

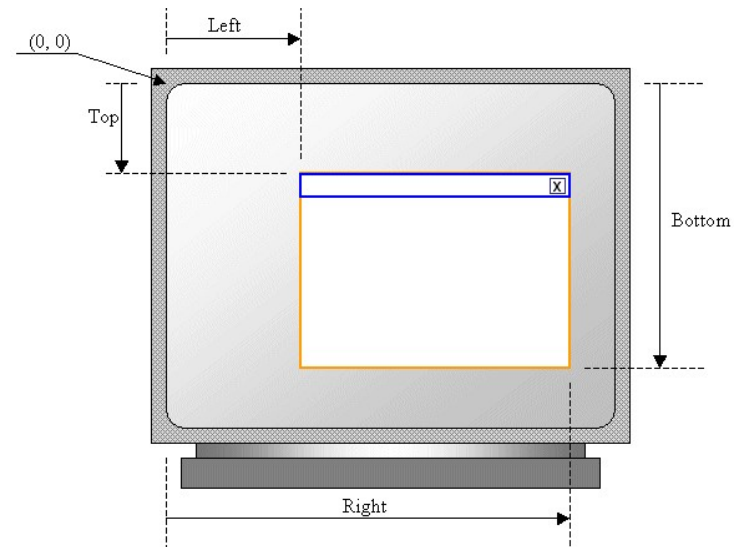
Osnovno o Windows kontrolama

DIZAJN Windows kontrola

► Osobine Windows kontrola

- Zajedničko za sve kontrole je da su po svojoj prirodi takođe **prozori**.
- Imaju većinu sličnih svojstava.
- Prihvatanje određenih događaja je tipično za kontrole.
- Neki događaji su specifični samo za određenu kontrolu dok su drugi zajednički.

Koordinatni sistem



Relacija *roditelj-dete*

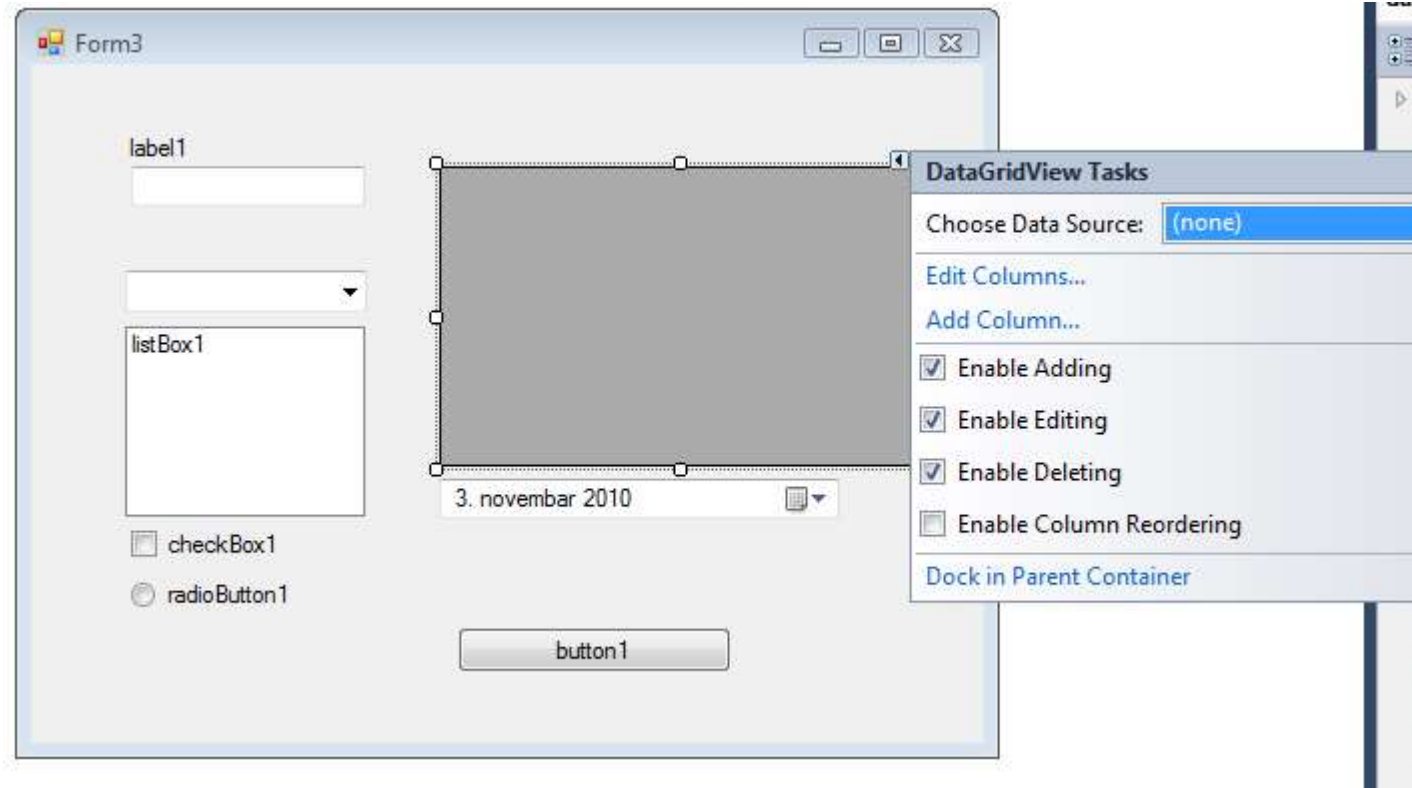
- ▶ *Parent-child relationship*
- ▶ *Ovo je podrazumevana relacija kada se kontrole postaljaju i manifestuje se “lepljenjem” kontrole za tu formu. Pomeranjem forme pomeraju se sve kontrole na njoj, tj. sva deca prozori.*

Neka zajednička svojstva Windows kontrola

- ▶ *Name*
- ▶ *Location*
- ▶ *Size*
- ▶ *Visible*
- ▶ *Enabled*
- ▶ *TabIndex i TabStop*
- ▶ *.....*

