

# Primena OOP u izradi složenih aplikacija (programski jezik C#)

## Uvod u WinForms (1)

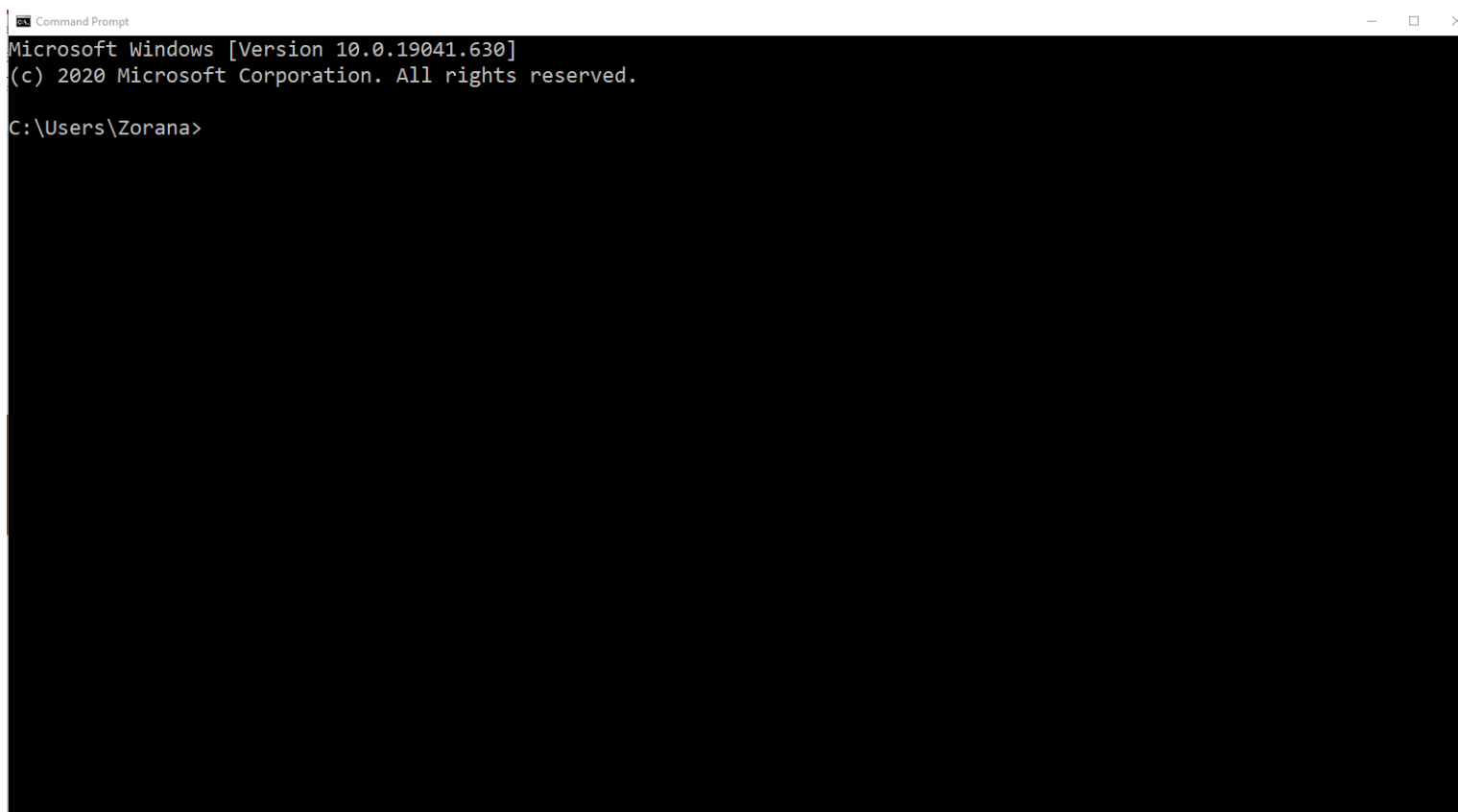
# Šta je korisnički interfejs?

