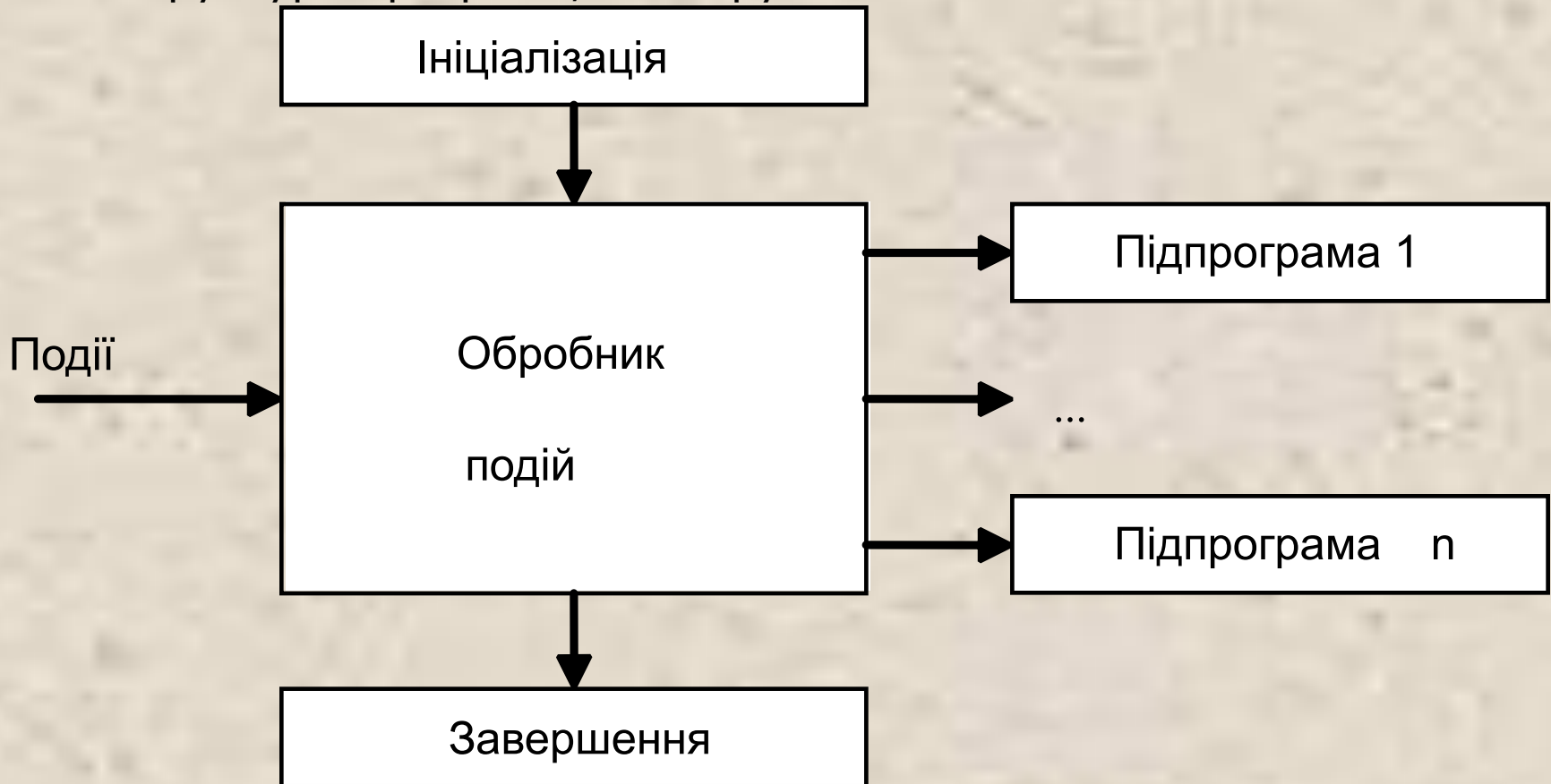

Основні особливості Windows

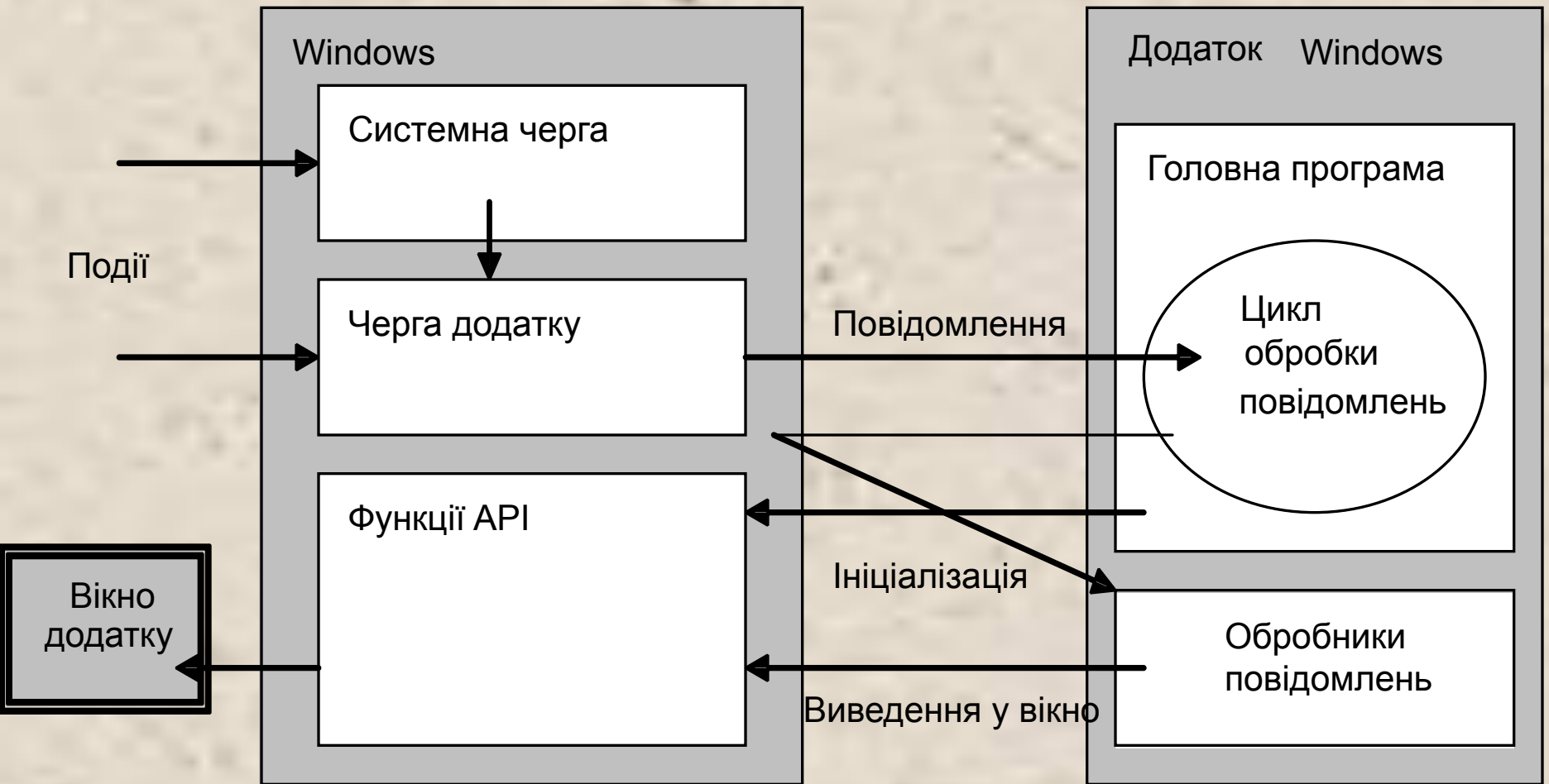
- Багатозадачність
- Незалежність програм від апаратури
- Стандартний графічний інтерфейс з користувачем
- Підтримка віртуального адресного простору для кожного додатка.
- Можливість обміну даними між додатками
- Можливість запуску старих програм

Подійно-кероване програмування

- Структура програми, що керується подіями:

