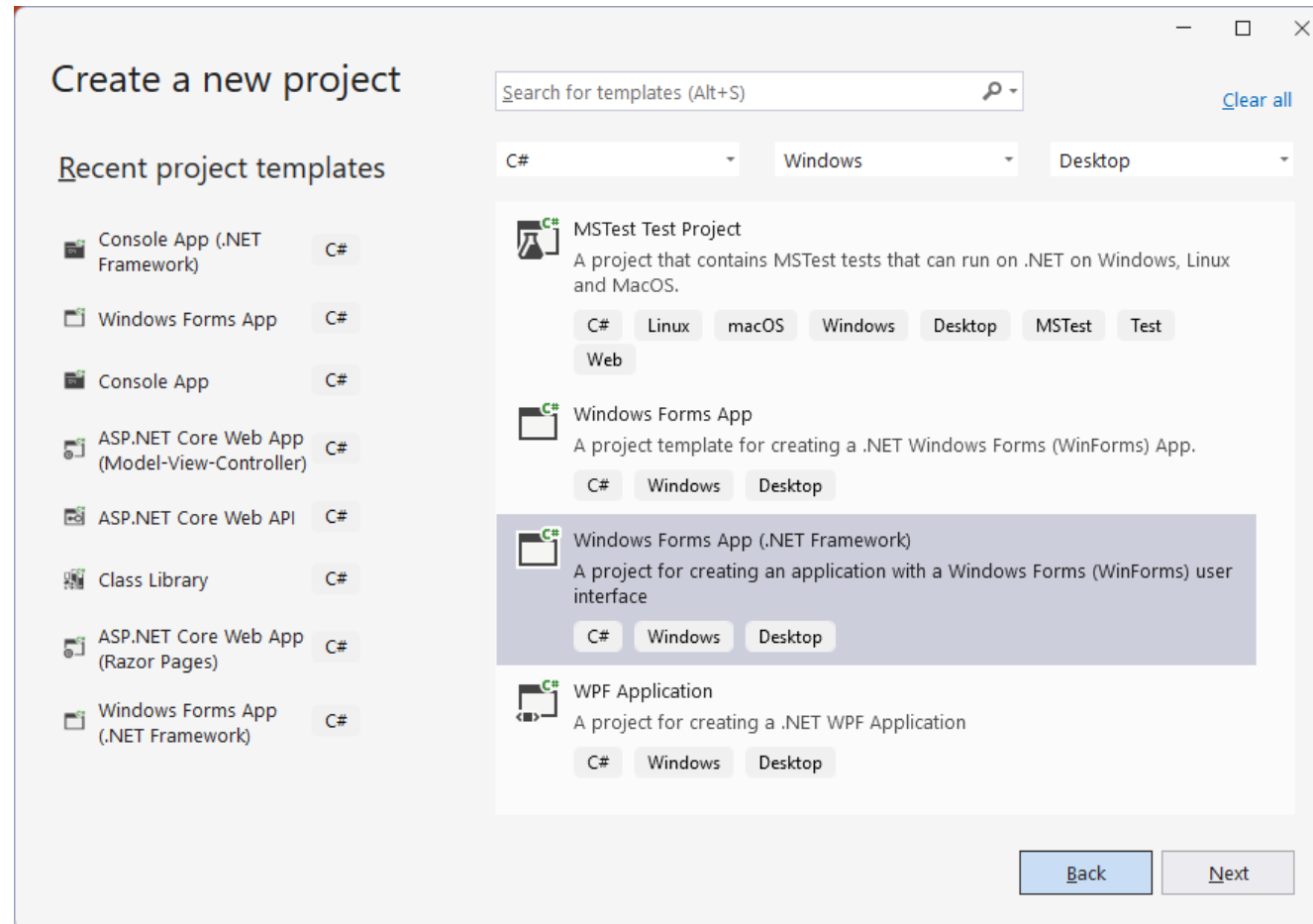
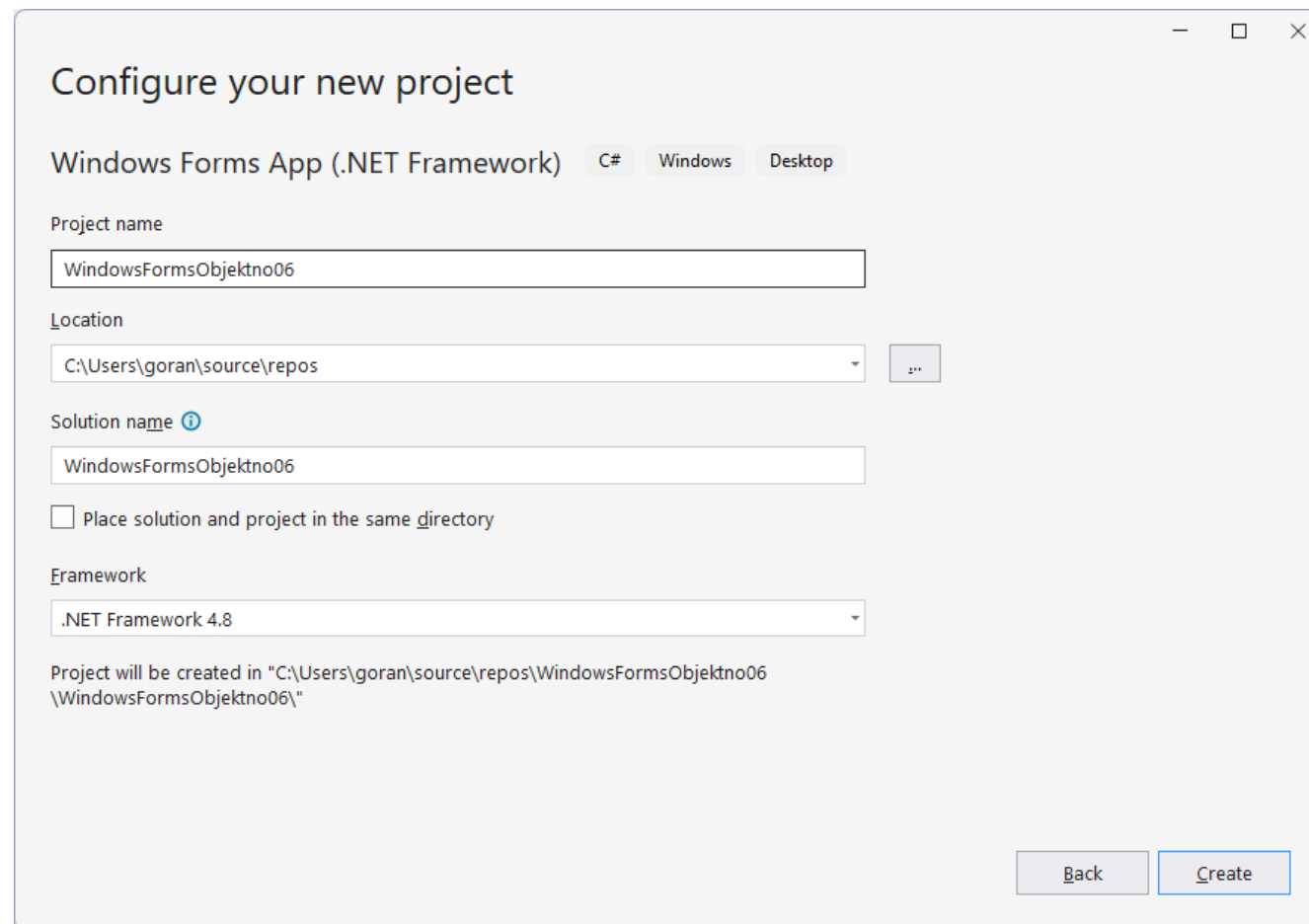