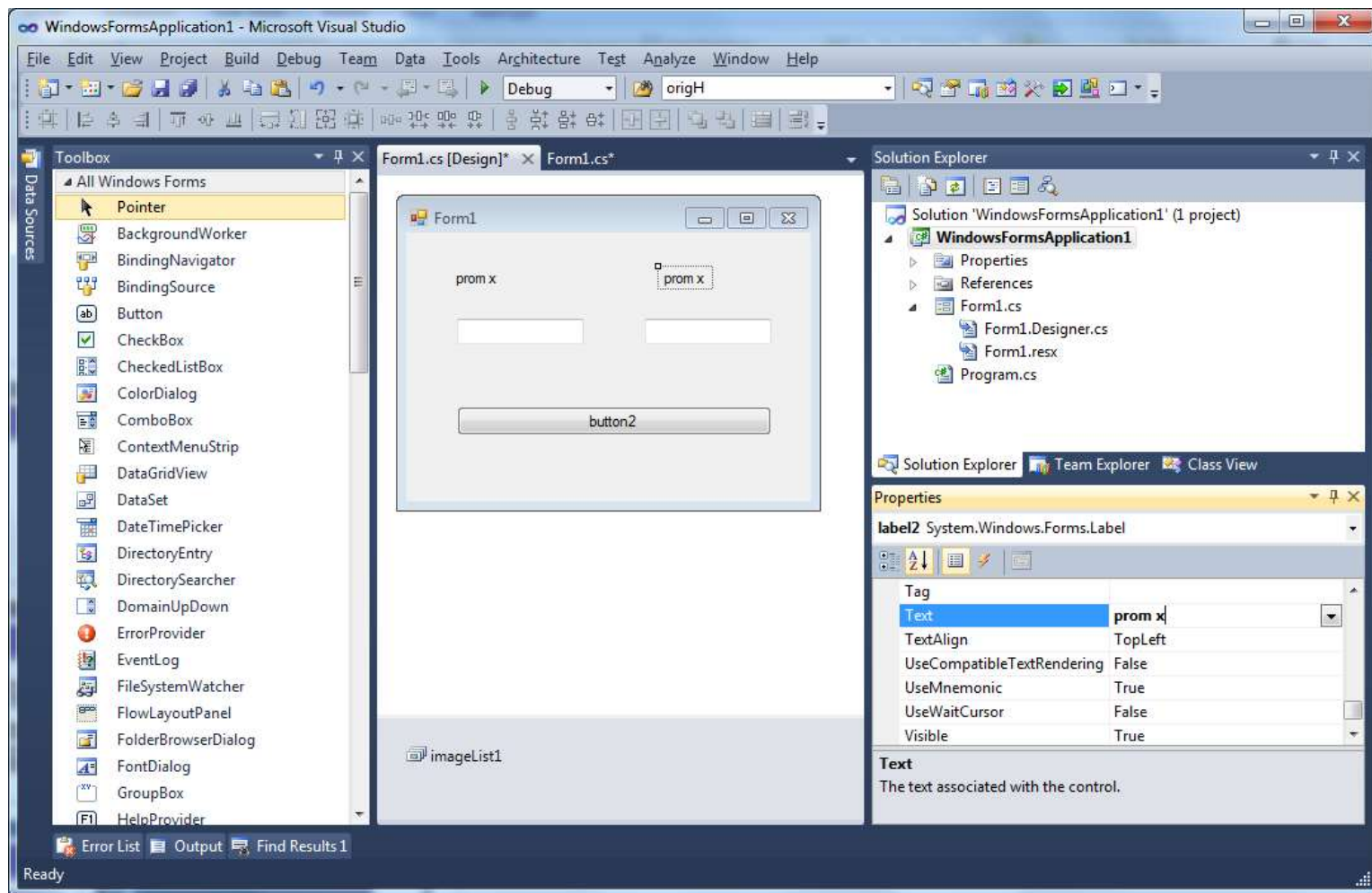
Osnovne kontrole

- ▶ Label
- ▶ TextBox
- ▶ ListBox
- ▶ ComboBox
- ▶ DataGridView
- ▶ Button
- ▶ CheckBox
- ▶ RadioButton



Primer

- ▶ Napisati Win program za sabiranje 2 vrednosti. Koristiti kontrole: Label, TextBox, Button i MessageBox klasu za prikaz poruka.
- ▶ Rezultate smestati u kontrolu ListBox.

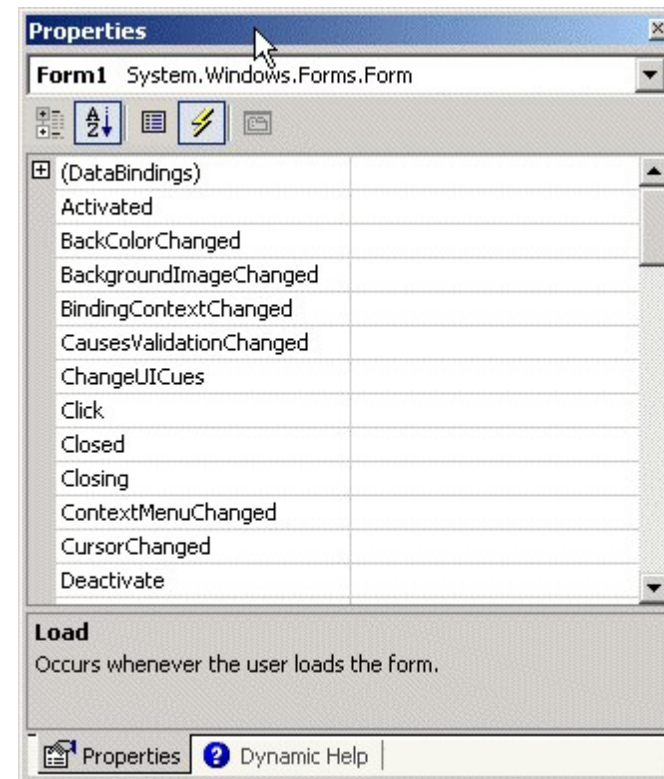


DOGAĐAJI

- ▶ Događaji (*events*) obezbeđuju aplikacijama, odnosno formama odnosno kontrolama da prepoznaju i da reaguju na akcije bilo korisnika bilo drugih aplikacija.
- ▶ Spadaju u najvažnije osobine višeprocenih operativnih sistema kao što su Windows.
- ▶ U toku rada neke aplikacije, kontrole i forme (preciznije - prozori) neprekidno šalju poruke operativnom sistemu o svim promenama i akcijama korisnika.

Događaji i IDE

- ▶ Ime svakog događaja je prikazano na levoj strani.
- ▶ Sa desne strane je kombo polje gde možete uneti novo ime i time kreirati novu metodu koja će biti vezana za taj događaj, ili izabrane neku od već postojećih sa liste.
- ▶ Kratak opis se nalazi na samom dnu i on vam može pomoći oko lakšeg snalaženja.



Metode koje reaguju na događaje...