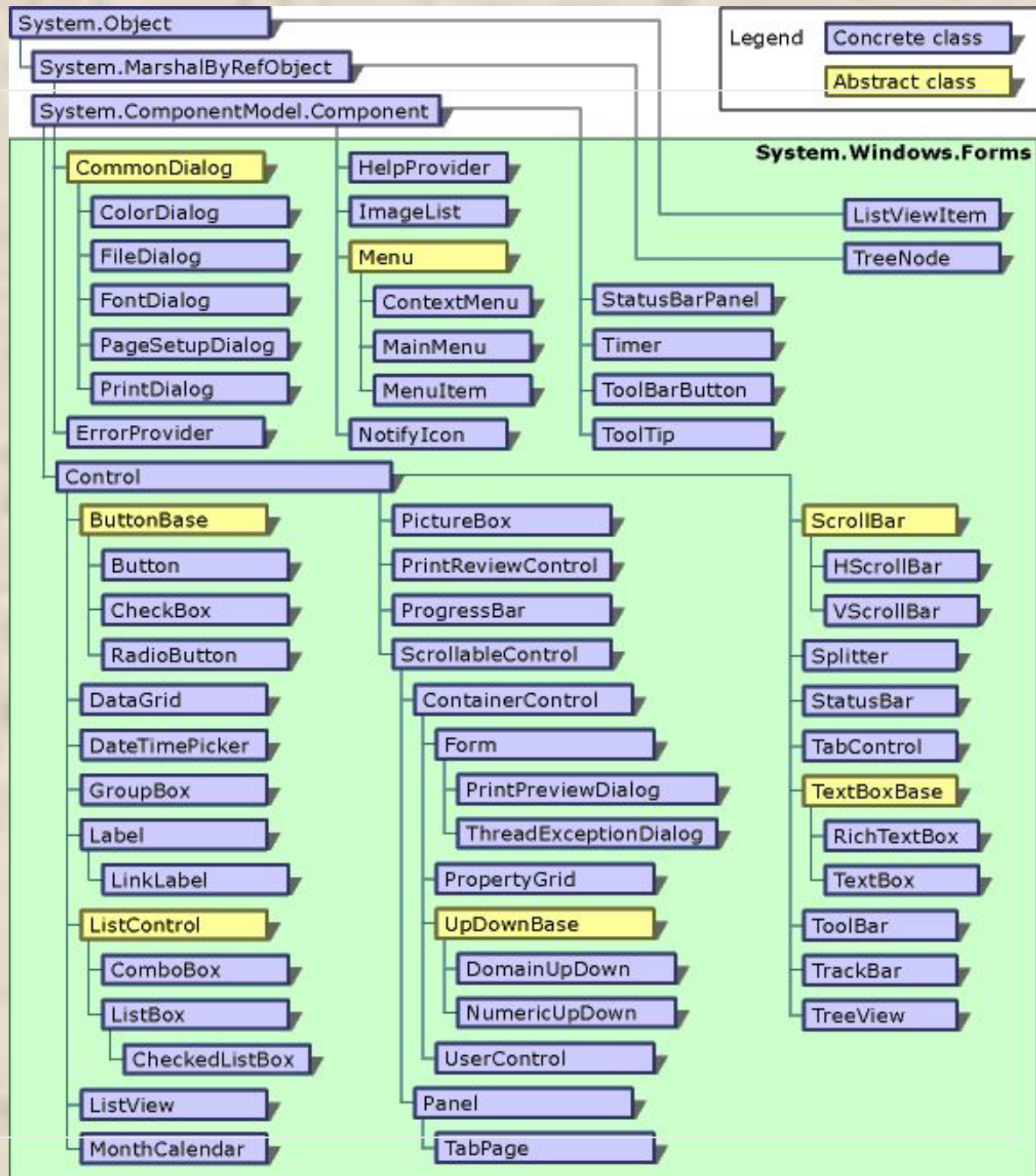


Структура Windows- додатку



-
- **Середовище** Visual Studio.NET містить зручні **засоби розробки** Windows-додатків, наприклад:
 - створення шаблонів додатків і форм;
 - створення заготовок обробників подій.
 - **Бібліотека** класів .NET включає простір імен System.Windows.Forms, що містить величезну кількість типів - **будівельних блоків** Windows-додатків, наприклад:
 - Application
 - Button, CheckBox, DataGrid, GroupBox, ListBox, PictureBox
 - Form
 - ColorDialog, FileDialog, FontDialog
 - Menu, MainMenu, MenuItem
 - Clipboard, Help, Timer, Screen, Cursors
 - StatusBar, ToolBar, ScrollBar
-

