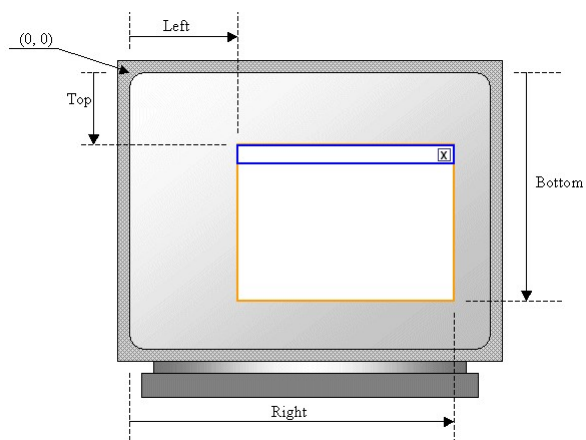


Osnovno o Windows kontrolama

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

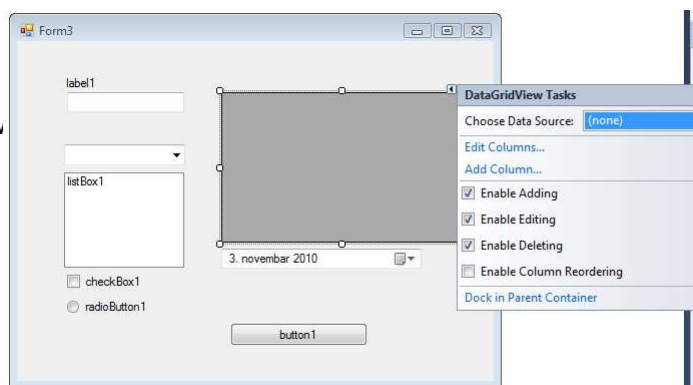
- ▶ *Parent-child relationship*
- ▶ *Ovo je podrazumevana relacija kada se kontrole postajaju i manifestuje se "lepljenjem" kontrole za tu formu. Pomeranjem forme pomeraju se sve kontrole na njoj, tj. sva deca prozori.*

Neka zajednička svojstva Windows kontrola

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

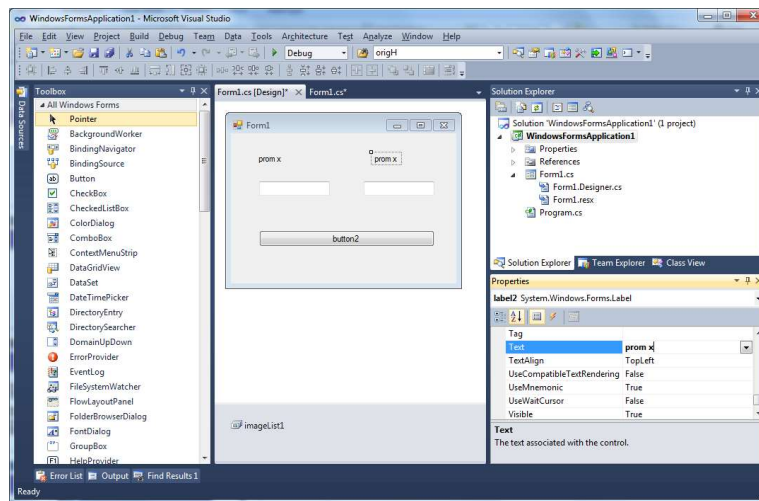
Osnovne kontrole

- ▶ 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.

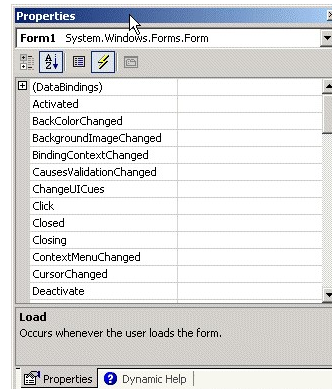


DOGAĐAJI

- ▶ 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**, **System.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, MouseDowns,...