- ▶ ...imaju potpis:
- ▶ ***private void myHandler***
(object sender, EventArgs e)
 - ▶ ***sender*** – objekat koji šalje događaj
 - ▶ ***e*** – argumenti tj. podaci koji se šalju uz događaj
- ▶ Da li možemo praviti naše događaje?
 - ▶ Naravno

Vrste događaja

- ▶ **Događaji miša**

- ▶ Click, Right-Click, MouseUp, MouseDows,...

- ▶ **Događaji tastature**

- ▶ KeyDown,KeyUp, KeyPress,...

- ▶ **Događaji vezani za formu i kontrole**

- ▶ Paint, Resize,

Više o njima na vežbama!

Kreiranje događaja u kodu koristeći IDE...

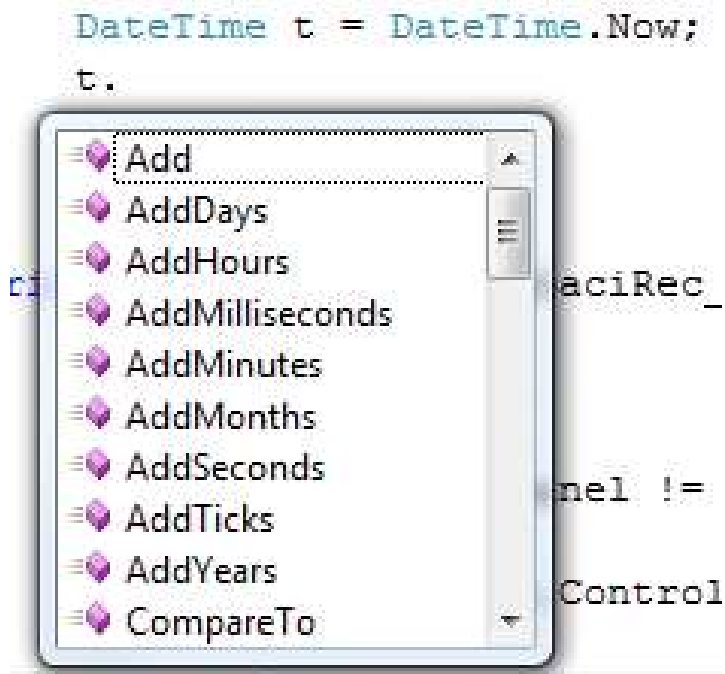
```
Button okbut = new Button();  
okbut.Click +=  
    new EventHandler(okbut_Click); (Press TAB to insert)
```

```
Button okbut = new Button();  
okbut.Click += new EventHandler(okbut_Click);  
Press TAB to generate handler 'okbut_Click' in this class
```

```
void okbut_Click(object sender, EventArgs e)  
{  
    throw new Exception("The method or operation is not implemented.");  
}
```

DateTime

- ▶ Služi za merenje apsolutnog vremena.
- ▶ DateTime je struktura
 - ▶ `DateTime t = DateTime.Now;`



- ▶ DateTime klasa može da konvertuje string u **DateTime** object i obrnuto.
- ▶ **Parse** metod klase DateTime koji konvertuje string u objekat DateTime. Primer:

- ▶ *string MyString = "Jan 1, 2002";*
- ▶ *DateTime MyDateTime = DateTime.Parse(MyString);*
- ▶ *Console.WriteLine(MyDateTime);*

- ▶ *using System.Globalization;*
- ▶ *CultureInfo MyCultureInfo = new CultureInfo("de-DE");*
- ▶ *string MyString = "12 Juni 2002";*
- ▶ *DateTime MyDateTime = DateTime.Parse(MyString, MyCultureInfo);*
- ▶ *Console.WriteLine(MyDateTime);*

- ▶ ----- za studente koji žele više da nauče -----

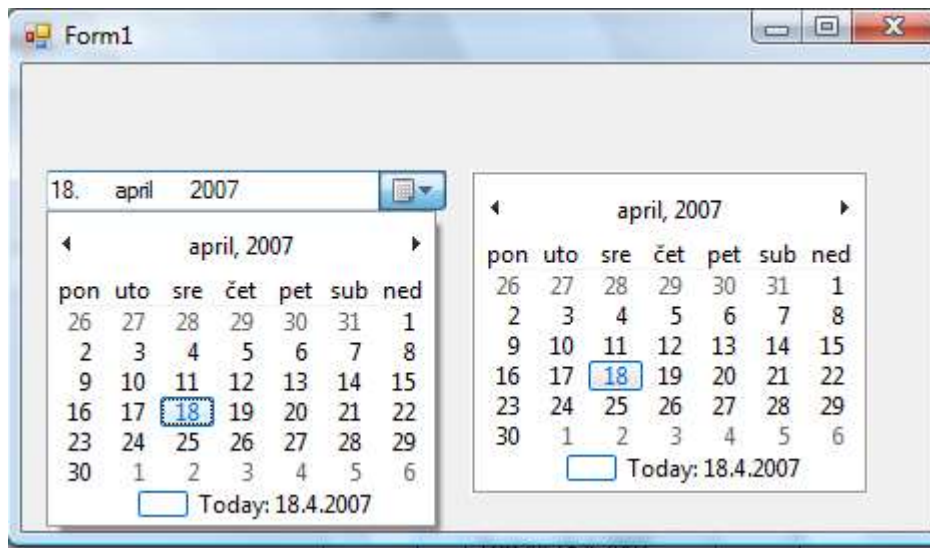
- ▶ *using System.Globalization;*
- ▶ *CultureInfo MyCultureInfo = new CultureInfo("en-US");*
- ▶ *string MyString = "Tuesday, April 10, 2001";*
- ▶ *DateTime MyDateTime = DateTime.ParseExact(MyString, "D", MyCultureInfo);*
- ▶ *Console.WriteLine(MyDateTime);*

- ▶ Formatiranje se obavlja koristeći ToString metodu i string za formatiranje datuma.

- ▶ `DateTime MyDate = new DateTime(2000, 1, 1, 0, 0, 0);`
- ▶ `String MyString = MyDate.ToString("dddd - d - MMMM");`
- ▶ `MyString = MyDate.ToString("yyyy gg");`

Kontrole za rad sa datumom i vremenom

- ▶ **DateTimePicker**
- ▶ MonthCalendar



TimeSpan

- ▶ **TimeSpan** predstavlja vremenski interval ili trajanje mereno kao pozitivan ili negativan broj dana, sati, sekundi ili delova sekundi. **Najkrupnija jedinica je dan.** (Broj dana u mesecu i godini varira).
- ▶ Vrednost **TimeSpan** objekta je broj tick-ova koje predstavljaju taj interval. Jedan tik je 100 nanoseconds a opseg vrednosti **TimeSpan** objekta je od MinValue do MaxValue.
- ▶ **TimeSpan** vrednost može biti predstavljen i kao **[-]d.hh:mm:ss.ff**, gde je minus opcioni znak koji pokazuje negativni interval. **ff** su delovi sekunde.