- ❑ Instrument za interakciju softvera sa korisnikom
- ❑ Veoma važan segment jednog softverskog rešenja
- ❑ Dobro projektovani korisnički interfejs je ono što čini dobar softverski proizvod
- ❑ Govoreći terminologijom tržišta informacionih tehnologija, korisnički interfejs je ono što prodaje softver, ono što jedno rešenje čini drugačijim od ostalih

# Vrste korisničkog interfejsa

## ❑ Komandni interfejs

- Potrebno je poznavati komande kako bi bilo moguće upravljati softverom

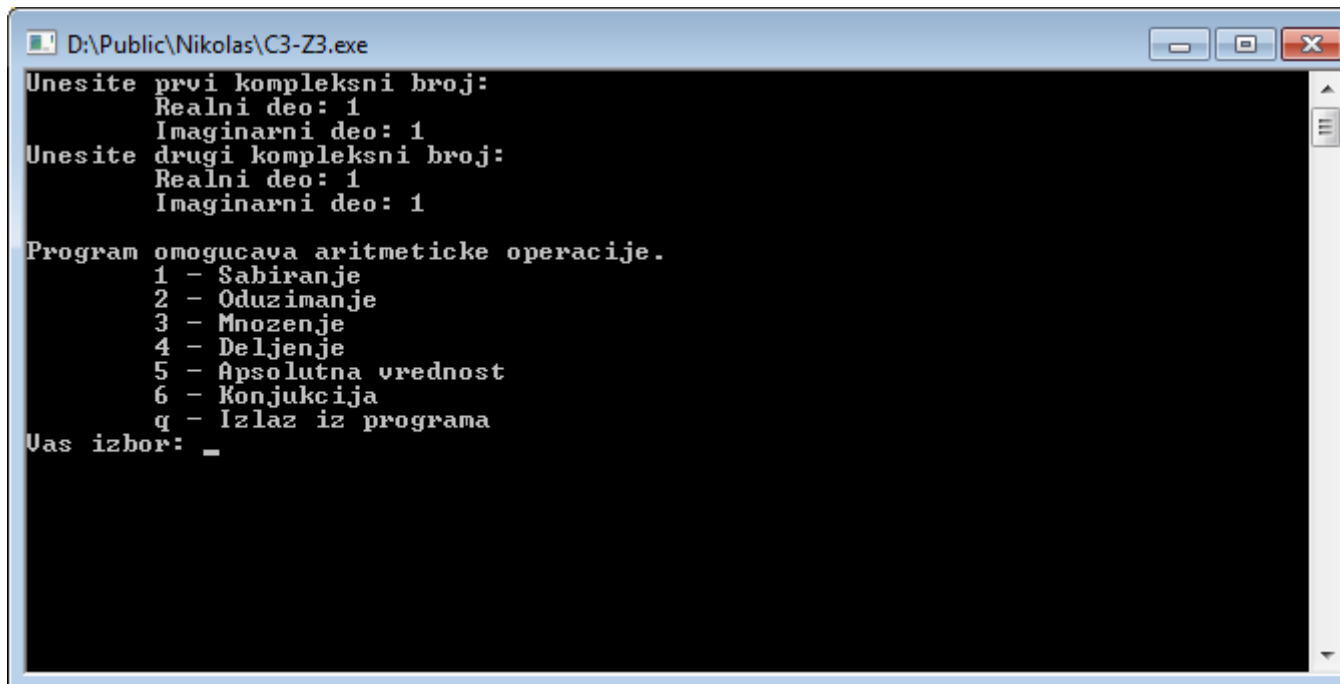


```
Command Prompt
Microsoft Windows [Version 10.0.19041.630]
(c) 2020 Microsoft Corporation. All rights reserved.
C:\Users\Zorana>
```

