

### Креирање апликације

Све апликације које се користе у интегрисаним, графичким и ситемским окружењима су стандардизоване.

Апликације су скуп стандардних форми са којих се покрећу разни догађаји изазвани акцијама корисника или стањем система.

Стандардне форме, као и контроле које се постављају на њих, садржане су у системским библиотекама класа и доступне су свим програмерима који креирају апликације за одређени оперативни систем.

Осим системских, програмери могу креирати и своје библиотеке класа које касније користе у склопу различитих апликација.

Библиотеку класа у програмском језику C# зовемо именским простором (namespace).

При креирању апликације C# аутоматски генерише нов именски простор.

Ако користимо неке класе које припадају другим именским просторима, неопходно је називе именских простора навести коришћењем директиве using (usin imenskiProstor).

Системски именски простор који садржи све основне класе које користимо у апликацијама је System.Windows.Forms.

Његове основне класе су Control и Form.

Класа коју програмер дефинише при креирању апликације наслеђена је од системске класе Form, што значи да садржи сва њена својства и методе.

Програмер даље надограђује и мења класу.

Графички, она представља прозор у коме се извршава апликација.

На њу се могу додавати системске контроле које су наслеђене од класе Control.

### Својства класе

Најчешће се користе својства класе Form:

- Name – име инстанце класе форме
- Text – текст исписан у насловној линији форме
- Left, Top – координате положаја левог горњег угла форме у активној резолуцији екрана
- BackColor – боја позадине форме
- Width, Height – ширина и висина форме
- Font – врста слова коришћена за Text
- FormBorderStyle – дефинише изглед оквира (ако је подешено на Sizable, могућа је промена величине форме, док FixedDialog онемогућава промену величине форме)
- MaximizeBox – ако је true могуће је форму максимизирати помоћу дугмета на насловној линији, у супротном није могуће
- MinimizeBox – ако је true могуће је форму минимизирати помоћу дугмета на насловној линији, у супротном није могуће
- MiximumSize – максимална дозвољена величина форме, ако је подешена на 0, 0 нема ограничења максималне величине
- MinimumSize – минимална дозвољена величина форме, ако је подешена на 0, 0 нема ограничења минималне величине

### Програмирани догађаји

У току извршавања апликације корисник изводи различите акције на форми.

Неке од тих акција могу изазвати реакцију апликације ако је програмер испрограмирао одговарајући догађај.

Такође, програмер може испрограмирати реакцију апликације на одређена стања система или на тренутно стање апликације.

Сваки догађај је потпуно одређен оним који га је изазвао (sender) као и одређеним параметрима који га описују (EventArgs).

Догађаји који се начешће програмирају:

- Load – реакција на учитавање форме
- Paint – реакција на појављивање форме
- MouseClick – реакција на клик дугметом миша
- MouseDown – реакција на притисак на дугме миша
- MouseUp – реакција на отпуштање дугмета миша
- MouseMove – реакција на померање миша
- KeyDown – реакција на притисак тастера тастатуре
- KeyPress – реакција на откуцан тастер тастатуре
- KeyUp – реакција на отпуштање тастера тастатуре
- Resize – реакција на промену величине форме

Програмер проширује форму додавањем нових, претходно дефинисаних компоненти (системских и кодираних).

Затим, правилним избором догађаја које кодира усмерава ток апликације.

У класи, програмер може дефинисати потпуно нове методе и предефинисати већ постојеће.

Да би апликација била уопште и покренута, мора постојати метод из којег се може генерисати полазни објекат (форма) како би затим корисник, различитим акцијама преузео контролу над њеним извршавањем.

Тај метод је Main() и мора бити static јер мора бити покренут пре креирања било ког објекта и нема повратну вредност.

Main() обавезно у себи садржи команду покретања апликације креирањем објекта стартне класе: Application.Run(new Form1());

У изради апликације постоје две етапе: дизајн и кодирање, и оне се најчешће међусобно преплићу.

### Етапа дизајна

У оквиру ове етапе, на форми се постављају претходно дефинисане компоненте (најчешће контроле).

Контроле су објекти помоћу којих корисник комуницира са апликацијом.

Већина контрола је изведена из класе System.Windows.Forms.Control.

Најчешће коришћене контроле су Button, RadioButton, CheckBox, TextBox, RichTextBox, Label, ListBox, ComboBox, GroupBox, PictureBox.

Нека својства ових контрола су заједничка за све контроле: Name, Top, Left, Width...

Етапа кодирања

У овој етапи програмирамо методе који представљају реакцију апликације на одређени догађај изазван акцијом корисника или стањем система.

Пример: Креирати апликацију у којој клик на радну површину форме изазива појаву поруке на конзоли.

```
using System;
using System.Windows.Forms;
namespace ProjekatCS004
{
    public partial class Program : Form
    {
        public Program()
        {
            InitializeComponent();
        }
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Program());
        }
        private void InitializeComponent()
        {
            this.SuspendLayout();
            this.ClientSize = new System.Drawing.Size(500, 200);
            this.Name = "Program";
            this.MouseClick += new
System.Windows.Forms.MouseEventHandler(this.M_Click);
            this.ResumeLayout(false);
        }
        private void M_Click(object sender, EventArgs e)
        {
            Console.WriteLine("Kliknuo misem");
        }
    }
}
```

Овде се користи посебна метода М\_Click која користи основну класу Object као и класу EventArgs која ради са догађајима.

Метода се позива унутар методе за иницијализацију компоненти преко класе MouseEventHandler која обрађује све догађаје клика мишем на које класа мора да реагује.

Реакција на догађај MouseClick је појава истог стринга "Kliknuo misem" на конзоли.

Пример: Креирати апликацију где се од клика мишем на површини апликације добија пар целобројних вредности координата места клика који се исписује на конзоли.

```
using System;
using System.Windows.Forms;
namespace ProjekatCS004
{
    public partial class Program : Form
    {
        public Program()
        {
            InitializeComponent();
        }
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Program());
        }
        private void InitializeComponent()
        {
            this.SuspendLayout();
            //
            // Program
            //
            this.ClientSize = new System.Drawing.Size(500, 200);
            this.Name = "Program";
            this.MouseClick += new
System.Windows.Forms.MouseEventHandler(this.M_Click);
            this.ResumeLayout(false);
        }
        private void M_Click(object sender, MouseEventArgs e)
        {
            Console.WriteLine($"Kliknuta je tacka ({e.X}, {e.Y})");
        }
    }
}
```

#### Задаци за самосталан рад

1. Креирати апликацију у којој се реагује на клик мишем тако да генерише случајан број којим се поставља вредност позадинске боје на форми а те вредности се исписују на конзоли.
2. Креирати апликацију у којој се реагује на клик мишем тако да генерише два случајна броја којима се поставља промена димензија форме. На конзоли исписати добијену димензију форме.

3. Креирати апликацију у којој се реагује на клик мишем тако што се на линији наслова форме исписују координате тачке са површине форме на коју је кликнуто мишем.