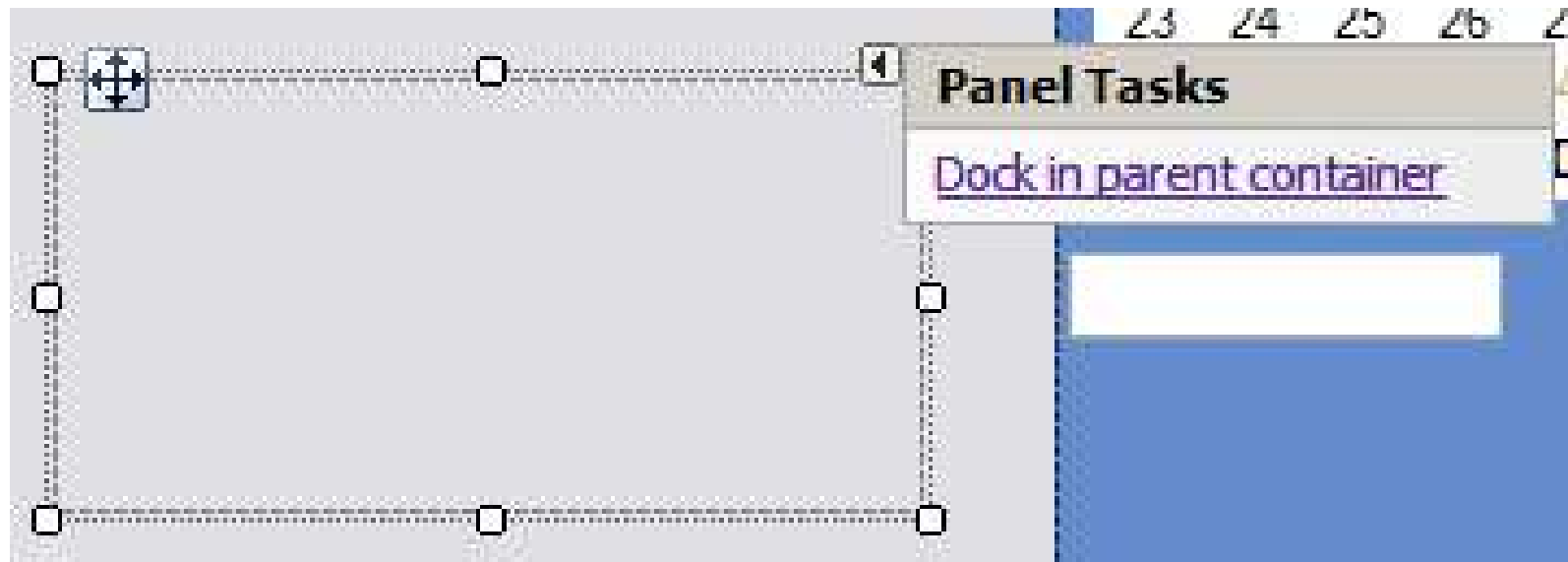
Kontejneri kontrola su...

- ▶ ...objekti koji mogu čuvati i prikazivati više kontrola.
- ▶ Forma je jedan od kontejnera kontrola.
- ▶ Kolekcija **Controls** je tipična za ovu vrstu kontrola.
- ▶ Ova kolekcija nije dostupna preko prozora *Properties* već se puni preko Dizajnera pre pokretanja aplikacije ili se koristi direktno u kodu pri programskom kreiranju kontrola.

Kontejneri kontrola u toolbox-u



Panel



System.Windows.Forms.Panel

- ▶ Inicijalno poprima boju forme ili kontejnera kome pripada.
- ▶ Svaka kontrola na Panelu ima svoja svojstva ali istovremeno su nadjačana istim svojstvima kontejnerske kontrole. Šta ovo znači konkretno?

Na primer: Visible

1. Ako je **Visible** svojstvo panela postavljeno na *false* nevidljive su sve kontrole na ovom **kontejneru** bez obzira na vrednost svojstva Visible kod pojedinačnih kontrola.
2. Ako je Visible = true, onda je vidljivost definisana vrednošću ovog svojstva kontrola koje pripadaju Panelu

Location

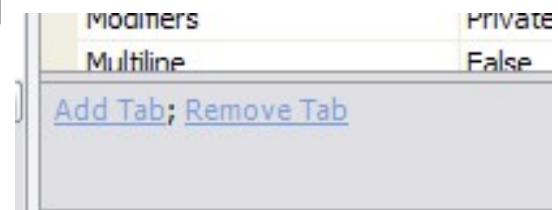
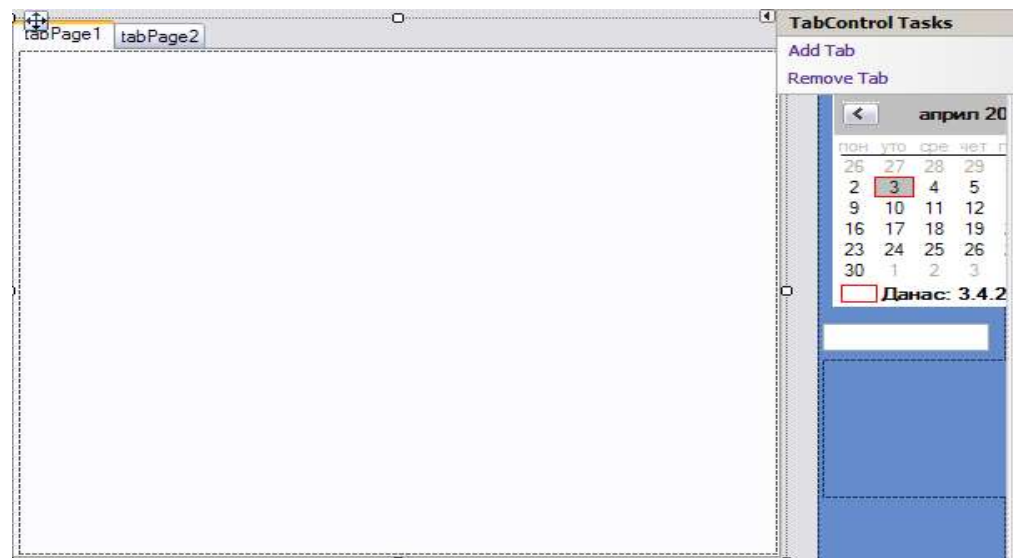
- ▶ Svaki kontejner definiše svoj koordinatni sistem koji je isto orijentisan, ali sa početnom tačnom (0,0) koja je gornji levi ugao te kontejnerske kontrole.
- ▶ Sve kontrole u kontejneru imaju koordinate definisane u odnosu na kontejner.
- ▶ Promena pozicije kontejnera znači promenu pozicije svih kontrola na istom.
- ▶ Kontrola ima svojstvo **Parent** koje pokazuje na kontejner kome ista pripada!