Елементи управління Windows.Forms



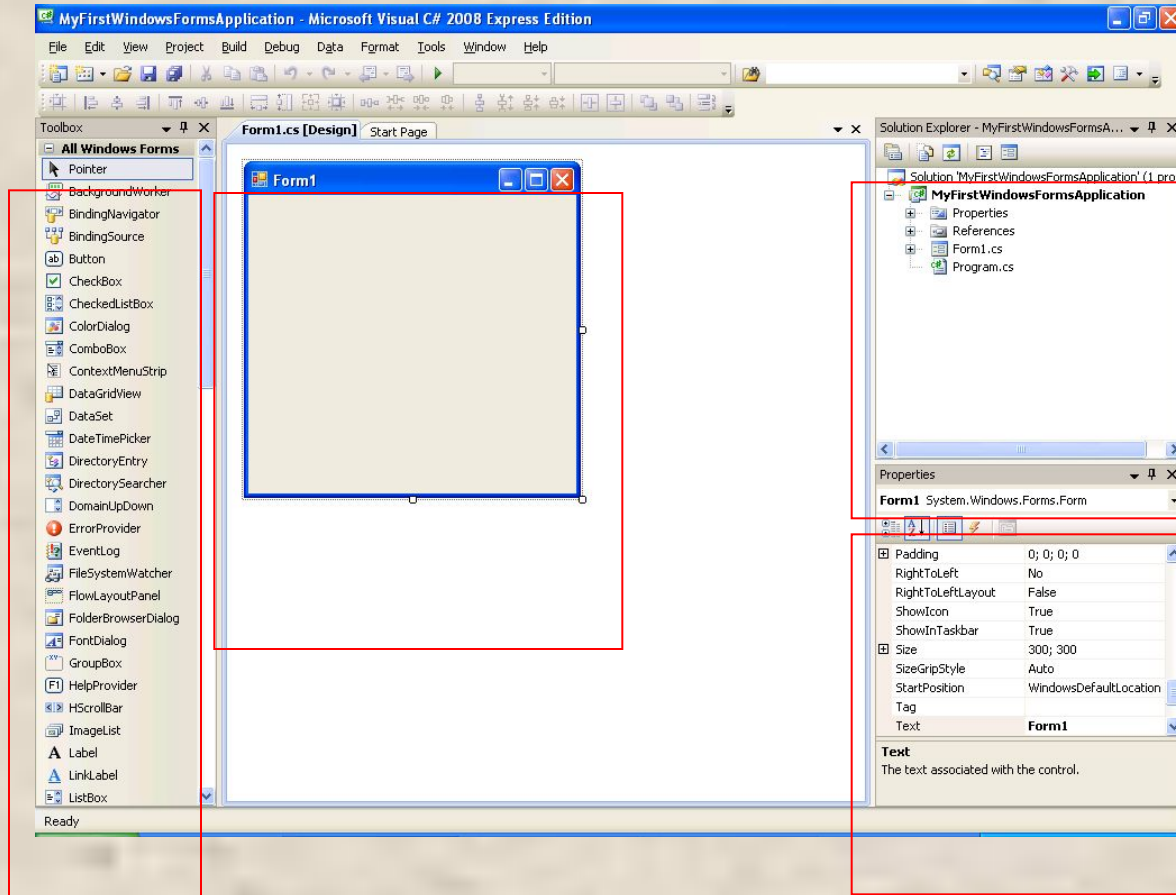
-
- Процес створення Windows-додатку складається із двох основних етапів:
 - *візуальне проектування*, тобто задання зовнішнього вигляду додатку
 - *визначення поведінки* додатку шляхом написання процедур обробки подій.
 - *Візуальне проектування* заключається в розміщенні на формі компонентів (елементів управління) і заданні їх властивостей і властивостей самої форми.
-

Огляд.

- Створення форми Form
 - Робота з об'єктами керування Controls
 - Створення SDI та MDI Applications
 - Робота з подіями
-

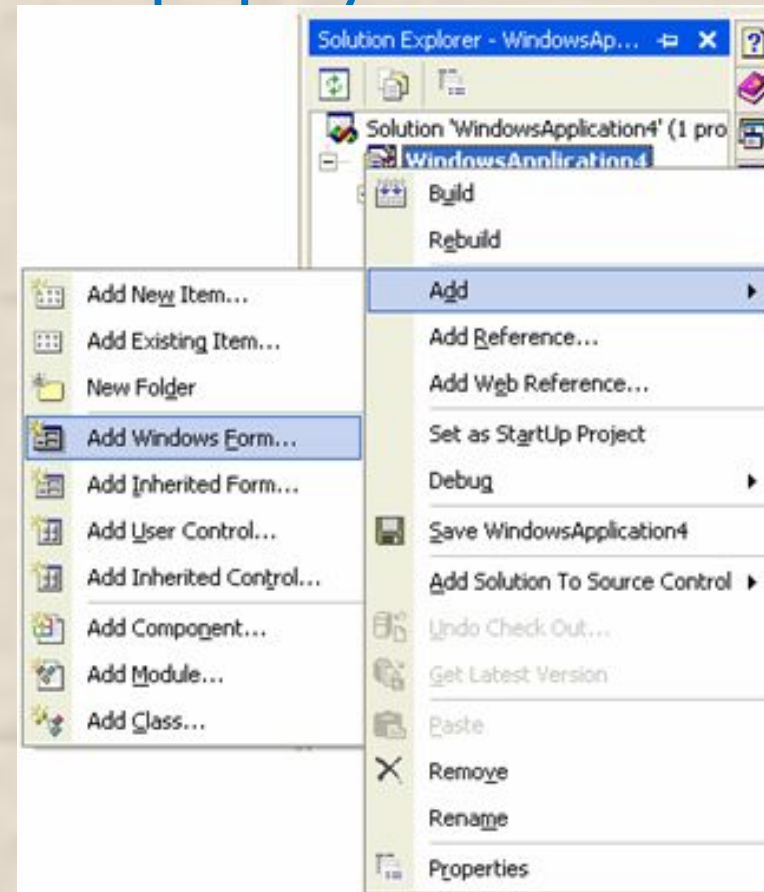
Після створення Windows-додатку...

Середовище візуальної розробки



Як створити Windows-форму?

- Під час створення проекту створюється **базова форма**
- Додати форму до проекту:
 - 1) правий клік у вікні **Solution Explorer**
 - 2) натиснути **Add**
 - 3) обрати **Add Windows Form**



Як встановити властивості Windows-форми?

The image shows a screenshot of the Visual Studio Properties window for a Windows Form named 'Form1'. The window is titled 'Properties' and shows the following properties:

Form1 System.Windows.Forms.Form	
[Expand/Collapse] [Sort] [Filter] [Refresh] [Reset]	
Padding	0; 0; 0; 0
RightToLeft	No
RightToLeftLayout	False
ShowIcon	True
ShowInTaskbar	True
Size	300; 300
SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	Form1

Below the properties list is the 'Text' property description: 'The text associated with the control.'