Windows forme

Kreiranje Windows Forms aplikacije



Kreiranje Windows Forms aplikacije



Configure your new project

Windows Forms App (.NET Framework) C# Windows Desktop

Project name

WindowsFormsObjektno06

Location

C:\Users\goran\source\repos

Solution name ⓘ

WindowsFormsObjektno06

☐ Place solution and project in the same directory

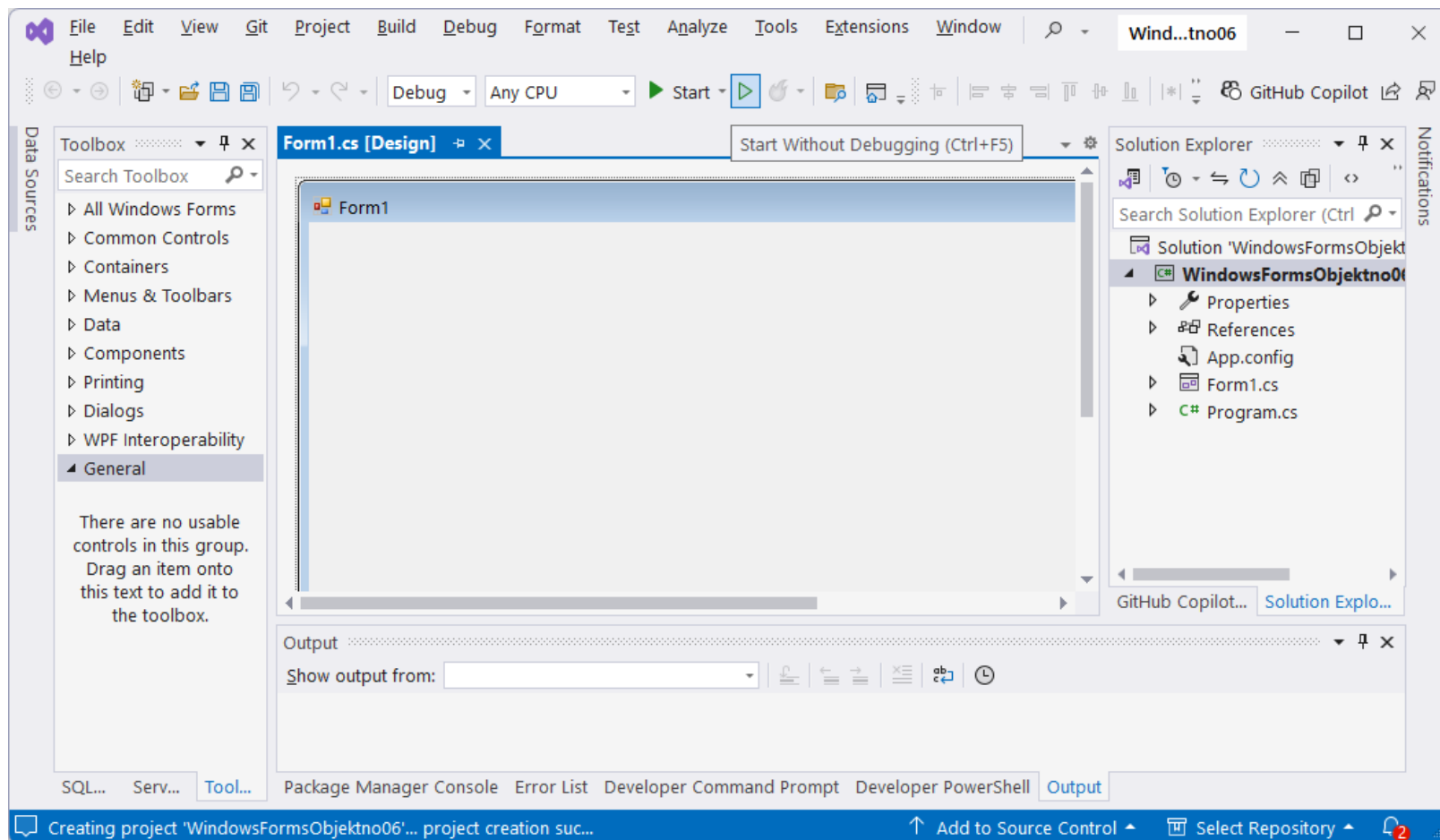
Framework

.NET Framework 4.8

Project will be created in "C:\Users\goran\source\repos\WindowsFormsObjektno06\WindowsFormsObjektno06\"

Back Create

Dizajner Windows Forms aplikacije



Pojam windows forme

- Forma je osnovni element korisničkog interfejsa
- Forma je kontejnerska kontrola izvedena iz klase **Form** koja se nalazi u biblioteci System.Windows.Forms
- Windows forme se koriste pri izradi desktop aplikacija, naročito kada se očekuje da se obrada podataka izvršava na klijentskom računaru

```
public partial class Form1 : Form
```