Group box

- ▶ Sličan Panel kontroli.
- ▶ Uobičajeno se koristi pri grupisanju radio dugmadi.
- ▶ Sadrži okvir oko kontrola, a može sadržati i tekst.



Tab kontrola

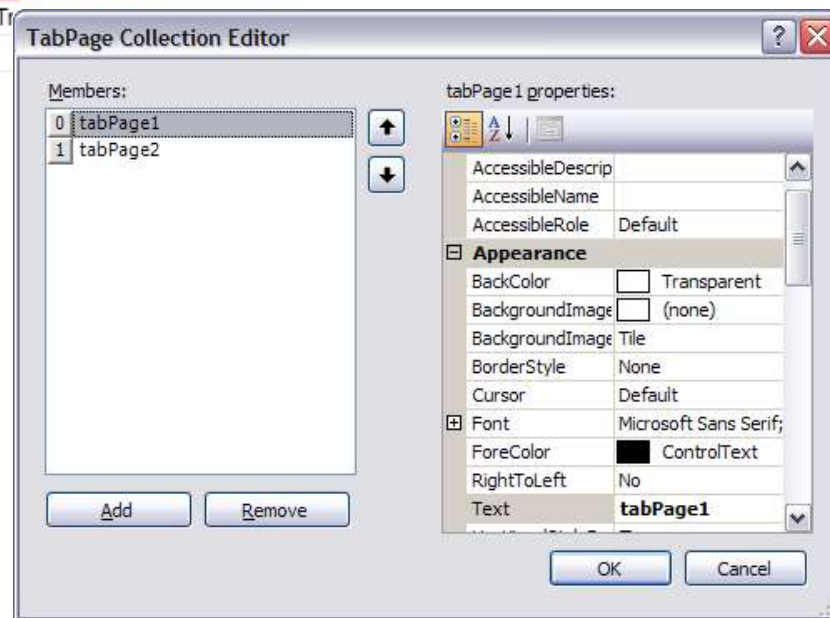


- **TabControl** sadrži *tab pages(stranice)*, tipa TabPage.
- Dodavanje nove stranice se obavlja pomoću svojstva TabPages.
 - Redosled u kolekciji se reflektuje na redosled u kontroli.

Svojsta i metode i događaji od značaja

- ▶ SelectedIndex
- ▶ SelectedTab
- ▶ SelectTab
- ▶ DeselectTab
- ▶ **DOGAĐAJI**
- ▶ Deselecting
- ▶ Deselected
- ▶ Selecting
- ▶ Selected

Size	463; 390
SizeMode	Normal
TabIndex	2
TabPages	(Collection)
TabStop	True
Tag	



Povezivanje kontrola sa podacima



--



--

Složeno povezivanje

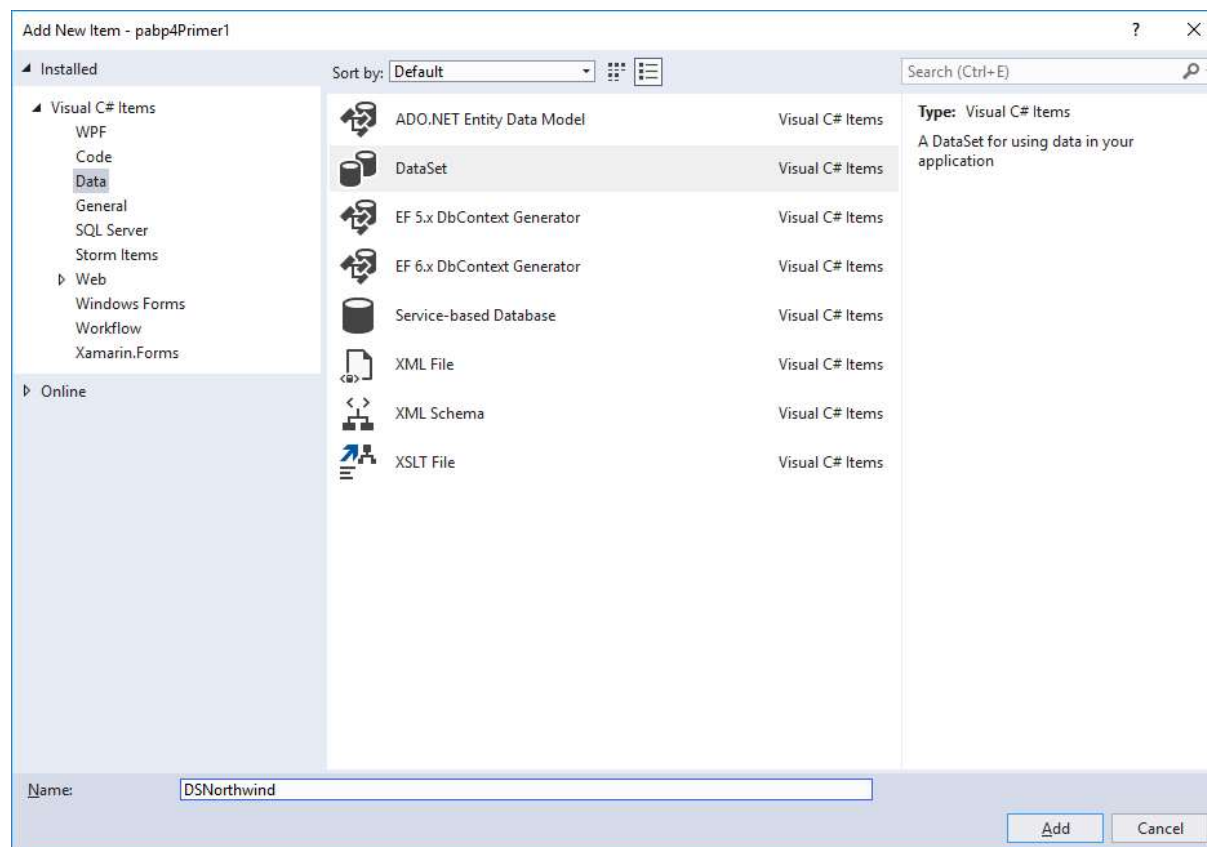
- ▶ ...je jednostavnije
- ▶ Koristi se za kontrole koje prikazuju tabele ili bar jednu kolonu iz tabele. Kontrole su:
- ▶ DataGridView
- ▶ ListBox
- ▶ **ComboBox**