# Vrste korisničkog interfejsa (nastavak)

## ☐ Meni interfejs

- Upravljanje softverom se vrši odabirom opcija iz menija



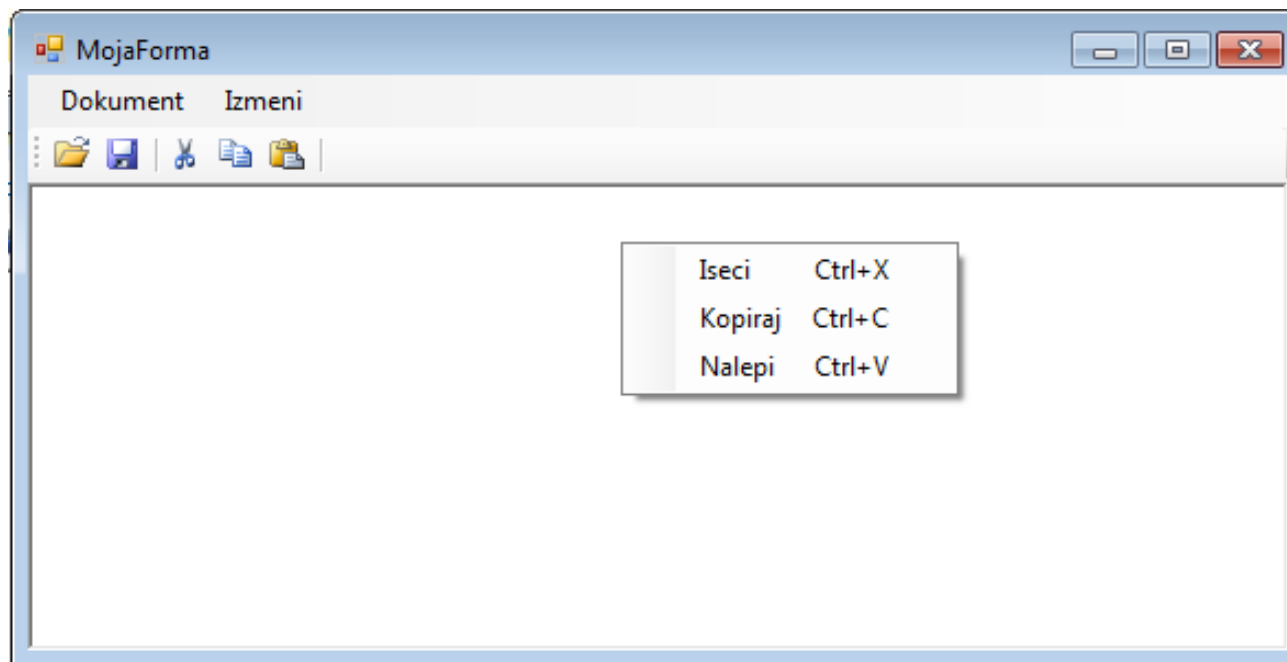
```
D:\Public\Nikolas\C3-Z3.exe
Unesite prvi kompleksni broj:
  Realni deo: 1
  Imaginarni deo: 1
Unesite drugi kompleksni broj:
  Realni deo: 1
  Imaginarni deo: 1

Program omogućava aritmetičke operacije.
  1 - Sabiranje
  2 - Oduzimanje
  3 - Množenje
  4 - Deljenje
  5 - Apsolutna vrednost
  6 - Konjugacija
  q - Izlaz iz programa
Vas izbor: _
```

# Vrste korisničkog interfejsa (nastavak)

## □ Grafički korisnički interfejs

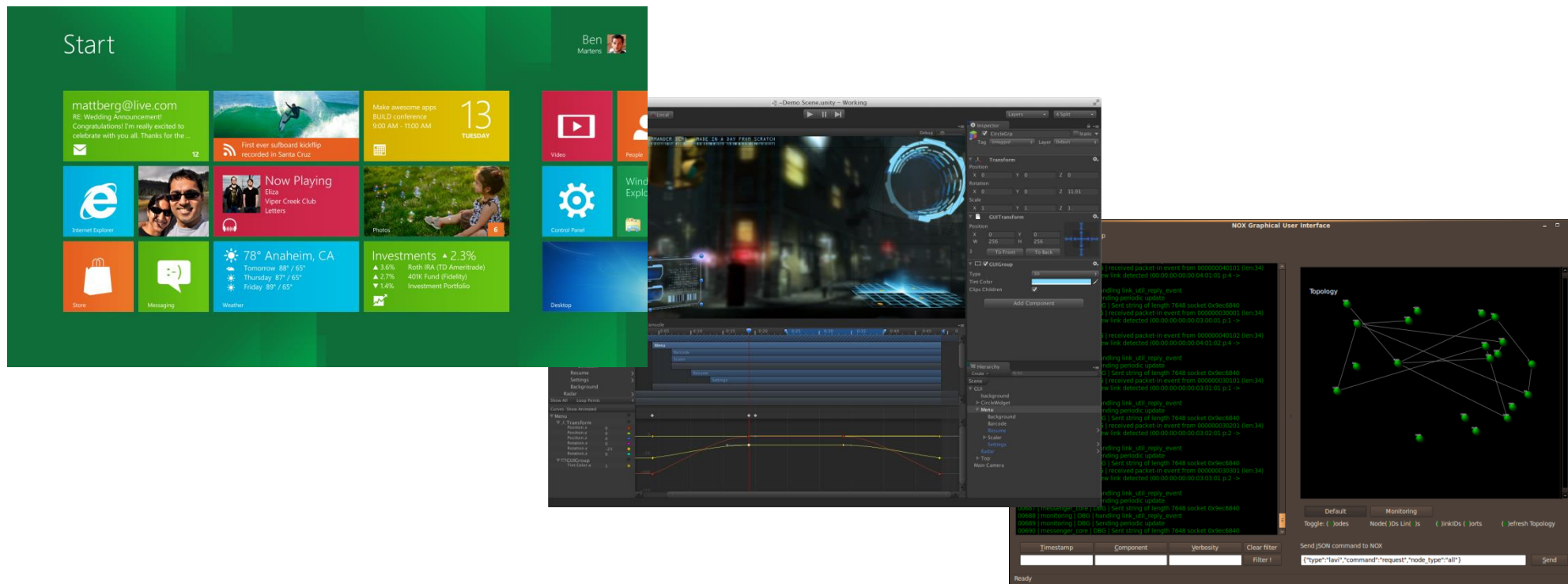
- Upravljanje softverom se vrši interakcijom sa različitim grafičkim komponentama kao što su ikonice, meniji, dugmići...