Annotations in Russian point to various parts of the window:

- Имя формы** (Form name) points to the 'Form1' text in the title bar.
- Открыть категории** (Open categories) points to the expand/collapse icon.
- Сортировать по алфавиту** (Sort by alphabet) points to the sort icon.
- Кнопка событий** (Event button) points to the lightning bolt icon.
- Панель описания** (Description panel) points to the 'Text' property description.

Визначення поведінки програми

- Визначення поведінки програми починається з прийняття рішення, які дії повинні виконуватися при клацанні кнопки, введенні тексту, виборі пунктів меню і т. д. (по яким подіям будуть виконуватися дії, що реалізують функціональність програми).
- Заготовка шаблону обробника події формується подвійним клацанням на полі, розміщеному зправа від імені відповідної події на вкладці Events вікна властивостей, при цьому з'являється вкладка вікна редактора коду з заготовкою відповідного обробника.
- Для кожного класу визначено свій набір подій, на які він може відреагувати.

Події, які виникають найчастіше

- Activated — отримання формою фокуса введення;
- Click, DoubleClick — одинарне і подвійне клацання мишею;
- Closed — закриття форми;
- Load — завантаження форми;
- KeyDown, KeyUp — натиснення і відпускання будь-якої клавіші і їх поєднань;
- KeyPress — натиснення клавіші, що має ASCII-код;
- MouseDown, MouseUp — натиснення і відпускання кнопки миші;
- MouseMove — переміщення миші;
- Paint — виникає при необхідності перемалювання форми.