Povezivanje DataGridView kontrole

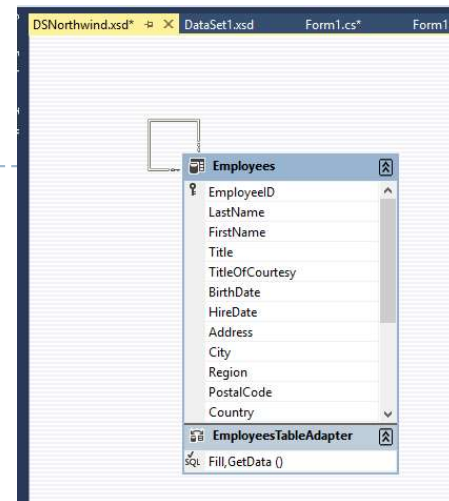
- ▶ Primer: Dodati dve DataGridView kontrole na istu formu i prikazati sve podatke iz tabele Employees koristeći objekte za povezivanje DataBindigs koji su kreirani u fazi dizajniranja i koji su kreirani u kodu.



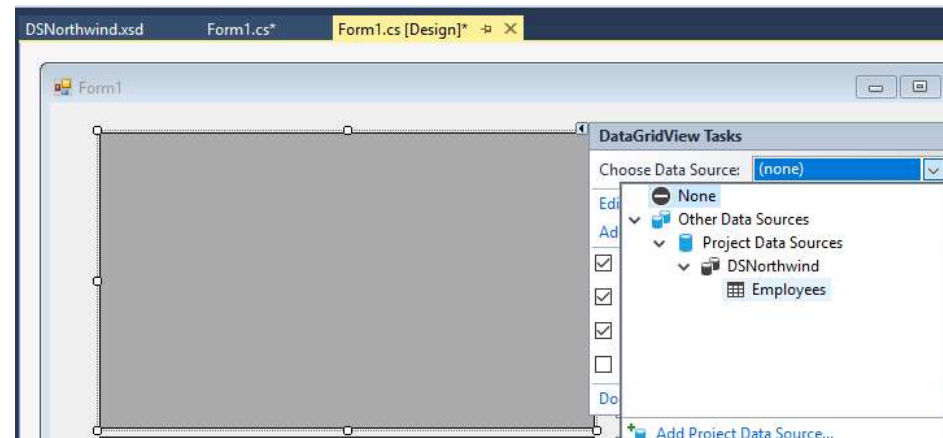
► I. Dodavanje imenovanog DS



- ▶ 2. Zatim u prozoru ServerExplorer naći tabelu Employees za već postojeću konekciju do baze Northwind a zatim je prevući na otvoreni prozor.



- ▶ 3. Na kontroli odabrati Data Source kao na slici.
- ▶ Obratiti pažnju da se formiraju novi objekti i oni se prikazuju na dnu ekrana.
- ▶ Ukoliko se bira postojeći onda se radi samo povezivanje.



dSNorthwind

employeesBindingSource

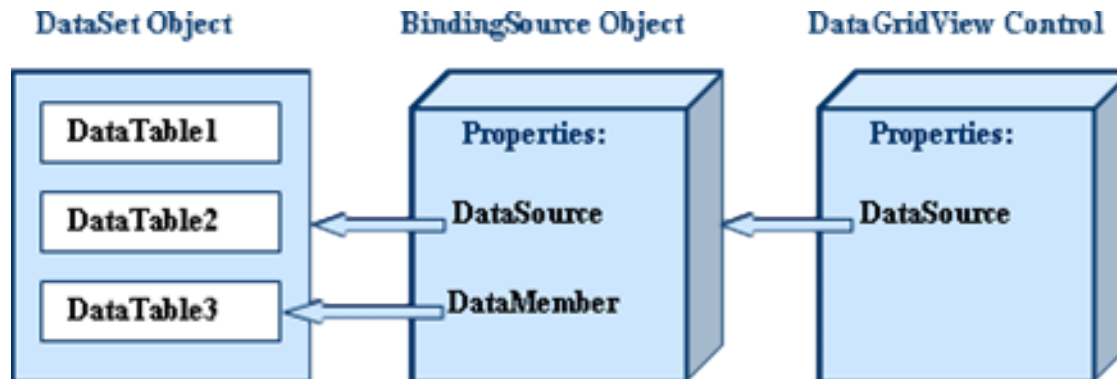
employeesTableAdapter

-
- ▶ 4. Punjenje podataka u DS
 - ▶ Negde u kodu, po potrebi koja zavisi od same aplikacije, treba obezbediti preuzimanje podataka i njihovo skladištenje u DS objekat, tačnije u odgovarajuću tabelu:
 - ▶ `this.employeesTableAdapter.Fill(this.dSNorthwind.Employees);`



BindingSource

- ▶ Mediator između Windows kontrola i nekog *DataSet*-a



BindingNavigator

- ▶ Omogućava korisniku navigaciju kroz podatke



- ▶ VAŽNO

- ▶ Obratiti pažnju da je kontola sada povezana sa objektom **employeesBindingSource**.

- ▶ Ovaj objekat nije DataSet već je povezan sa DataSet objektom.
- ▶ Objekat employeesBindingSource treba da sadrži opis podataka za povezivanje. Kako se podaci prikazuju u DataGridView kontroli koja je prilagođena za prikaz tabele onda i ovaj objekat treba da se odnosi na neku tabelu u DataSet objektu. Zato se za definisanje ovog objekta koriste dva svojstva kao na slici.

DataMember	Employees
DataSource	dSNorthwind
...	

- ▶ Napomena.
- ▶ Ako je DataSource svojstvo povezano sa nekom tabelom u DataSet objektu onda nema potrebe za definisanjem vrednosti u svojstvu DataMember. U slučaju, kao što je gornji, kada je DataSource ceo DataSet onda DataMember treba da sadrži naziv tabele.



U kodu

```
public partial class Form1 : Form{
    BindingSource bs;
    DSNorthwindTableAdapters.EmployeesTableAdapter da;
    public Form1(){
        InitializeComponent();
        bs = new BindingSource(dSNorthwind, "Employees");
        da = new DSNorthwindTableAdapters.EmployeesTableAdapter();
    }
    private void btn2_Click(object sender, EventArgs e)
    {
        da.Fill(dSNorthwind.Employees);
        dataGridView2.DataSource = bs;
    }
}
```

VAŽNO:

Obratiti pažnju da je u kodu kreiran novi BindingSource objekat i da je on povezan sa drugom kontrolom DataGridView. Na ovaj način se nezavisno prikazuju podaci u obe kontrole, ali

Istovremeno su obe kontrole povezane na isti izvor tj. DataSet. To znači da će sve izmene urađene u jednoj kontroli na podacima biti prenete i vidljive istovremeno na drugoj kontroli!