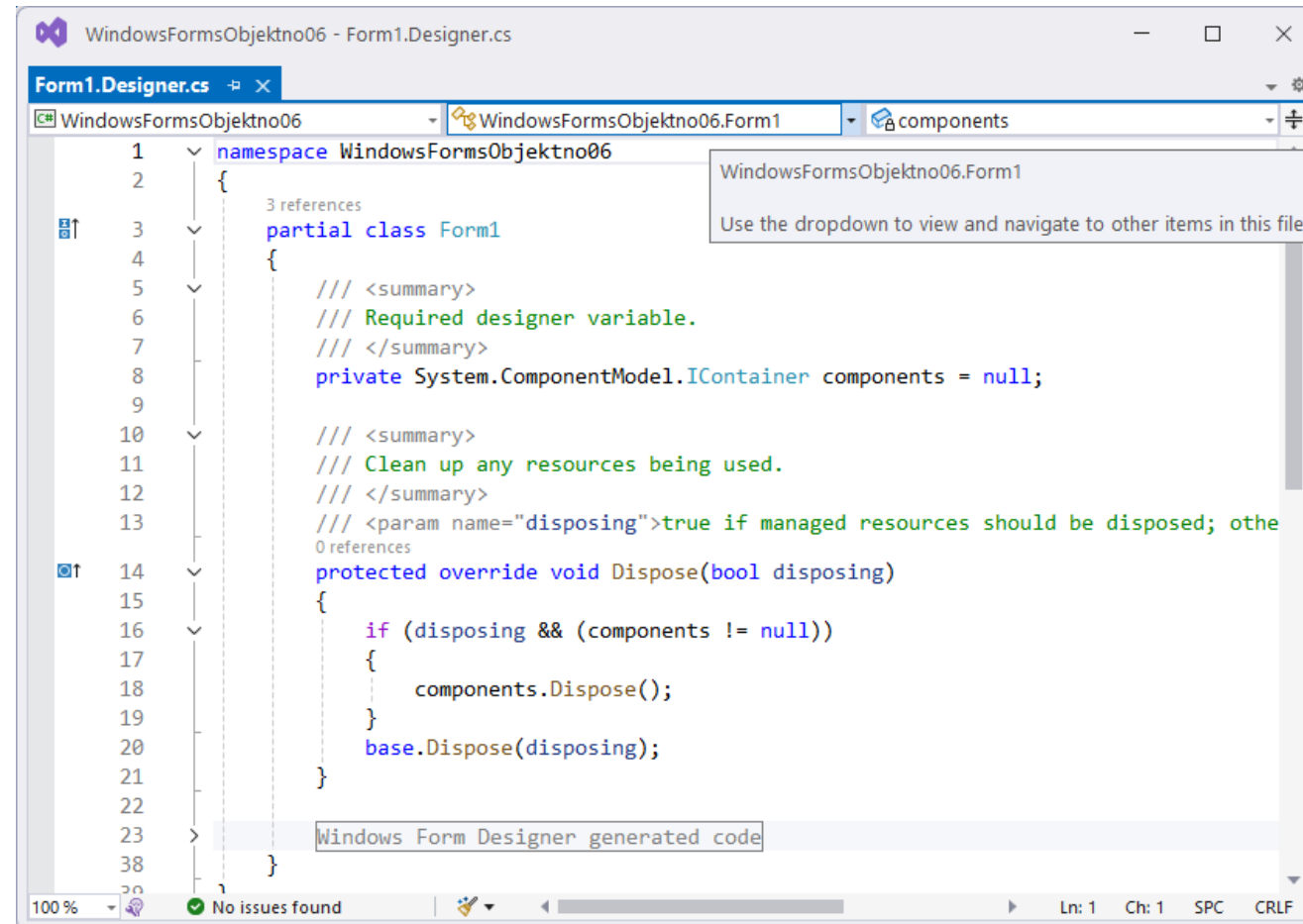
Parcijalne klase i struktura forme

- Klasa se deli u više fajlova pomoću ključne reči **partial**
- Omogućavaju odvajanje automatski generisanog koda od koda koji piše programer
- Poboljšavaju preglednost i olakšavaju održavanje projekta
- Prilikom dodavanja Windows Forme u projekat, Visual Studio automatski kreira dva povezana fajla:
 - **Form1.cs**
 - Sadrži logiku aplikacije (eventi, funkcije, obrada podataka)
 - Kod piše programer
 - **Form1.Designer.cs**
 - Sadrži automatski generisan kod dizajnera
 - Definiše izgled forme, kontrole i njihovu inicijalizaciju
 - Ne menja se ručno