▶ 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)

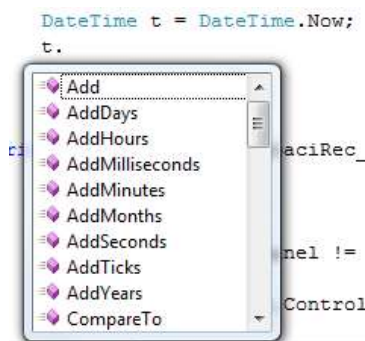
Button okbut = new Button();
okbut.Click += new EventHandler(okbut_Click);
    Press TAB to generate handler 'okbut_Click' in this class

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;

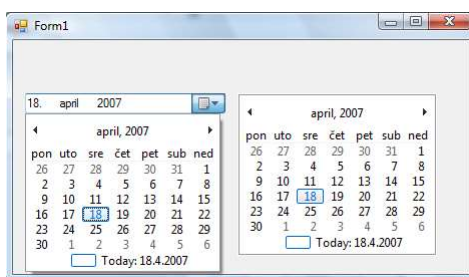


- DateTime klasa može da konvertuje string u **DateTime** object i obrnuto.
- **Parse** metod klase DateTime koji konvertuje string u objekat DateTime. Primer:
 - `string MyString = "Jan 1, 2002";`
 - `DateTime MyDateTime = DateTime.Parse(MyString);`
 - `Console.WriteLine(MyDateTime);`
- `using System.Globalization;`
- `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.

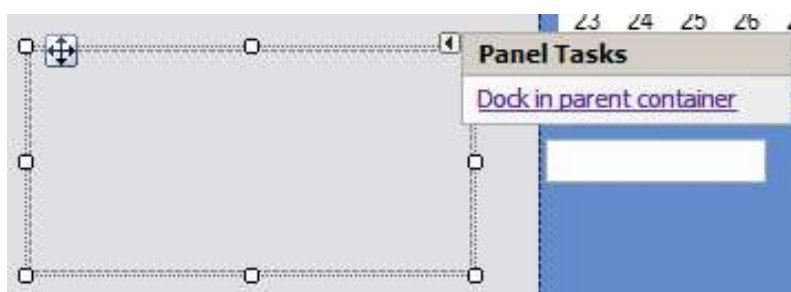
Kontejneri kontrola su...

- ▶ ...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.

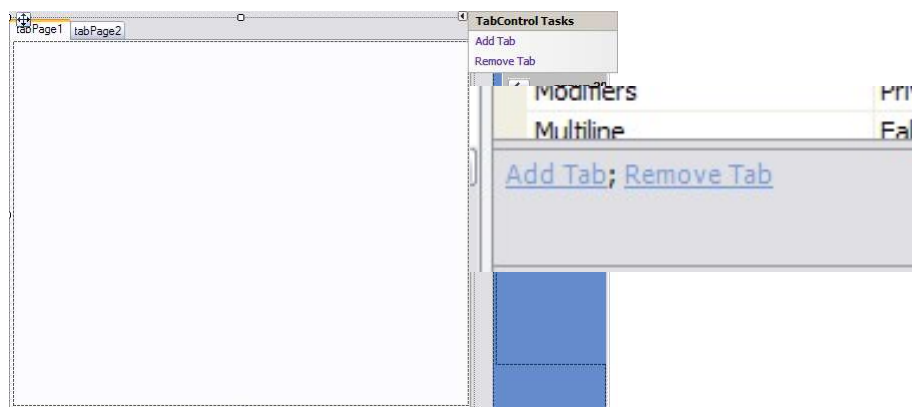
- ▶ Promena pozicije kontejnera znači promenu pozicije svih kontrola na istom.
- ▶ Kontrola ima svojstvo **Parent** koje pokazuje na kontejner kome ista pripada!