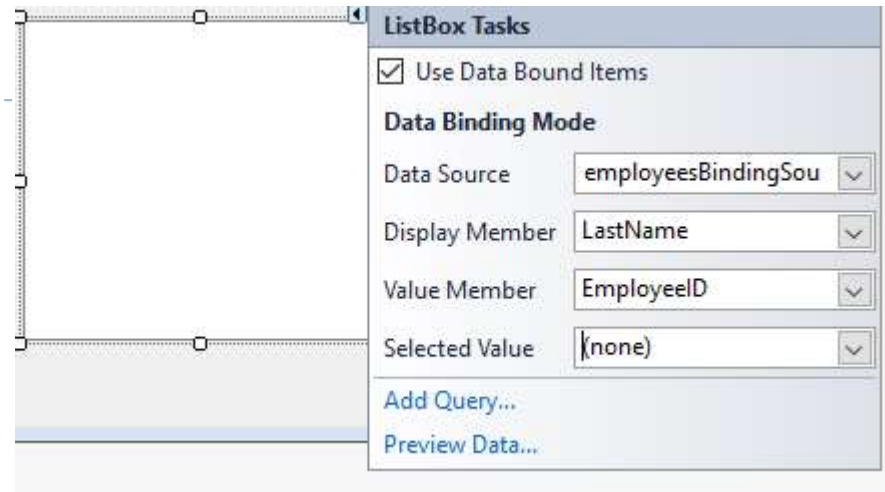


Povezivanje kontrola: ComboBox i ListBox

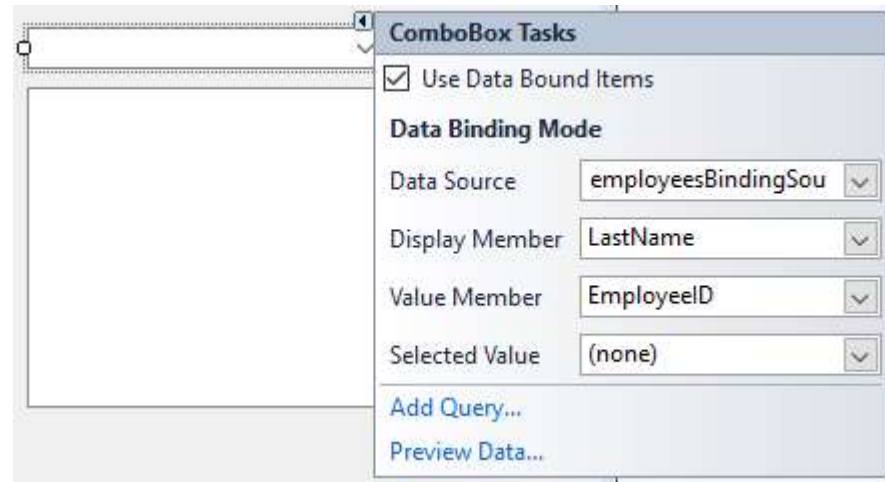
- ▶ ComboBox i ListBox kontrole prikazuju više podataka, ali ne i celu tabelu. Tačnije, ove kontrole prikazuju jednu kolonu u tabeli.
- ▶ Osim toga, ove kontrole ističu jedan red u odabranoj koloni. Ovaj red ima poseban značaj i on je vizuelno drugačije prikazan. U kontroli ListBox taj podatak izgleda selektovan tj. u plavoj boji pozadine, a u ComboBox-u to je vidljivi podatak.
- ▶ Važno:
- ▶ Koristeći istu logiku od prethodnog povezivanje povezivanje treba da obezbedi da se definiše tabela i kolona u tabeli.



► ListBox



► ComboBox



▶ **VAŽNO**

- ▶ Sve kontrole koje su vezane preko istog BindingSource objekta prikazuju sinhronizovano podatke. U funkcionisanju aplikacije to znači da ako se odabere u jednoj od kontrola neki red podataka, isti red postaće odabran u svim kontrolama koje su povezane na isti BindingSource objekat.
- ▶ Dakle, BindingSource objekat sadrži podatak o tekućem tj selektovanom redu.



Filtriranje i sortiranje

- ▶ BindingSource objekat obavlja jednostavno filtriranje i sortiranje podataka pomoću svojstava:
 - ▶ Filter
 - ▶ Sort
- ▶ Vrednost koja se postavlja u ova svojstva odgovara vrednosti koja se navodi u sql iskazima iza ključnih reči:
 - ▶ Where
 - ▶ Order by
- ▶ Na primer:
 - ▶ `dSNorthwindBindingSource.Filter = "LastName like 'B%'"`;
 - ▶ `dSNorthwindBindingSource.Sort = "LastName asc"`;



Pogledati:

- ▶ Svojstva, metode i događaje BindingSource objekta.
- ▶ Obratiti pažnju kom prostoru imena pripada klasa ovog objekta!



Prosto-jednostavno povezivanje - 1

- ▶ Koristi se kada kontrola prikazuje samo jednu vrednost.
- ▶ Izvor podataka za kontrolu je tabelarni i za ovo povezivanje, prema tome
- ▶ Kontrola mora da sadrži informaciju o tabeli, koloni u tabeli i o redu u tabeli izvora podataka tj. tekućem zapisu koji se prikazuje.
- ▶ Na kraju, pitanje je kako se podatak koristi u kontroli?



Prosto-jednostavno povezivanje - 2

- ▶ Podatak koji se koristi u kontroli može da se prikaže kao tekstualna/brojčana vrednost ili u vidu bilo vrednosti koja odgovara nekom svojstvu kontrole, dakle
- ▶ Može se koristiti **bilo koje svojstvo kontrole** za povezivanje sa izvorom podataka.
- ▶ Tipični primeri kontrola i najčešće korišćenih svojstava za povezivanje:
 - ▶ TextBox - "Text"
 - ▶ Label - "Text"
 - ▶ CheckBox - "Checked"
 - ▶ TrackBar - "Value"
 - ▶ ----- itd. "Text"



Prosto-jednostavno povezivanje - 3

- ▶ Povezivanje se ostvaruje preko kolekcije **DataBindings**. Ovo znači da se za jednu kontrolu može vršiti povezivanje više svojstava. Na primer, ukoliko podatak u bazi/izvoru podataka odgovara boji može se koristiti BackColor ili ForeColor... Neka svojstva kontrole mogu biti značajna iako nisu vidljiva, na primer svojstvo “**Tag**”
- ▶ Povezivanje se ostvaruje preko DataSet objekta, ali...
- ▶ ...se najčešće ostvaruje preko posrednika tj. objekta za vezivanje **BindingSource** –a
- ▶ Povezivanje se može uraditi u toku dizajna ili u samom kodu ručno.