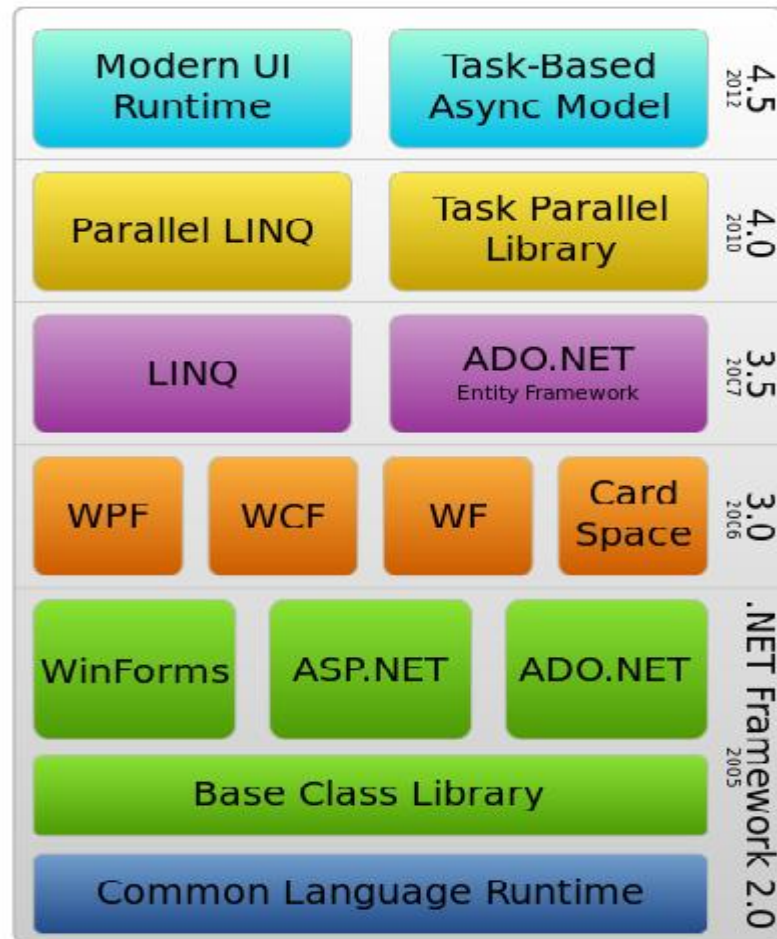
# Šta je GUI?

CENTAR ZA  
mlade talente

- GUI (eng. *Graphical User Interface*) je grafički korisnički interfejs koji predstavlja i određuje način interakcije korisnika sa računarom



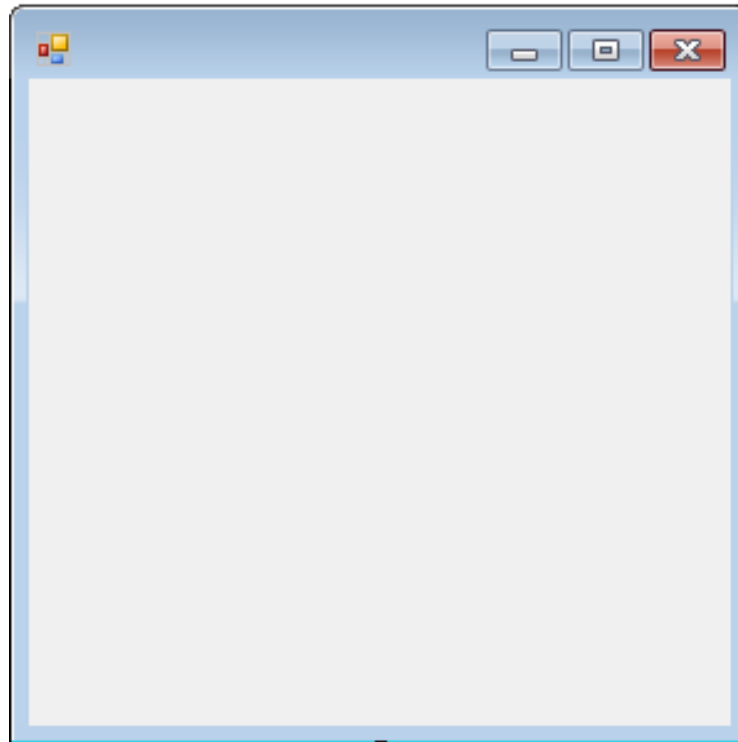
# .NET Framework stek - WinForms



The .NET Framework Stack

# Forme

- ❑ GUI aplikacije se sastoje od prozora koji u sebi imaju druge elemente grafičkog interfejsa. Ti prozori se nazivaju forme