Group box

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



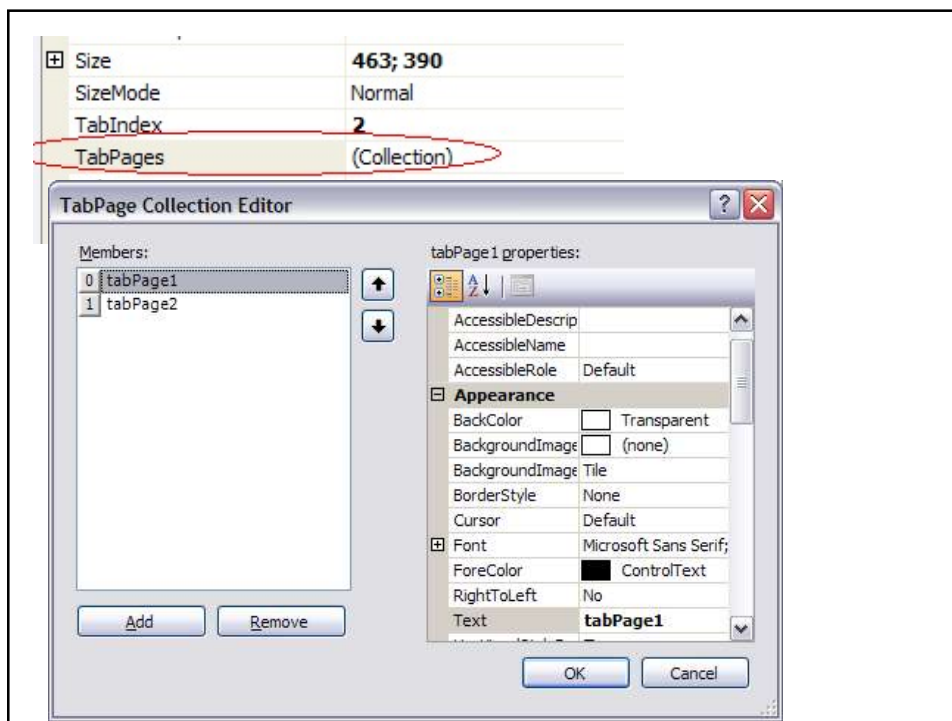
Tab kontrola



- **TabControl** sadrži *tab pages(stranice)*, tipa [TabPage](#).
- Dodavanje nove stranice se obavlja pomoću svojstva [TabPage](#).
 - 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](#)



DataSet - podsećanje

Čemu služi?

- ▶ DataSet je jedan od najbolje osmišljenih i najviše korišćenih objekata ali i je i jedna od važnijih karakteristika Microsoftove .NET platforme.
- ▶ Sličan je tradicionalnom ADO recordsetu, ali i sa bitnim razlikama.
- ▶ **1)DataSet može da čuva rezultate više različitih SQL upita.**
- ▶ **2)Možete koristiti ovaj objekat nezavino od konekcije.**
- ▶ **3)Može se kreirati iz XML fajla .**
- ▶ **4)Možete kreirati XML fajl iz DataSeta.**



Od čega se sastoji?

- ▶ Kolekcija tabela
 - ▶ Tabela
 - ▶ Kolekcija redova
 - Red
 - ▶ Kolekcija kolona
 - Kolona
 - ▶ Ograničenja
 - Ograničenje
- ▶ Veze
 - ▶ Veza



Kako se koristi?

- ▶ Pretpostavimo da imamo konekciju do SQL servera, kreirajmo data adapter na sledeći način:

```
sqlDataAdapter1.SelectCommand.CommandText="SELECT * FROM radnik"
```

```
DataSet ds= new DataSet();
```

```
sqlDataAdapter1.Fill( ds );
```

```
DataTable dt = ds.Tables[0];
```

```
DataColumn dc      = dt.Columns[2];
```

```
                = dt.Columns["naziv"];
```

```
DataRow dr = dt.Rows[2];
```



Vrste DataSet objekata

- ▶ **Tipizirani**
- ▶ **Netipizirani**



Neka svojstva DataSet-a

- ▶ **Tables**
- ▶ **Relations**
- ▶ **HasErrors** – označava da li neki od objekata *DataRow* sadrži grešku
- ▶ **EnforceConstraints** – određuje da li se prilikom izmena poštuju pravila za ograničenja
- ▶ **DataSetName**
- ▶ **CaseSensitive** – da li su poređenja osetljiva na velika slova



Metode objekta DataSet

- ▶ **Clear** – briše sve tabele
- ▶ **Clone** – kopira strukturu DataSet-a
- ▶ **Copy** – kopira i strukturu i podatke
- ▶ **HasChanges** – da li u objektu DataSet postoje izmene koje čekaju.



Filtriranje, sortiranje

- ▶ DataTableCollection dtc = ds.Tables;
- ▶ dtc["Customers"].**Select**("xxxx", "yyyy");
- ▶ pr: xxxx : Country = 'Germany'
- ▶ pr: yyyy: CompanyName ASC



Dodavanje reda

- ▶ DataRow newrow= dt.NewRow();
- ▶ newrow[""] = ;
- ▶
- ▶ dt.Rows.Add(newrow);



Prenošenje izmena

- ▶ UpdateCommand
- ▶ InsertCommand
- ▶ DeleteCommand



Povezivanje kontrola sa podacima

Prosto-jednostavno povezivanje

- ▶ Kada kontrola prikazuje samo jednu vrednost.
- ▶ Izvor podataka za kontrolu je tabelarni.
- ▶ Kontrola treba da sadrži informaciju o redu u tabeli izvora podataka i tekućem zapisu koji se prikazuje.

Prosto-jednostavno povezivanje

- ▶ Može se koristiti bilo koje svojstvo kontrole za povezivanje sa izvorom podataka.
- ▶ Na primer:
 - ▶ TextBox - "Text"
 - ▶ Label - "Text"
 - ▶ CheckBox - "Checked"
 - ▶ TrackBar - "Value"
 - ▶ ----- itd. "Text"

Prosto-jednostavno povezivanje

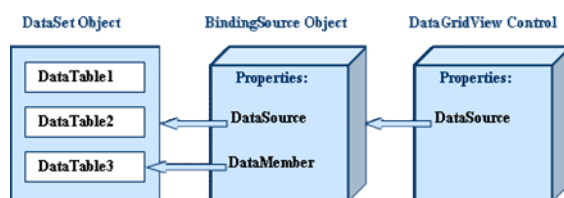
- ▶ Ukoliko podatak u bazi/izvoru podataka odgovara boji može se koristiti BackColor ili ForeColor...
- ▶ Neka svojstva kontrole mogu biti značajna iako nisu vidljiva, na primer svojstvo **"Tag"**

Prosto-jednostavno povezivanje

- ▶ 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.

BindingSource

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

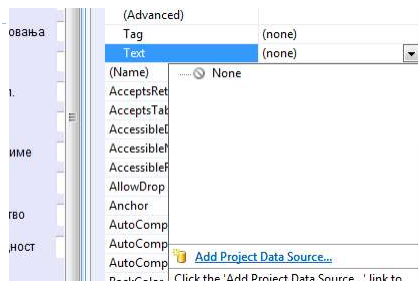


BindingNavigator

- ▶ Omogućava korisniku navigaciju kroz podatke

Prosto povezivanje

- ▶ Postaviti dve TextBox kontrole za prikaz imena i prezimena. Povezati ih sa odgovarajućim kolonama u tabeli Employees.
- ▶ Pokazati povezivanje u fazi dizajna.



```
lblImePrezime.DataBindings.Add("Text", bindingSourceKandidat, "naziv");
```

- ▶ Omogućiti kretanje kroz redove tabele, dodajući 4 dugmeta za pomeraj napred, nazad, na poslednji, na prvi.

Zadatak 2.

- ▶ Dodati 2 "DateTime picker" kontrole i povezati ih sa BirthDate, HireDate.
- ▶ Kontrolu za navigaciju dodati preko dizajnera! Povezati istu za rad sa podacima koji se prikazuju.

Zadatak 3.

- ▶ Dodati kontrolu TextBox i TrackBar i povezati prvu za ime proizvoda a drugu za jediničnu cenu iz tabele *Products*.
- ▶ Povezivanje uraditi u kodu.
- ▶ Dodati i *BindingNavigator* takodje.



Složeno povezivanje

- ▶ ...je jednostavnije
- ▶ Koristi se za kontrole koje prikazuju tabele ili bar 1 kolonu iz tabele
- ▶ DataGridView
- ▶ ListBox
- ▶ **ComboBox**



Zadatak 4

- ▶ Dodati kontrolu ListBox i u njoj prikazati podatak *City* iz iste tabele.
- ▶ Napraviti novi BS preko dizajnera i povezati ga sa istom kontrolom. Šta se događa?



DGV

- ▶ Dodati DataGridView na istu formu i povezati ga sa izvorom podataka.



-
- ▶ **Dodati dugme za brisanje tekućeg zapisa.**
 - ▶ **- brisanje iz DS**
 - ▶ **- brisanje iz baze**
-
- 

-
-
- ▶ **Dodati dugme za ubacivanje novog zapisa.**
 - ▶ **-dodavanje u DS**
 - ▶ **-dodavanje u bazu**
-
- 

Pogledati:

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