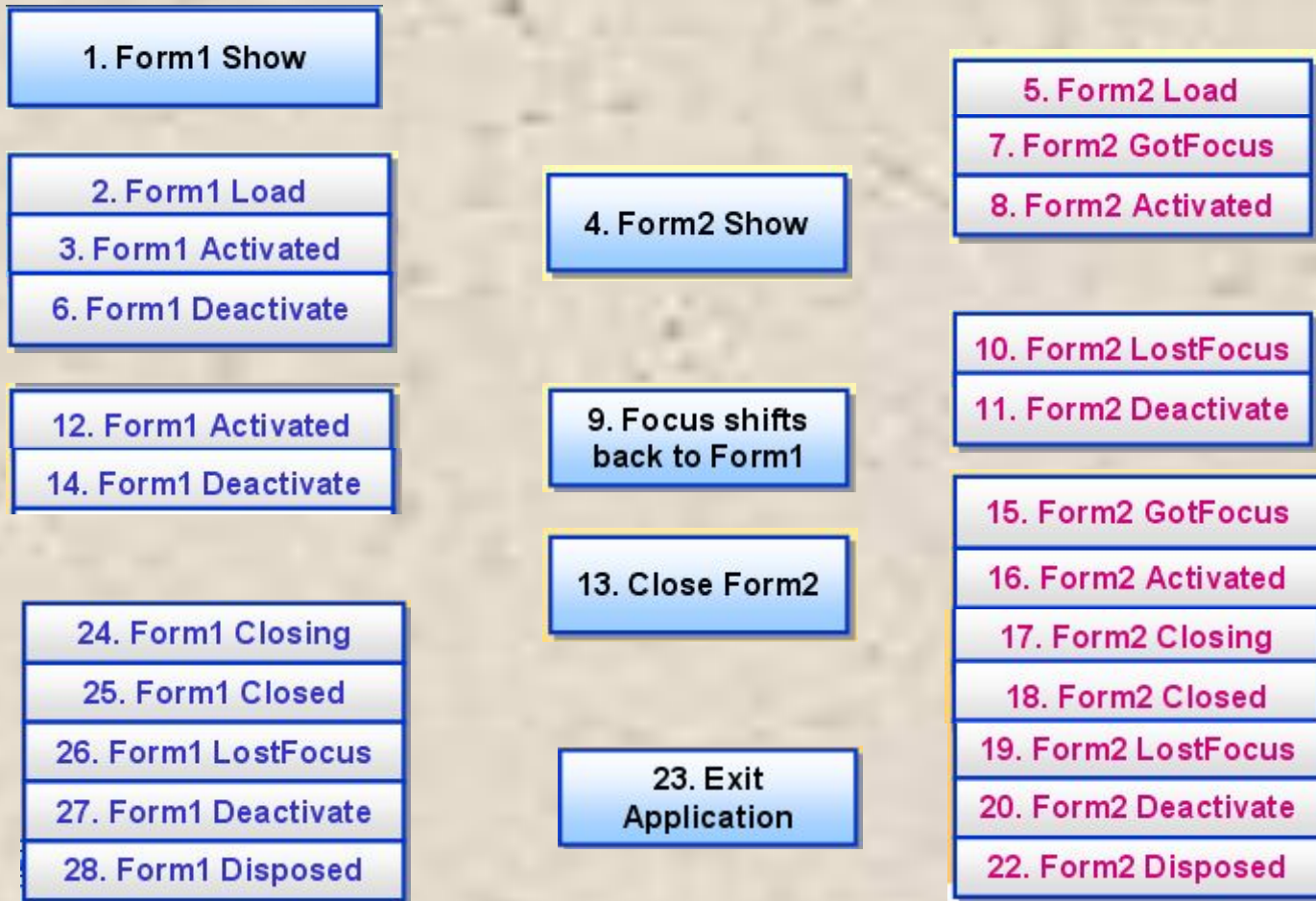
Види вікон

- **Модальне вікно** не дозволяє користувачу переключатися на інші вікна того ж додатку, поки не буде завершена робота з поточним вікном.
- Модальними зазвичай роблять **діалогові вікна**, що вимагають від користувача введення будь-якої інформації.
- **Немодальне вікно** дозволяє переключатися на інші вікна того ж додатку. Немодальні вікна зазвичай є інформаційними. Вони використовуються в тих випадках, коли