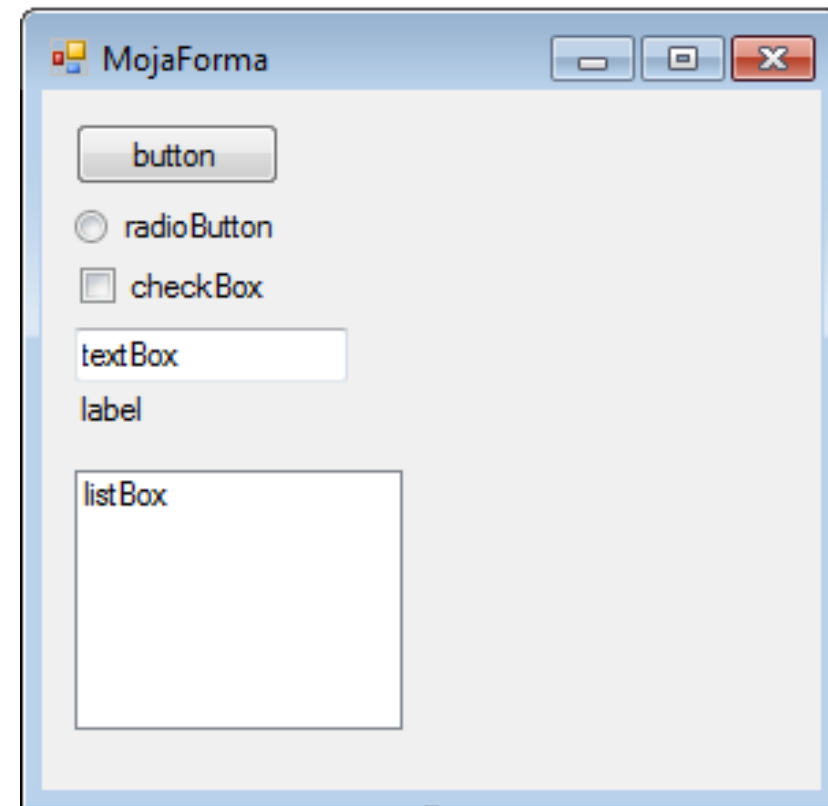


# Kontrole

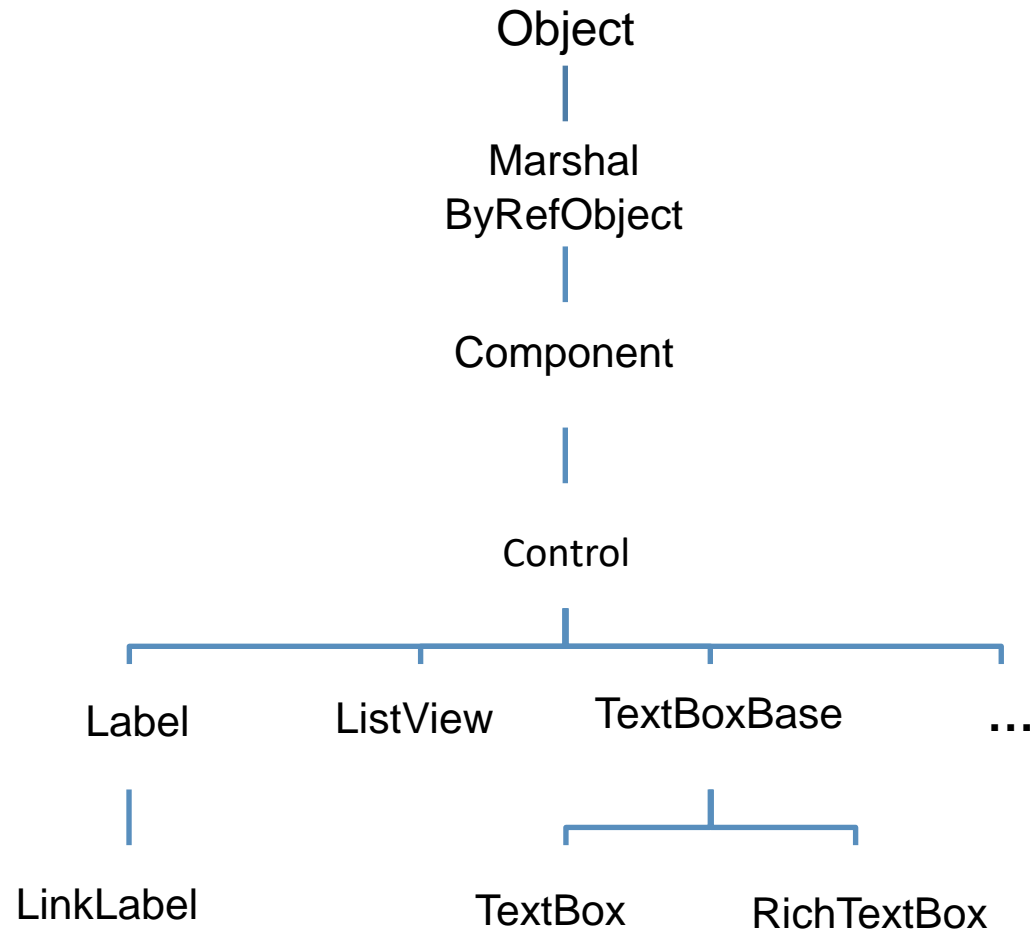
❑ Kontrole su elementi grafičkog interfejsa koji služe za manipulaciju podacima

❑ Neke od kontrola su:

- Button
- RadioButton
- CheckBox
- ListBox
- Label
- TextBox
- ....



# Logičko stablo kontrola



# Svojstva

- ❑ Svaka kontrola ima određeni broj svojstava (eng. *Properties*) koja služe za podešavanje njenog ponašanja
- ❑ Osnovna klasa za većinu kontrola (**Control**) ima svoja svojstva koja ostale kontrole direktno nasleđuju ili redefinišu kako bi dobili neko izmenjeno ponašanje

# Spisak najbitnijih svojstava

Ime	Dostupnost	Opis
Anchor	Čitanje/Pisanje	Korišćenjem ovog svojstva može se specificirati ponašanje kontrole pri menjanju veličine kontejnera
BackColor	Čitanje/Pisanje	Pozadinska boja kontrole
Bottom	Čitanje/Pisanje	Rastojanje donje ivice kontrole u odnosu na gornju ivicu prozora
Dock	Čitanje/Pisanje	Omogućuje da se kontrola smesti i spoji sa ivicama prozora
Enabled	Čitanje/Pisanje	Određuje da li kontrola reaguje na korisnikove akcije. Postavljanje na <i>true</i> obično znači da može da reaguje, a za vrednost <i>false</i> važi obrnuto
ForeColor	Čitanje/Pisanje	Primarna boja kontrole
Height	Čitanje/Pisanje	Visina kontrole
Left	Čitanje/Pisanje	Rastojanje leve ivice kontrole u odnosu na levu ivicu prozora
Name	Čitanje/Pisanje	Ime kontrole koje može da se koristi kao referenca na kontrolu u kodu
Parent	Čitanje/Pisanje	Kontrola koja sadrži trenutnu kontrolu

