

هنا انت محتاج ممارسه بقى وتحاول تجمع في ال controls زي البازل  
وان الفرق بين كل واحد والثانوي هو الممارسه

**The End**