користувачу бажано надати свободу вибору — залишати на екрані яку-небудь інформацію чи ні.
- Кожен додаток містить одне **головне вікно**. Клас головного вікна додатку містить точку входу в додаток (статичний метод Main). При закритті головного вікна додаток завершується.
- Вид вікна визначає його функціональність, наприклад, вікно з одинарною рамкою не може змінювати свої розміри.

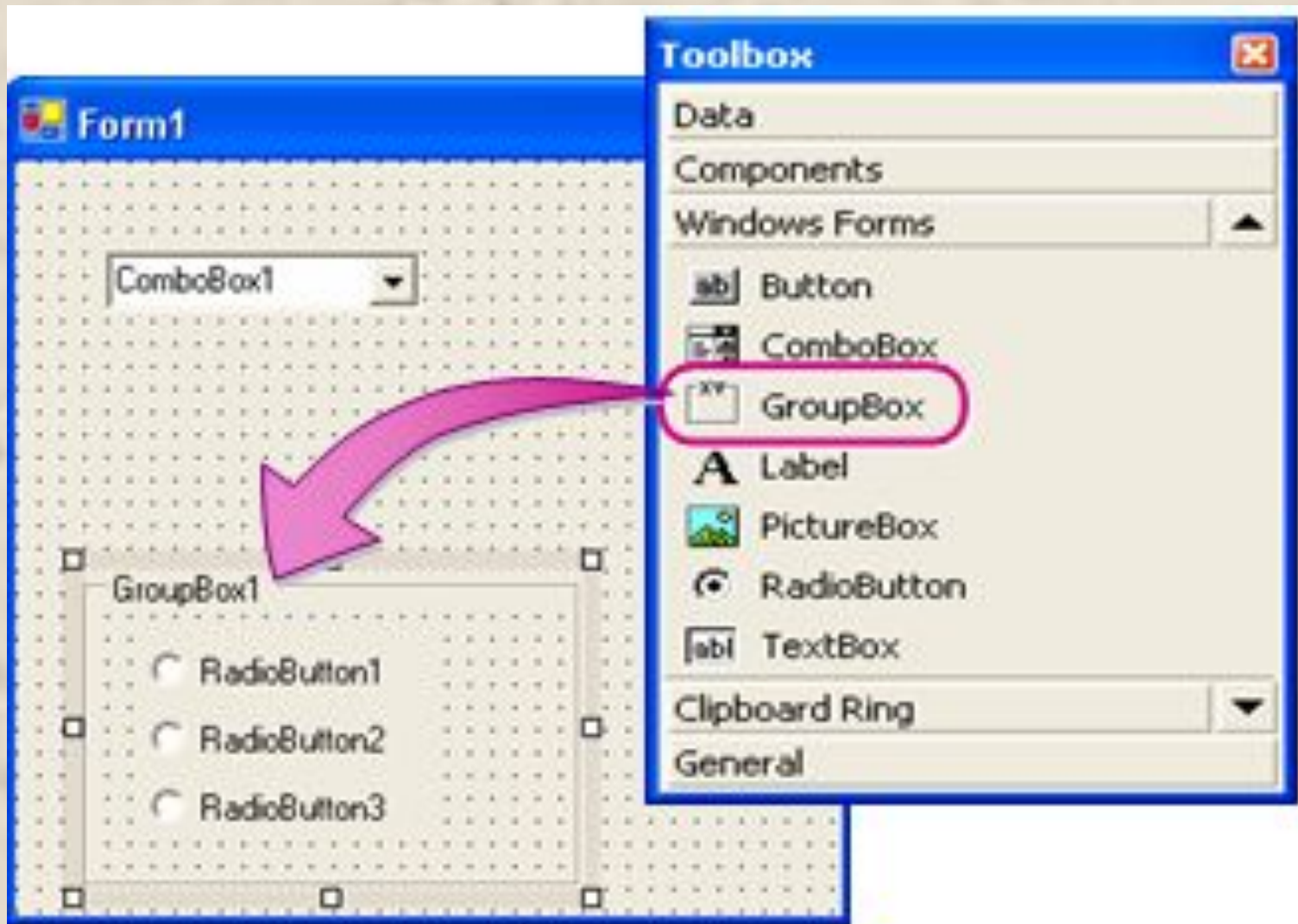
Як розуміти “цикл існування” об’єкта Form?