# Spisak najbitnijih svojstava (nastavak)

Ime	Dostupnost	Opis
Right	Čitanje/Pisanje	Rastojanje desne ivice kontrole u odnosu na levu ivicu prozora
TabIndex	Čitanje/Pisanje	Redni broj koji kontrola ima u <i>Tab</i> redosledu svog kontejnera
TabStop	Čitanje/Pisanje	Definiše da li se može pristupiti kontroli upotrebom <i>Tab</i> tastera
Tag	Čitanje/Pisanje	Vrednost ovog svojstva obično se ne koristi od strane kontrole. Služi za smeštanje proizvoljne vrednosti koja je korisniku potrebna tokom razvoja programa, a koja je vezana za datu kontrolu
Top	Čitanje/Pisanje	Rastojanje gornje ivice kontrole u odnosu na gornju ivicu prozora
Visible	Čitanje/Pisanje	Specificira da li će kontrola biti vidljiva ili ne. Postavljanje na <i>true</i> znači da je vidljiva, a za vrednost <i>false</i> važi obrnuto
Width	Čitanje/Pisanje	Širina kontrole

# Događaji

- ❑ Događaji (eng. *Events*) definišu ponašanje kontrole u slučajevima kada se izvrše određene akcije (klik mišem, dupli klik, prevlačenje, itd.) nad korisničkim interfejsom
- ❑ Da bi bilo moguće odreagovati na akcije nad kontrolom, implementiraju se metode za obradu događaja (eng. *Handler*) koje se izvršavaju kao rezultat izvođenja akcije
- ❑ Klasa **Control** definiše određeni broj događaja koji su uobičajeni za većinu kontrola
- ❑ Svaka kontrola može da implementira neke sebi karakteristične događaje

# Spisak najčešćih događaja

Ime	Opis
Click	Okida se kada je izvršen klik mišem na kontrolu ili u nekim slučajevima kada korisnik pritisne taster Enter
DoubleClick	Okida se kada je izvršen dupli klik mišem na kontrolu. Na nekim kontrolama gde je podržan <i>Click</i> događaj, nemoguće je generisati <i>DoubleClick</i> događaj
DragDrop	Okida se kada je operacija prevlačenja završena
DragEnter	Okida se kada objekat koji se prevlači uđe u "oblast" kontrole
DragLeave	Okida se kada objekat koji se prevlači napusti "oblast" kontrole
DragOver	Okida se kada se objekat koji se prevlači prevuče preko kontrole
KeyDown	Okida se kada je dugme na tastaturi pritisnuto dok kontrola ima fokus <sup>1</sup> , i okida se uvek pre <i>KeyPress</i> i <i>KeyUp</i> događaja