Fajl Form1.cs

```
namespace WindowsFormsObjektno06
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

Fajl Form1.Designer.cs



Inicijalizacija komponenti forme

- `InitializeComponent()` je automatski generisana metoda u `Form1.Designer.cs`
- Poziva se u konstruktoru forme u fajlu `Form1.cs` prilikom njenog kreiranja
- Kreira sve kontrole korisničkog interfejsa
- Postavlja njihova svojstva (tekst, veličina, pozicija, boja...)
- Dodaje kontrole na formu i povezuje ih sa događajima
- Podešava osnovna svojstva same forme
- Ne menja se ručno – ažurira se kroz Visual Designer

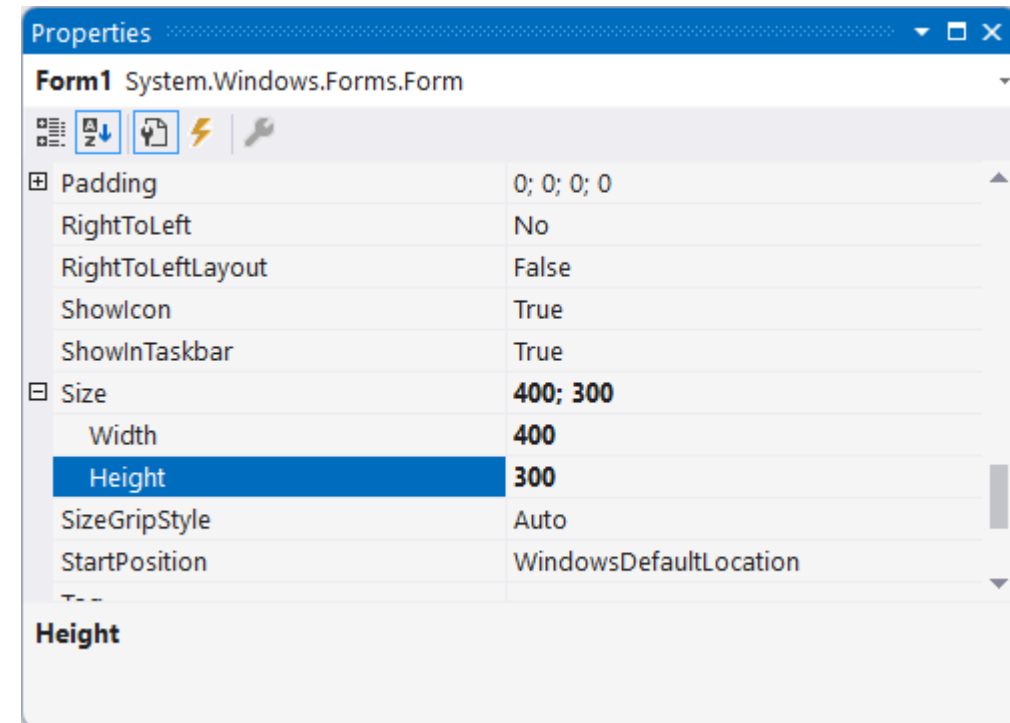
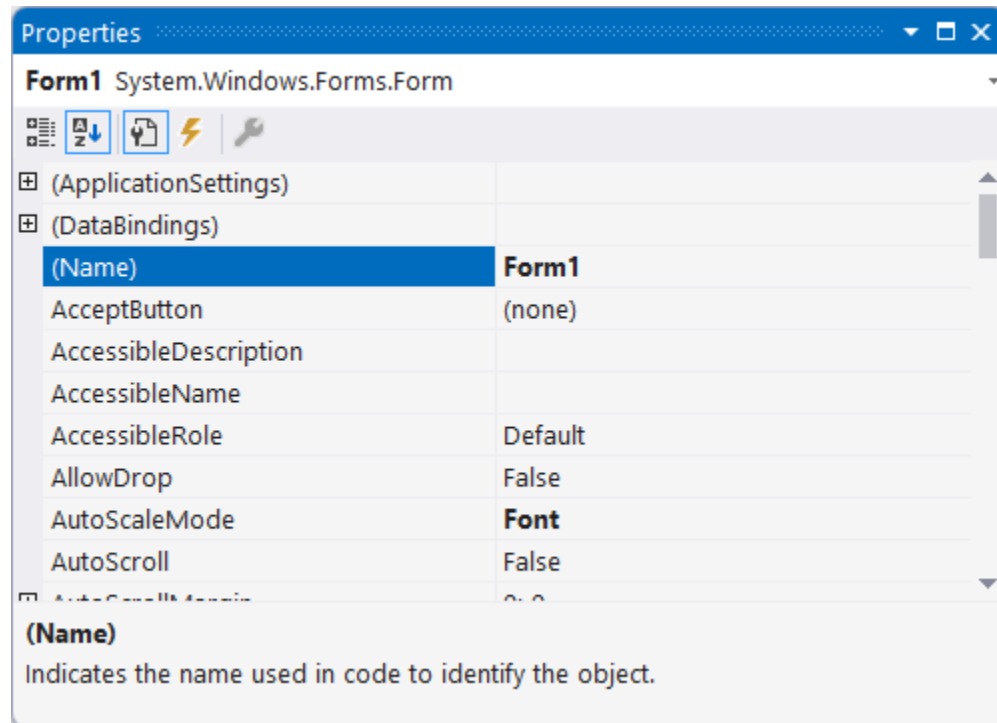
Klasa Program

```
internal static class Program
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main()
    {
        Application.EnableVisualStyles();
        Application.SetCompatibleTextRenderingDefault(false);
        Application.Run(new Form1());
    }
}
```

Klasa Program

- Kod klase Program se automatski generiše prilikom kreiranja Windows Forms projekta
- Main metoda
 - Glavna ulazna tačka aplikacije
 - Pokreće inicijalizaciju i otvara prvu formu
- `Application.Run(new Form1())` – pokreće glavnu formu i aplikaciju

Svojstva windows forme



Svojstva windows forme

- **Name** – uobičajeno ima isto ime kao klasa forme, ali nema istu ulogu kao kod drugih kontrola. Name je samo **svojstvo** instance, a ne ime reference u kodu
- **Text** - određuje tekst koji će biti prikazan na naslovnoj liniji forme
- **StartPosition** - određuje početnu poziciju forme na ekranu
- **Size (Width / Height)** - određuje širinu i visinu forme
- **WindowState** - omogućava prikaz forme u normalnoj veličiti, minimiziranu i maksimiziranu
- Pri pokretanju aplikacije kreira se objekat forme, zajedno sa objektima koji odgovaraju njenim kontrolama na korisničkom interfejsu

Event-vođeno programiranje

- Windows Forms aplikacije koriste event-vođeno programiranje
- Kod se ne izvršava redom kao u konzolnim aplikacijama
- Program čeka da korisnik ili sistem izazove određenu akciju, npr.:klik mišem, pritisak tastera, učitavanje forme...
- Forma reaguje izvršavanjem odgovarajuće metode koja se naziva **event handler**
- **Event handler** je metoda koja se izvršava kada se dogodi određeni događaj
- Predstavlja reakciju programa na akciju korisnika ili sistemski događaj
- Programer povezuje handler sa konkretnim događajem (npr. Load, Click, KeyDown...)

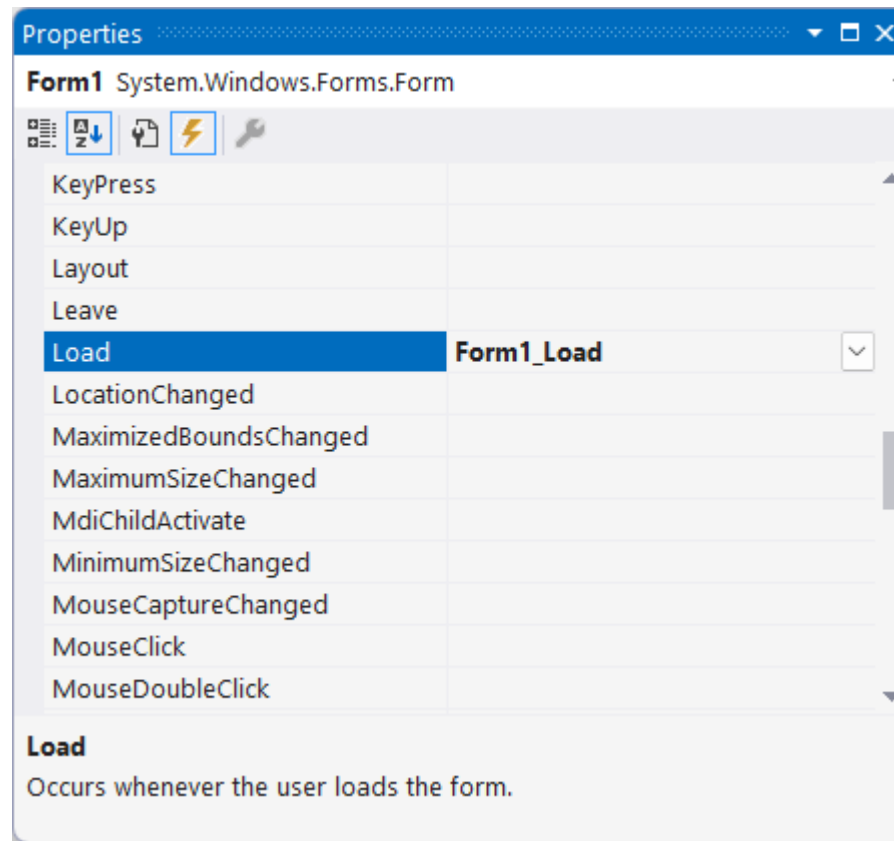
Događaji Windows forme (Events)

- Forma poseduje veliki broj događaja koji se aktiviraju tokom njenog životnog ciklusa
- Događaji omogućavaju reagovanje na akcije korisnika ili promene stanja forme
- Najčešći događaji:
 - Load – forma se učitava u memoriju
 - Shown – forma je prikazana korisniku
 - Resize – promenjena veličina forme
 - FormClosing – aplikacija ili forma se zatvara
 - Click, MouseMove, KeyDown – korisničke interakcije

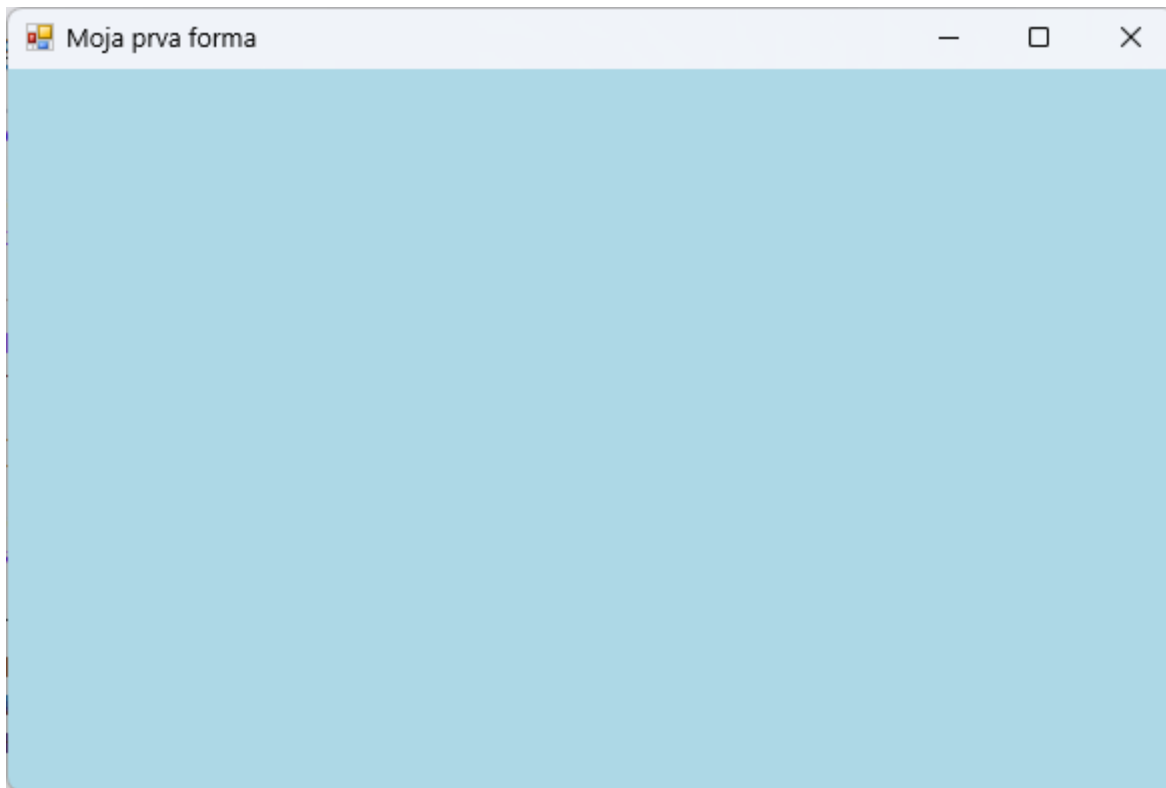
Load događaj forme

- Aktivira se pre nego što forma postane vidljiva
- Koristi se za:
 - Inicijalizaciju podataka
 - Podešavanje kontrola
 - Učitavanje resursa ili konekciju sa bazom
 - Pripremu korisničkog interfejsa pre prikazivanja
- U Properties prozoru postoji kartica Events (ikona munje)
- U ovoj kartici se prikazuju svi događaji forme
- Dvostrukim klikom na naziv događaja automatski se kreira odgovarajući event handler
- Na primer, dvostruk klik na Load kreira metodu Form1_Load
- Dvostruki klik na samu formu takođe automatski kreira handler za Load događaj (Form1_Load)

Primer obrade Load događaja forme



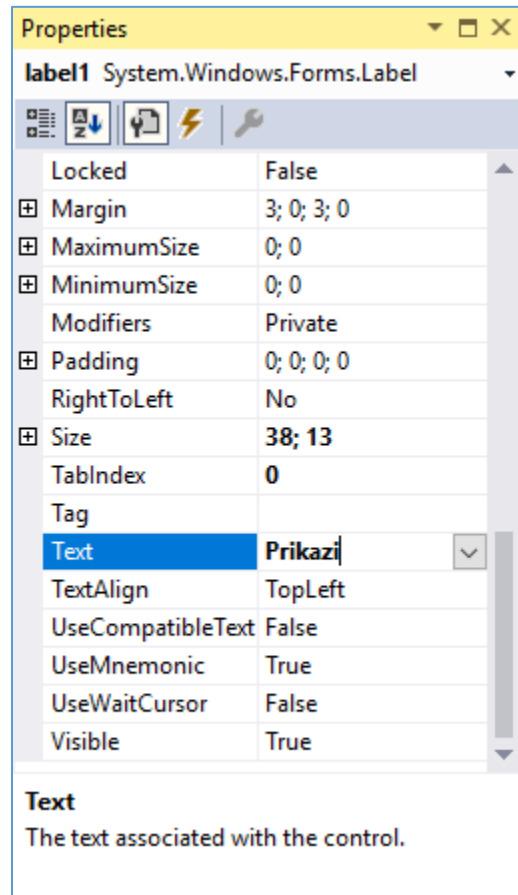
```
private void Form1_Load(object sender, EventArgs e)
{
    Text = "Moja prva forma";
    BackColor = Color.LightBlue;
    Width = 600;
    Height = 400;
}
```



Kontrola Label

- Koristi se za predstavljanje opisnog teksta korisniku
- Obično se koristi u kombinaciji sa kontrolama za unos i editovanje teksta
- **Text** property definiše tekst koji će se prikazati na labeli
- **Name** svojstvo predstavlja ime reference na objekat Label kontrole koji se kreira pri pokretanju aplikacije
- Korisnik ne može menjati tekst na labeliLabela ne može da dobije fokus

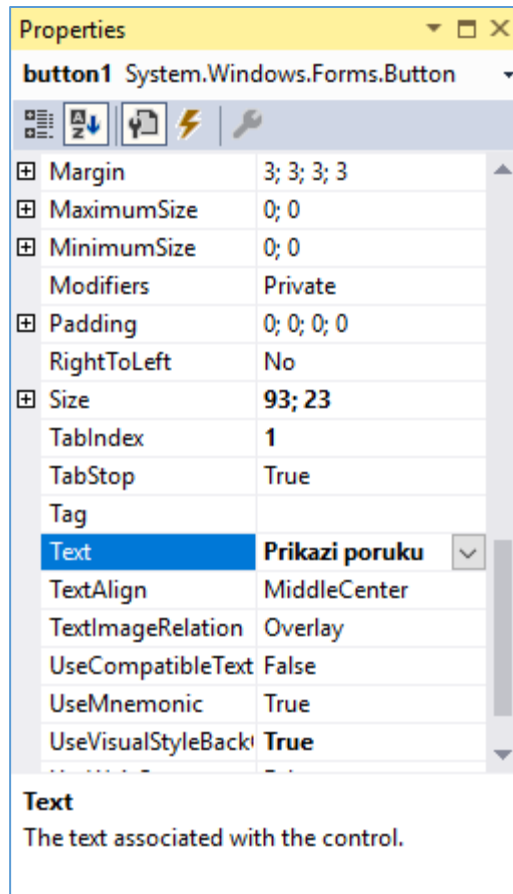
Properties prozor kontrole Label



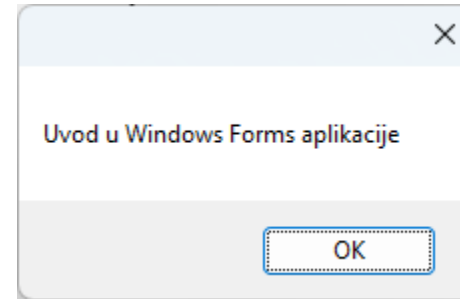
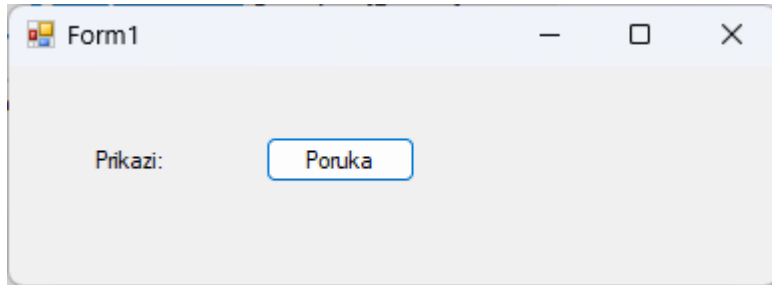
Kontrola Button

- Podrazumevani događaj ove kontrole je događaj Click
- Na dugme se može kliknuti mišem ili pritiskom na taster ENTER ili SPACE kada dugme ima fokus
- Svojstvo **Name** daje ime reference na objekat dugmeta koje će biti kreirano pri pokretanju aplikacije
- Svojstvo **Text** definiše tekst koji će se prikazati na dugmetu
- Svojstvo **Enabled** kontroliše da li je dugme omogućeno ili onemogućeno
- Svojstvo **Visible** određuje da li je dugme vidljivo na formi

Properties prozor kontrole Button



Uvodna Windows Forms aplikacija



```
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show("Uvod u Windows Forms aplikacije");
}
```

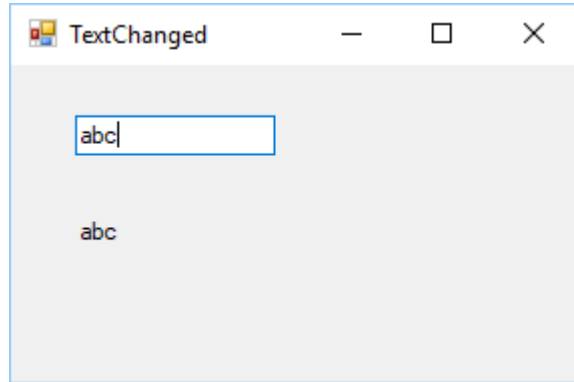
TextBox kontrola

- Koristi se za prihvatanje ulaza od strane korisnika ili za prikaz vrednosti
- Svojstvo **Name** predstavlja ime reference na objekat klase TextBox koji će biti kreiran pri pokretanju aplikacije
- Svojstvo **Text** se koristi za čitanje sadržaja TextBox-a ili prikaz sadržaja u TextBox-u
- Metoda **Clear()** briše sadržaj ove kontrole
- Metoda **Focus()** postavlja fokus na određenu TextBox kontrolu
- Svojstvo **MaxLength** definiše maksimalni broj karaktera za TextBox

TextBox kontrola

- Svojstvo **PasswordChar** omogućava da se prikazuje određeni znak kada korisnik unosi tekst
- Svojstvo **Enabled**, postavljeno na false, omogućava da se TextBox koristi kao read-only kontrola
- Podrazumevani događaj ove kontrole je **TextChanged**, koji se generiše svaki put kada se promeni tekst u TextBox kontroli
- Događaj **KeyDown** generiše se kada se pritisne neki taster na tastaturi pre nego što se otpusti

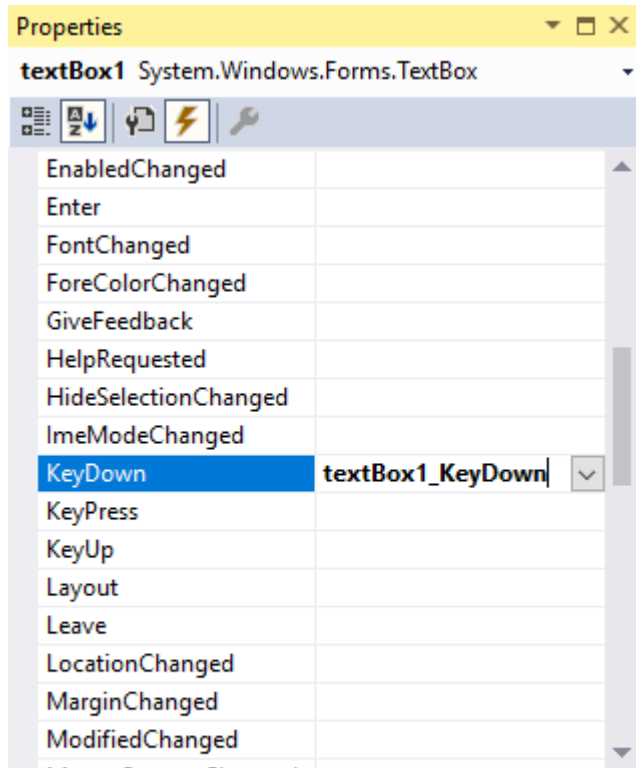
Obrada događaja TextChanged kontrole TextBox



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

sender – predstavlja kontrolu koja je pokrenula događaj(u ovom slučaju to je textBox1)
EventArgs e – sadrži informacije o događaju

Obrada događaja KeyDown



KeyDown

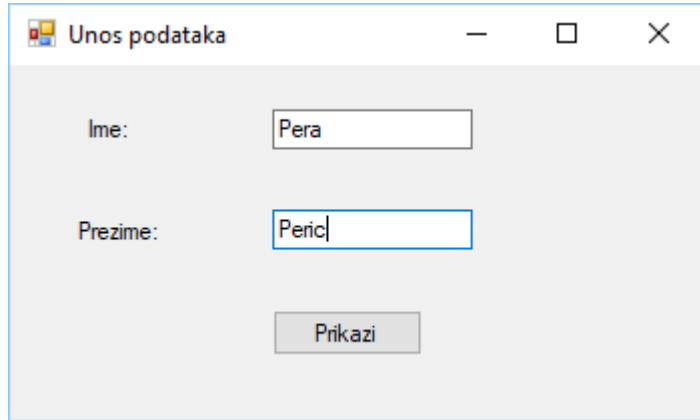
Occurs when a key is first pressed.

```
private void textBox1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        MessageBox.Show("Pritisnuli ste taster Enter", "Poruka");
    }
}
```

sender – predstavlja TextBox koji je pritisnuo taster

KeyEventArgs e – sadrži informacije o tasteru (e.KeyCode, e.Shift, e.Control...)

Primer upotrebe TextBox kontrole - GUI

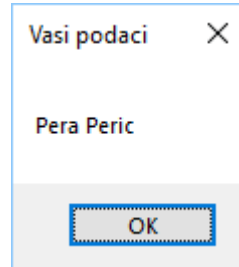


Unos podataka

Ime: Pera

Prezime: Peric

Prikazi



Vasi podaci

Pera Peric

OK

Primer upotrebe TextBox kontrole - kod

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

    MessageBox.Show(ime + " " + prezime, "Vasi podaci");
    textBox1.Clear();
    textBox2.Clear();
    textBox1.Focus();
}
```

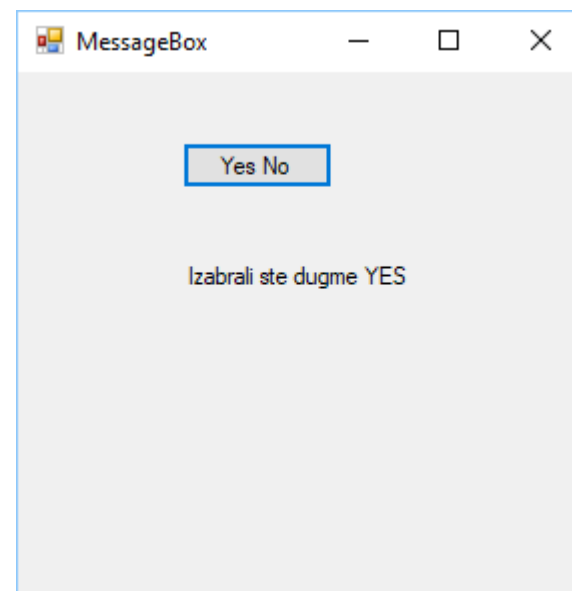
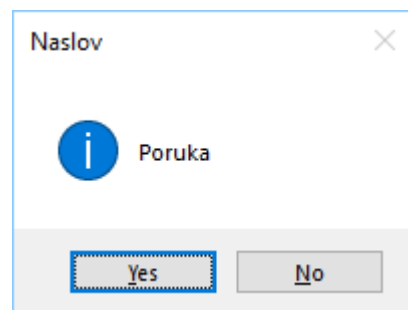
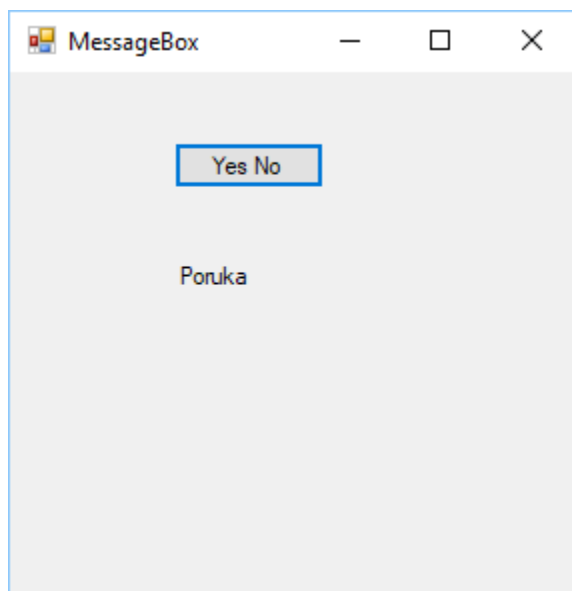
MessageBox klasa i DialogResult enumeracija