- «Життя» форми починається з виклика її назви методом **Application.Run()**;
- Після розміщення об’єкта в динамічній пам’яті генерується подія **Load()**;
- Коли форма отримує фокус введення – генерується подія **Activated()**;
- Після втрати фокуса введення – **Deactivate()**;
- Після закриття форми по черзі генеруються:
 - 1) подія **FormClosing()** безпосередньо перед закриттям;
 - 2) подія **FormClosed()** – після закриття.

Життєвий цикл" форми:

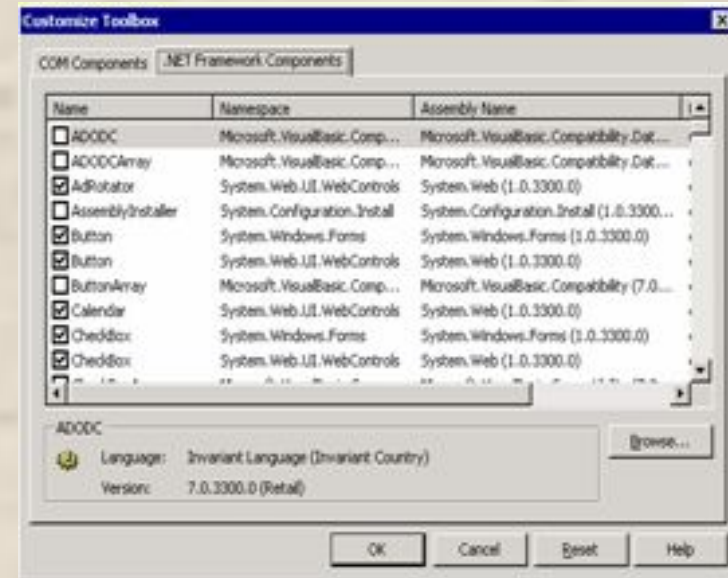


Як додати елементи керування на форму?



Як змінювати Controls Toolbox?

- Правий-клік на **Toolbox**
- Натиснути **Choose Items**
- Обрати необхідний Control на сторінці **.Net Framework Components**



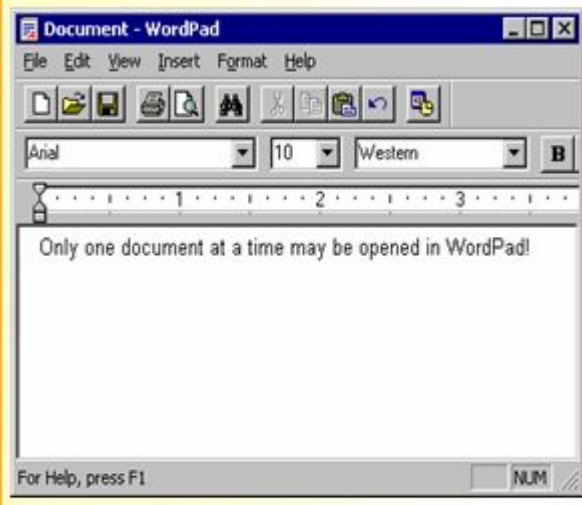
SDI- и MDI- додатки.

Існує три типи додатків:

- Що базуються на діалоговому вікні.
- Однодокументні (**Single Document Interface, SDI**).
- Багатодокументні (**Multi-Document Interface, MDI**).

SDI- и MDI- добавки.

SDI

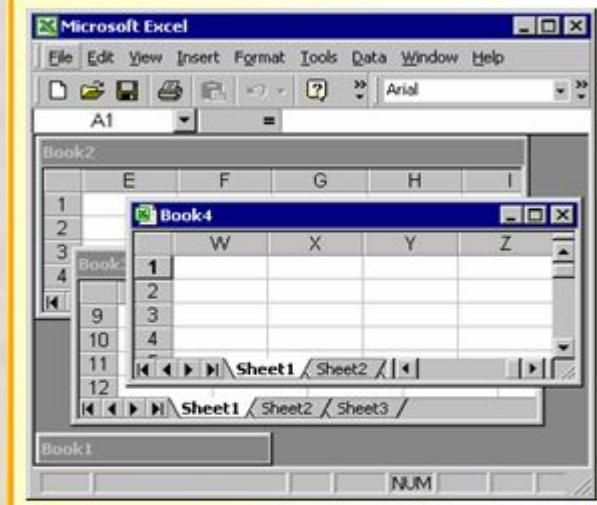


The screenshot shows a single window titled "Document - WordPad". The menu bar includes File, Edit, View, Insert, Format, and Help. The toolbar contains icons for file operations and editing. The font settings are set to Arial, size 10, and Western font. The main text area contains the message: "Only one document at a time may be opened in WordPad!". The status bar at the bottom indicates "For Help, press F1" and "NUM".

Only one document is visible

You must close one document before you open another

MDI



The screenshot shows a Microsoft Excel window titled "Microsoft Excel" with a menu bar including File, Edit, View, Insert, Format, Tools, Data, Window, and Help. The toolbar includes icons for file operations and editing. The font settings are set to Arial. The main area displays a spreadsheet with columns E through I and rows 1 through 12. A secondary window titled "Book4" is overlaid on the main window, showing a spreadsheet with columns W through Z and rows 1 through 4. The status bar at the bottom indicates "NUM".

Displays multiple documents at the same time

Each document is displayed in its own window

Стандартний **Delegate** загальної бібліотеки.

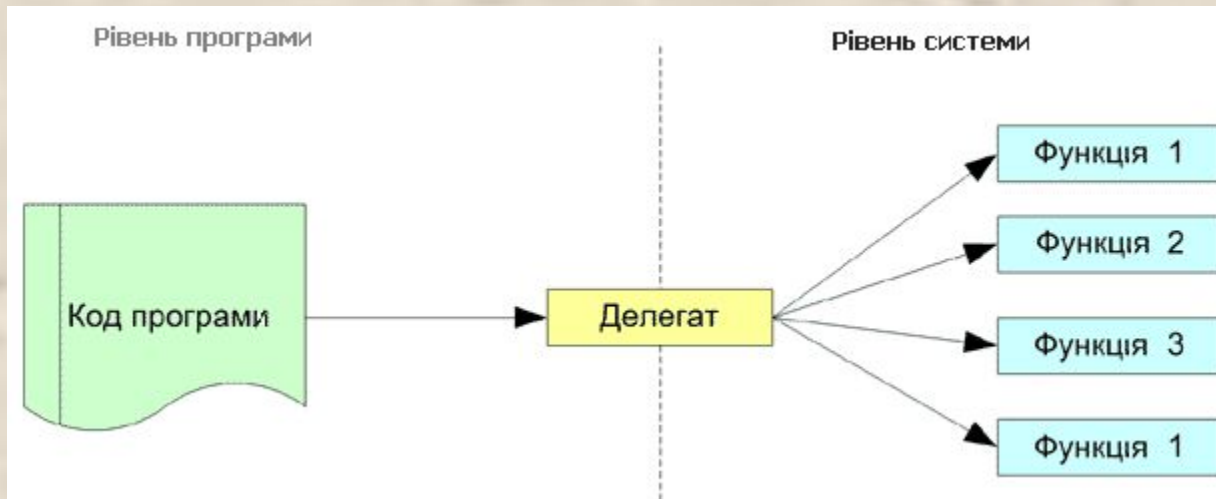
Стандартний делегат та події.

1. Опис поняття
2. Створення делегата
3. Використання делегата

Створення та використання делегатів. Опис поняття.

Делегат – це об'єкт, що використовується для виклику деякого методу. Головною властивістю делегатів є спроможність вказати метод не в період компіляції, а в процесі виконання програми.

Делегат може посилатися на декілька методів або функцій.



- Делегат використовується як засіб формування подій при отриманні повідомлень від додатка. Всі дії реалізуються через обробники подій.

Створення та використання стандартних делегатів. Постановка задачі.

- Друзі відправилися на рибалку. На березі вони розставили вудки, конструкції яких відрізнялись одна від одної, але інструкції по використанню були присутні. Самі сіли відпочивати, а **делегатом** був назначений Дядя Федір. Іншими словами:
 - **ДЕЛЕГАТ** – це представник, який має визначені повноваження (в нашому випадку все, що описує процес рибалки, а саме “обробити” рибу, що клює, насадити черв’яка, тощо). На відміну від повноважень (дій) делегата наперед не відомі події, що ініціює дію (клює чи зірвалась риба). А як саме реагувати під час можливих подій делегату стане відомо з інструкцій, що були наперед визначені.
-

Створення та використання стандартних делегатів. Опис поняття.

Делегат забезпечує реакцію на подію:

```
public delegate void EventHandler(object sender, EventArgs e);
```

В цьому процесі ще присутні інші учасники:

- 1) подія, що трапилась з вудкою (`object sender`);
 - 2) інструкція щодо дій (`EventArgs e`).
- Ініціатор дії – завжди **sender (об'єкт-відправник – джерело події)**.
 - Другий учасник (інструкція щодо дій по використанню вудки, на яку впливає **sender**) – наперед визначений клас `EventArgs`, в якому зберігається інформація про подію.