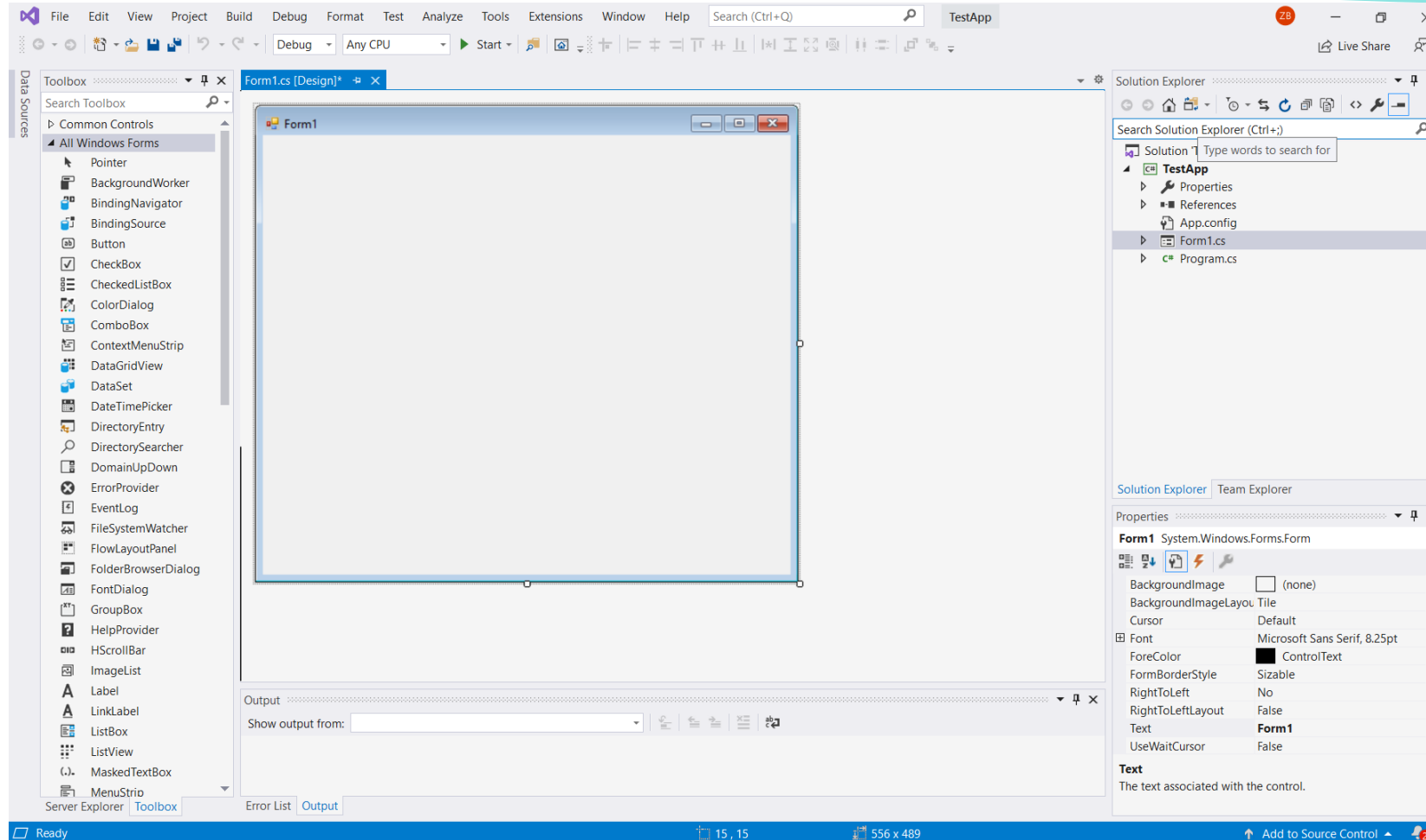
# Spisak najčešćih događaja (nastavak)

Ime	Opis
KeyPress	Okida se kada je dugme na tastaturi pritisnuto dok kontrola ima fokus. Okida se uvek posle <i>KeyDown</i> , a pre <i>KeyUp</i> događaja. Razlika između <i>KeyPress</i> i <i>KeyDown</i> događaja je što <i>KeyDown</i> prosleđuje kod dugmeta sa tastature koje je pritisnuto, dok <i>KeyPress</i> prosleđuje odgovarajuću vrednost karaktera za to dugme
KeyUp	Okida se kada je pušteno dugme na tastaturi dok kontrola ima fokus i okida se posle <i>KeyPress</i> događaja
GotFocus	Okida se kada kontrola dobije fokus
LostFocus	Okida se kada kontrola izgubi fokus
MouseDown	Okida se kada se kursor miša nalazi na kontroli i kada je dugme miša pritisnuto. Ovo nije isti događaj kao i <i>MouseClicked</i> zato što se <i>MouseDown</i> okida čim se pritisne dugme i pre nego što je pušteno
MouseMove	Okida se neprekidno dok kursor miša putuje preko kontrole
MouseUp	Okida se kada se kursor miša nalazi na kontroli i kada je dugme miša pušteno
Paint	Okida se kada se iscrtava kontrola



# Uvod u VS okruženje za razvoj WinForms aplikacija

CENTAR ZA  
mlade talente



# Raspored kontrola na formi

- ❑ Postoji više načina uz pomoć kojih je moguće definisati raspored i ponašanje kontrola pri izmeni veličine prozora
- ❑ Neki od načina su preko:
  - Anchor (svojstvo)
  - Dock (svojstvo)
  - Layout

# Layout

- ❑ *Layout* je kontejner koji je odgovoran za raspored kontrola i njihovo skaliranje na formi
- ❑ Takođe je odgovoran za njihovo ponašanje pri skupljanju i razvlačenju forme
- ❑ Neki od *layout*-ova su:
  - Panel
  - FlowLayout
  - TableLayout

# Hvala na pažnji!