- Klasa **MessageBox** služi za prikaz poruka, upozorenja i pitanja korisniku u vidu modalnog dijaloga
- Metoda **Show()** prikazuje dijalog i vraća vrednost tipa **DialogResult**
- **DialogResult** predstavlja dugme koje je korisnik pritisnuo
- Najčešće vrednosti: **OK, Cancel, Yes, No**
- Povratna vrednost se koristi za donošenje odluke u programu

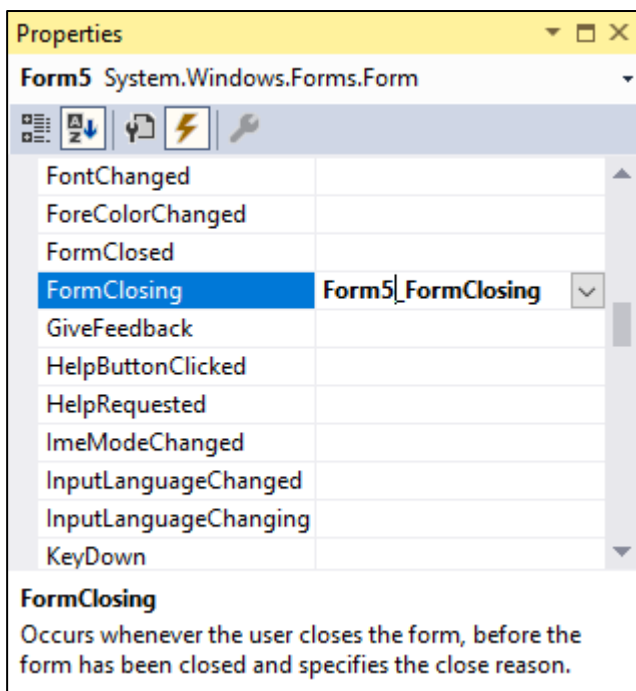
Show() metoda klase MessageBox

```
private void button1_Click(object sender, EventArgs e)
{
    DialogResult rez = MessageBox.Show("Poruka", "Naslov",
    MessageBoxButtons.YesNo,
    MessageBoxIcon.Information);
    switch (rez)
    {
        case DialogResult.Yes:
            label1.Text = "Izabrali ste dugme YES";
            break;
        case DialogResult.No:
            label1.Text = "Izabrali ste dugme NO";
            break;
    }
}
```

Show() metoda klase MessageBox

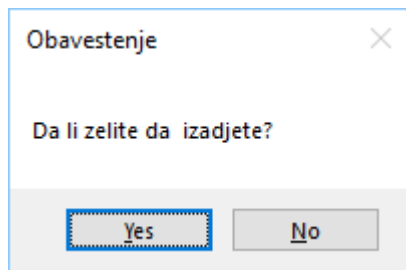


Generisanje handlera za Closing događaj forme



Obrada FormClosing događaja forme

```
private void Form1_FormClosing(object sender, FormClosingEventArgs e)
{
    DialogResult rez = MessageBox.Show("Da li zelite da izađete?",
        "Obavestenje",
        MessageBoxButtons.YesNo);
    if (rez == DialogResult.No)
    {
        e.Cancel = true;
    }
}
```



Pitanje 1

Zašto se klasa Form1 u Windows Forms projektu definiše kao partial?

- a. Da bi definicija klase mogla biti podeljena na više fajlova
- b. Da bi klasa automatski nasleđivala više drugih klasa
- c. Da bi kod unutar klase izvršavao brže

Odgovor: a

Pitanje 2

Gde je definisana metoda InitializeComponent() koja se poziva u konstruktoru forme?

- a. U fajlu Form1.Designer.cs kao deo automatski generisanog koda
- b. U fajlu *Form1.cs* gde programer piše svoj kod
- c. U Program.cs fajlu zajedno sa Main metodom

Odgovor: a

Pitanje 3

Koja od sledećih kontrola ne može da dobije fokus:

- a. TextBox
- b. Label
- c. Button

Odgovor: b

Pitanje 4

FormLoad događaj forme izvršava se:

- a. Nakon što se forma prikaže korisniku
- b. Svaki put kada forma postane aktivna
- c. Pre nego što forma prvi put postane vidljiva korisniku

Odgovor c

Pitanje 5

Metoda Show() klase MessageBox :

- a. vraća enumeraciju tipa DialogResult
- b. ne vraća ništa
- c. vraća true ako je rezultat pozitivan u protivnom vraća false

Odgovor: a

Pitanje 6

Na koji način se u FormClosing događaju može sprečiti zatvaranje forme?

- a. Postavljanjem `e.Cancel = true`
- b. `e.StopClosing = true`
- c. `e.BlockAction = true`

Odgovor: a

Pitanje 7

Na koji način se u FormClosing događaju može sprečiti zatvaranje forme?

- a. Postavljanjem `e.Cancel = true`
- b. `e.StopClosing = true`
- c. `e.BlockAction = true`

Odgovor: a