Primer

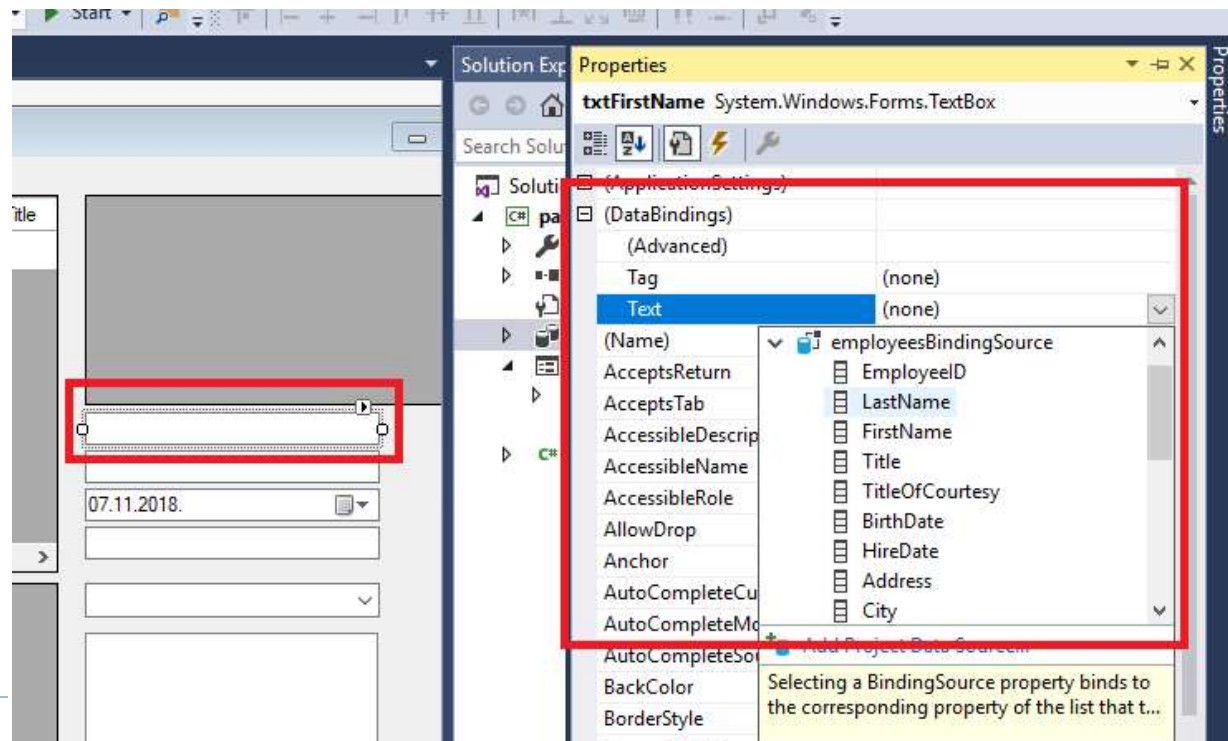
- ▶ Postaviti dve TextBox kontrole za prikaz imena i prezimena. Povezati ih sa odgovarajućim kolonama u tabeli Employees.
- ▶ Pokazati povezivanje fazi dizajna i u kodu.
- ▶ Omogućiti kretanje kroz redove tabele, dodajući kontrolu *BindingNavigator*.



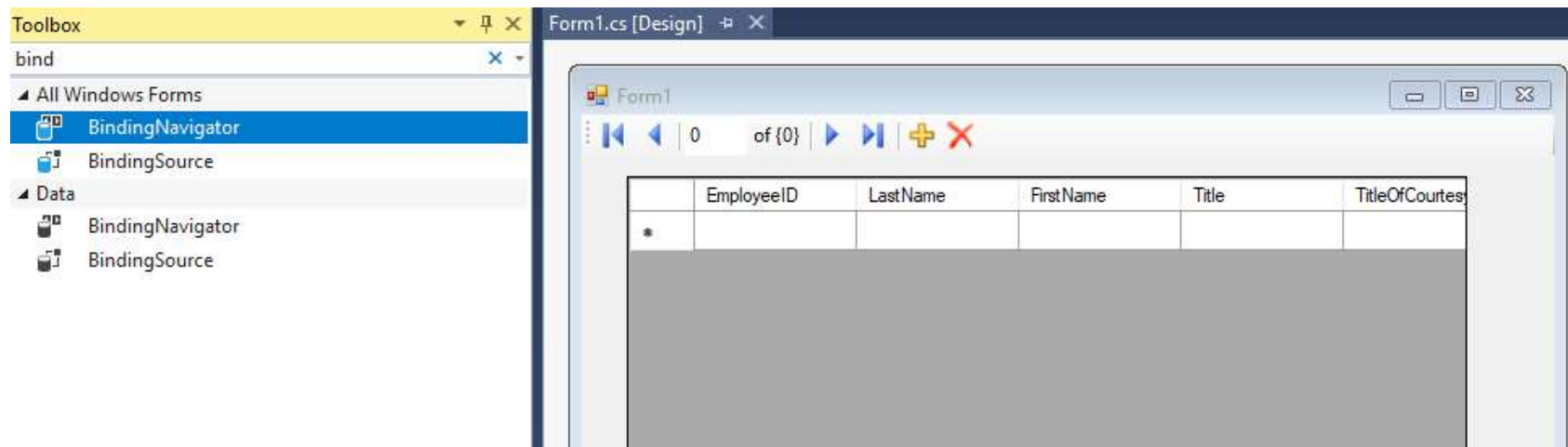
► U kodu:

- `txtFirstName.DataBindings.Add("Text", dsNorthwindBindingSource, "FirstName");`
- `txtLastName.DataBindings.Add("Text", dsNorthwindBindingSource, "LastName");`
- `txtCity.DataBindings.Add("Text", dsNorthwindBindingSource, "City");`
- `dtpHireDate.DataBindings.Add("Value", dsNorthwindBindingSource, "HireDate");`

► U dizajnu:



► Dodavanje BindingNavigator kontrole:



► Povezivanje sa izvorom podataka:

