

Visual Basic

- **Vizualno orijentirani programski jezik:** prvo načini sučelje za što ugodniji i intuitivniji rad korisnika (jedan ili više prozora)
- **Događajima vođeno programiranje:** sve što se izvodi potaknuto je nekim događajem, npr. klik miša, izbor nekog elementa u izborniku... Pošto si načinio sučelje, odaberi događaje kod kojih će biti pokrenuti dijelovi programa koji izvode željene akcije.
- **Objektno programiranje:** vizualne i nevizualne komponente koje koristiš su „objekti“: programski elementi koji dobro rade neku funkciju.

Objekti imaju tri skupine značajki:

Svojstva (Properties) – Možeš koristiti i mijenjati njihova svojstva, postavke (boja pozadine - slova, položaj, vrsta slova,...) neka su dostupna za vrijeme dizajniranja (design time), a neka za vrijeme izvođenja programa (run time)

Metode (Methods) – Potprogrami primjenjivi na objektima (vrše neku akciju s objektom)

Događaji (Events) – događaji (jednostruki-dvostruki pritisak lijeve tipke miša na upravljač,...), također potprogrami, ali koji opisuju događaje nad objektima poput klika (Click), dvostrukog klika (DBL Click), pomaka miša (Mouse Move),...

Primjer „objekt automobil“:



Svojstva (Property)

- boja automobila
- trenutna brzina
- preostala količina benzina
- temperatura motora
- svjetla upaljena (da/ne)

Metode (Method)

- pokreni motor
- okreni volan
- promijeni brzinu
- upali svjetla

Događaji (Event)

- pokrenut motor
- nestalo benzina

VB IDE – radna okolina za razvoj programa

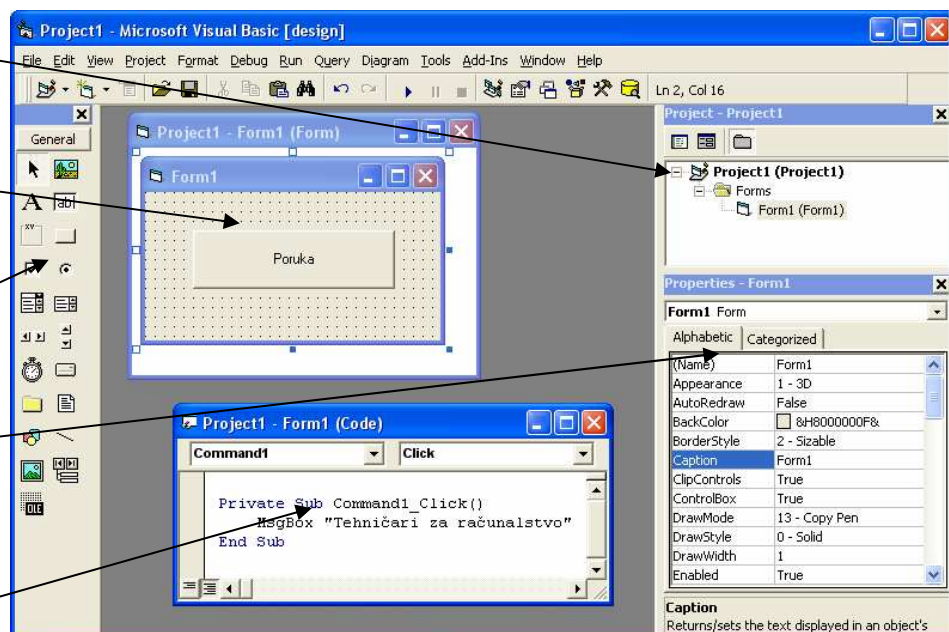
Projekt – aplikacija se sastoji od jedne ili više formi (prozora)

Dijalog editor – mišem razmjestiti željene kontrole po dijalogizima (formama)

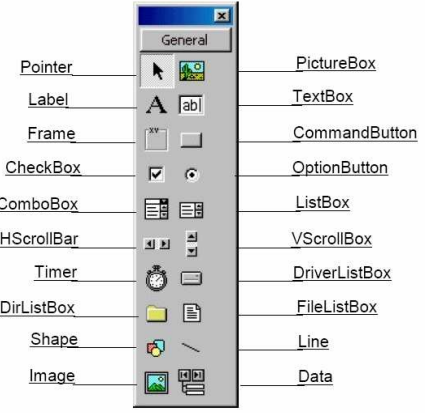
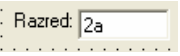

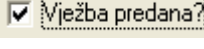

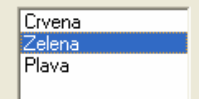
Kontrole – grafički objekti koje surađuju s korisnikom

Svojstva kontrola – svojstva kontrola koja se mogu namjestiti prilikom dizajna

Editor koda – upisuju se dijelovi programa koji se pokreću pri nekom događaju



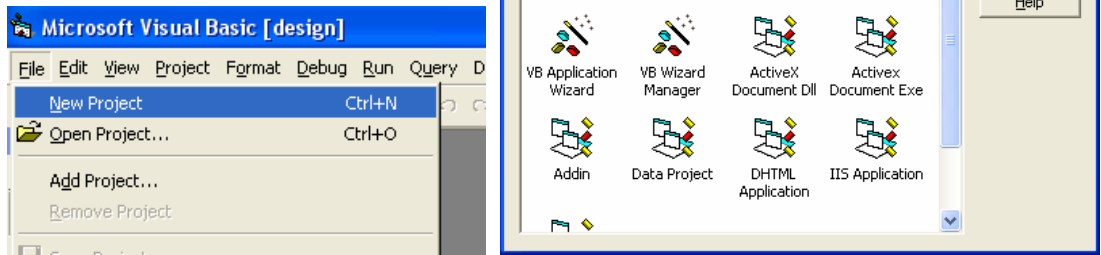
Najvažniji grafički objekti, njihova najvažnija svojstva, metode i događaji

Objekt	Uporaba	Svojstva	Metode	Događaji
	<p>Forma</p> <p>Sadrži ostale kontrole i programski kod koji se izvršava u skladu s događajima vezanim uz formu i kontrole na toj formi.</p>	<p>(Name) – naziv forme koji se koristi u programiranju (nije vidljiv za korisnika)</p> <p>Caption – tekst koji stoji u gornjem rubu prozora</p>	<p>Show – prikaži formu</p> <p>Hide – sakrij formu</p>	<p>Load – događa se pri učitavanju forme, pogodno mjesto za izvršenje inicijalizacije ostalih kontrola</p>
<p>Text Box</p>	<p>Kućica u koju korisnik može upisati neki tekstualni podatak.</p>	<p>(Name) – kao kod forme</p> <p>Text – tekst koji se nalazi u kućici</p> <p>Enabled – korisnik smije/ne smije mijenjati sadržaj teksta</p> <p>MultiLine – tekst se može pisati u više linija</p>	<p>SetFocus – za kontrolu na koju se proslijeđuju pritisnute tipke se kaže da ima fokus. SetFocus se može koristiti za skretanje pažnje korisnika na kontrolu u koju je nešto unijeti.</p>	<p>Change – registrira svaku promjenu teksta (nakon svakog slova)</p> <p>Validate – pokreće se pri izlasku iz kućice. Korisna je za provjeru, da li je korisnik unio valjane podatke. Ukoliko nije, može se onemogućiti izlaz iz kućice.</p>
<p>Label</p>	<p>Korisnik ne može mijenjati tekst koji je programer upisao. U labelu se obično upisuju upute, što znači text box do nje:</p> 	<p>(Name)...</p> <p>Caption – tekst koji piše u labeli</p>		
<p>Gumb</p>	<p>Kontrola kojom korisnik obično pokreće željenu akciju.</p> 	<p>(Name)...</p> <p>Caption – tekst na gumbu</p>	<p>SetFocus...</p>	<p>Click – korisnik je mišem ili tipkovnicom pritisnuo gumb. «Pokreni ono što piše na gumbu»</p>
<p>Check Box</p>	<p>Kontrola koja ima dva stanja označeno (checked) i neoznačeno (unchecked).</p> 	<p>(Name)...</p> <p>Caption – tekst ispisan pored kućice za izbor, poput labele</p> <p>Value – jedna od vrijednosti: 0 = neoznačeno 1 = označeno 2 = onemogućeno</p>	<p>SetFocus...</p>	<p>Validate – kao i kod Text Box-a pogodno mjesto za kontrolu, npr. da li je trenutni izbor u Check Box-u u skladu s dugim kontrolama</p>
<p>Option Box</p>	<p>Kao i Check Box posjeduje dva stanja ali se obično jednoj grupi koristi više Option Box-eva:</p> 	<p>(Name)...</p> <p>Caption – tekst ispisan pored kućice za izbor, poput labele</p> <p>Value – jedna od vrijednosti: False = neoznačeno True = označeno. U jednoj je grupi uvijek samo jedan option box označen.</p>	<p>SetFocus...</p>	<p>Validate...</p>
<p>List Box</p>	<p>Popis s kojeg korisnik može izabrati jedan redak:</p> 	<p>(Name)...</p> <p>List(i) – vrijednost teksta koji se nalazi u i-tom retku.</p> <p>ListCount – broj redaka u List Box.</p> <p>ListIndex – indeks trenutno izabranog retka.</p>	<p>SetFocus...</p> <p>AddItem – dodavanje novih redaka u List Box.</p> <p>Clear – brisanje svih redaka iz LB.</p>	<p>Validate...</p>

Projekt

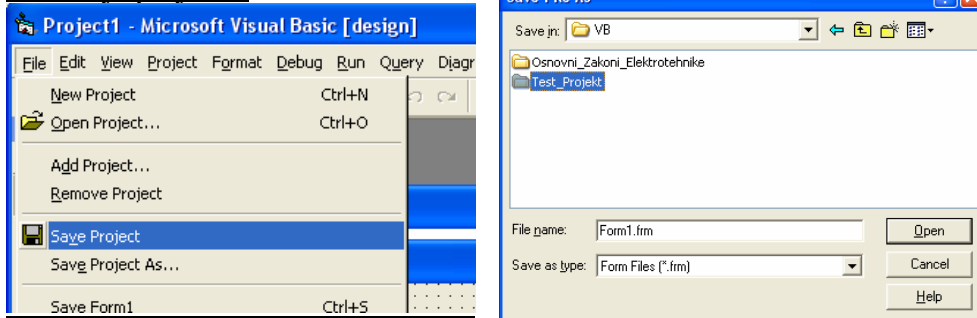
- jedna ili više formi i programskih modula
- svaka forma smještena u svoju datoteku
- projekt ih okuplja u jedinstvenu aplikaciju

Kreiranje projekta:



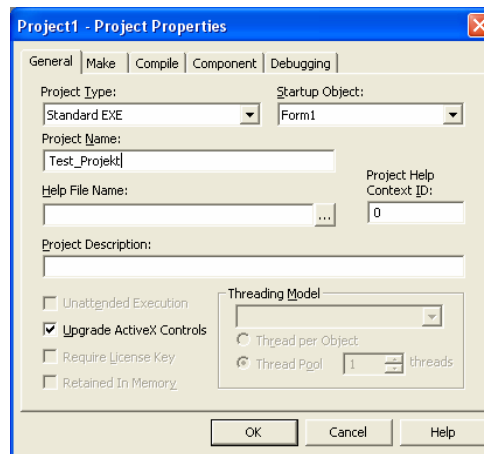
Izbornikom File/New Project, pojavljuje se dijalog „New Project“, izabrati Standard Exe, Ok... VB kreira projekt i jednu formu koja će biti prva pokrenuta pokretanjem programa.

Snimanje projekta:



Pošto se projekt sastoji od više datoteka, preporučljivo je za svaki projekt kreirati novi direktorij (ovdje Test_Projekt), pozicionirati se u njega i snimiti sve datoteke koje se traže. Za vrijeme rada periodično snimati načinjeno. U dijalogu „Project Properties“ mogu se namjestiti svojstva projekta:

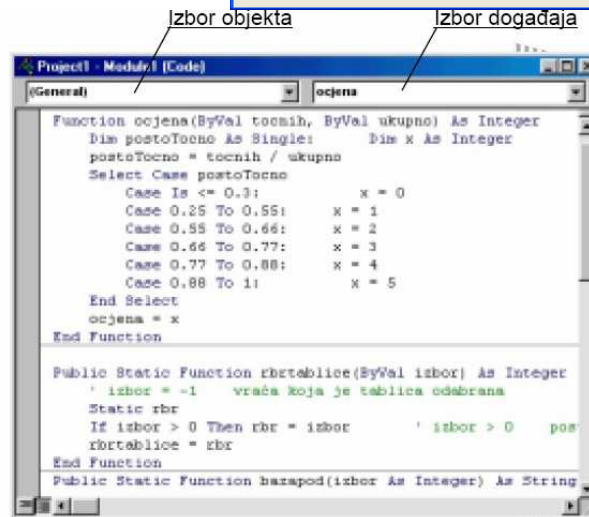
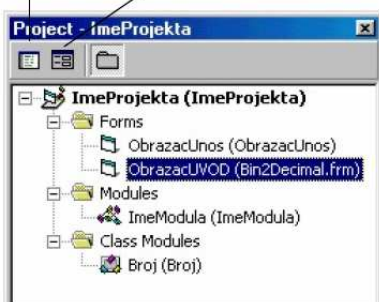
Project Name – izabrati ime smisljeno za predviđenu namjenu aplikacije



Startup Object – u slučaju aplikacije s više formi izabrati formu koja će se prva pokrenuti

(View Code)
Rad s naredbama.
događajima...

(View Object)
Rad s obrascem i
objektima na njemu



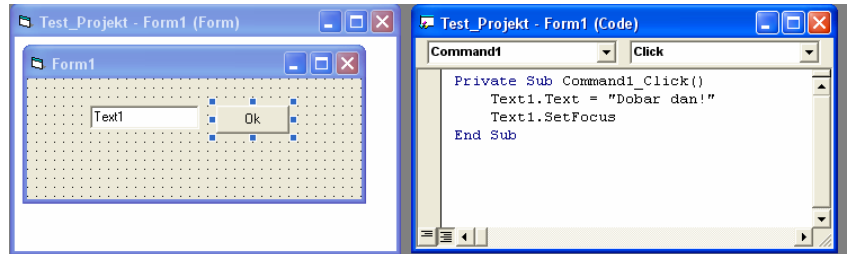
Programiranje

- programski kod grupiran je u odsječke, a svaki odsječak je reakcija na određeni događaj
- naziv odsjeka tj. procedure za obradu događaja:
(Name)_Događaj
Private Sub Command1_Click
- kod je «druga strana forme» (ona prva je grafički dizajn)
- VB je za svaku kontrolu smještenu na formu generirao varijablu preko koje se u kodu pristupa njezinim svojstvima i metodama
- pristupanje svojstvima i pokretanje metoda objekata tj. kontrola pomoću točke:
(Name).Svojstvo

Text1.Text = "Dobar dan!"

(Name).Metoda

Text1.SetFocus – Vraća pokazivač miša na tekst



UPRAVLJAČKI GUMB (COMMAND BUTTON)

Ovaj gumb se koristi za početak, prekid ili kraj nekog postupka. Dobije se ako kliknemo na njega i postavimo ga na formu držeći lijevu tipku miša i povlačeći na željenu veličinu ili dvostrukim klikom ili klik na njega i tipka ENTER (ako je dugme označeno i ako je postavka Default vrijednosti True)

NEKA SVOJSTVA UPRAVLJAČKOG GUMBA

NAME (Ime varijable koja čuva objekt tipa Form)

Ne može se mijenjati tijekom izvođenja programa jer se preko nje pristupa objektu tijekom izvođenja.

CAPTION (Postavlja se tekst u naslovu (Title Bar) obrasca tipa Form ili MDIForm)

Sintaksa: object.caption=String

String – niz znakova koji predstavljaju tekst u naslovu objekta. Predugi naslov se reže.

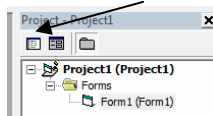
Kod stvaranja novog objekta, automatski se stvara i naslov koji predstavlja naziv tipu objekta.

Zadatak 1.

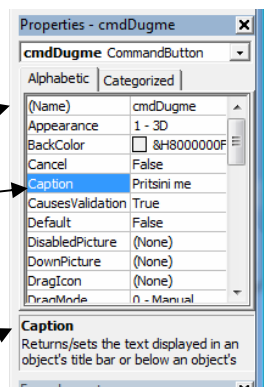
Napisati programski kod koji će, kad se stisne gumb pisati *Otpusti me*, a kad nije stisnuto *Pritisni me*.

- a) Stvori upravljački gumb tipa **CommandButton** i nazovi ga **cmdDugme**. Postavi mu svojstvo **Caption** na: **Pritisni me**.

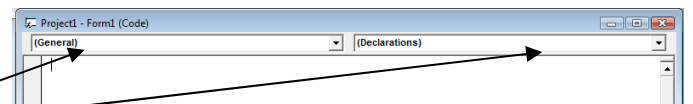
- b) Sada trebamo rad s naredbama odnosno događajima (klik na ikonicu)



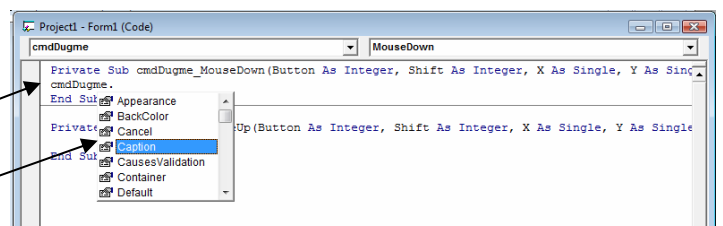
Objašnjenje za pojedino svojstvo



- c) U prozoru za upis koda na padajućem lijevom izborniku izaberite **cmdDugme**, a u desnom **Mouse Up** (otpuštena tipka miša), a potom **Mouse Down** (stisnuta tipka miša).



- d) Za **Mouse Up** postavi Svojstvo **Caption** na **Pritisni me**. Treba upisati **cmdDugme**. Onda se pojavi izbornik sa svojstvima gdje birate **Caption**, jer to svojstvo mijenjate.



- e) Za **Mouse Down** postaviti *Svojstvo Caption* obrasca na **Otpusti me** (znači kad stisnete gumb i držite stisnuto, pisat će **Otpusti me**, a kad otpustite gumb, pisat će **Pritisni me**).

Sintaksa:

```
Private Sub cmdDugme_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    cmdDugme.Caption = "Otpusti me"
End Sub
```

```
Private Sub cmdDugme_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    cmdDugme.Caption = "Pritisni me"
End Sub
```

OBJEKT TEXTBOX

Objekt TextBox prikazuje podatke unesene u vrijeme oblikovanja (design time), unesene od korisnika ili pridružene programski za vrijeme izvođenja programa (run-time). Podatak je tipa String.

NEKA SVOJSTVA TEXTBOX-a

NAME (*Ime varijable koja čuva objekt tipa TextBox*)

Ne može se mijenjati za vrijeme izvođenja programa).

MULTILINE (*Vraća ili postavlja vrijednost koja određuje je li tekst moguće staviti u više redaka*).

True – omogućeno, False (inače) – nije omogućeno

Za prikaz teksta u više redaka treba postaviti postavku MultiLine na True. Traka za navođenje bit će prikazana na ovom objektu uvijek kad je svojstvo MultiLine postavljeno na True, a svojstvo ScrollBars na bilo što osim None(0).

TEXT (*Vraća ili postavlja znakovni niz u objekt (tj. u TextBox-u)*).

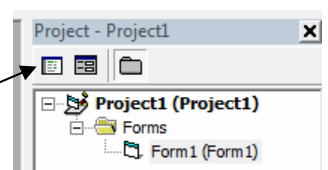
Sintaksa: object.Text="Tekst koji želite"

```
Private Sub Text1_Change()
    Text2.Text=Lcase(Text1.Text)           *Prikaz teksta malim slovima
    Text3.Text=Ucase(Text1.Text)         *Prikaz teksta velikim slovima
End Sub
Private Sub cmdBrisi_Click ()
    Text1.Text=""                         *Brisanje Teksta
End Sub
```

Zadatak 2.

Treba napraviti programski kod koji će, kad se klikne gumb **OK** napisati u TextBox-u *Dobar dan!*, a kad se mišem prijeđe preko gumba **OK** će u drugom TextBox-u napisati *Laku noć!*

- Stvori upravljački gumb tipa **CommandButton** i nazovi ga **cmdDugme**.
- Postavi mu *svojstvo Caption* na: **OK**.
- Stvori **Text Box** i postavi ga iznad upravljačkog gumba.
- Sada trebamo rad s naredbama odnosno događajima (klik na ikonicu)



- e) U prozoru za upis programa na padajućem lijevom izborniku izaberite **cmdDugme**, a u desnom **Click** (već piše), jer se događaj dešava kad kliknete.
- f) Sada moramo napisati programski kod koji kad se klikne na Gumb **OK** u Text Box-u piše *Dobar Dan!*

Sintaksa: `Text1.Text="Dobar dan!"`

- g) Pokrenimo program tipkom **F5** ili na traci s izbornicima izaberemo **RUN->Start** ili ikonu ▶

- h) Na istoj formi dodaj **Text Box** u koji će se upisati *Laku noć!* kada miš pređe preko gumba OK. Kad pokrenete program treba izgledati ovako:



Sintaksa: `Text1.Text="Laku noć!"`

OBRAZAC (FORM)

Obrazac je osnovni objekt na koji se postavljaju drugi objekti. Otvaranjem novog projekta (programa) automatski se otvara novi prazan početni obrazac.

NEKA SVOJSTVA OBRASCA (FORM-e)

NAME (*Ime varijable koja čuva objekt tipa Form*).

Ne može se mijenjati tijekom izvođenja programa, jer se preko nje pristupa objektu za vrijeme izvođenja.

CAPTION (*Postavlja se tekst u naslovu (TitleBar) obrasca tipa form ili MDIForm*)

BACKCOLOR (*vraća ili postavlja boju pozadine objekta*)

WINDOWSTATE (*vraća ili postavlja veličinu forme, 0-Normalna veličina forme, 1-Minijatura, 2-najveća veličina (cijeli prozor)*)

JOŠ NEKA SVOJSTVA UPRAVLJAČKOG GUMBA (COMMAND BUTTON)

VISIBLE (*Vraća vrijednost True (Default) ili False, da li je objekt vidljiv ili ne.*)

Zadatak 3.

Svojstvo *Caption* obrasca (Form-e) postavi na **Sakrij me**. Na obrazac postavi dva upravljačka gumba (**CommandButton**) i promijeni im imena u **cmdDugme1** i **cmdDugme2**. Svojstvo *Caption* treba postaviti na **Sakrij me**. Treba napisati programski kod za **cmdDugme1**: kada se klikne prvi gumb, on nestane, drugo ostane vidljivo, a kada se stisne drugo dugme, ono nestane, a postane vidljivo prvo dugme.

Sintaksa:

```
Private Sub cmdDugme1_Click()           *događaj klika na prvo dugme
    cmdDugme1.Visible=False            *sakrij prvo dugme
    cmdDugme2.Visible=True             *otkrij drugo dugme
End Sub
Private Sub cmdDugme2_Click()           *događaj klika na drugo dugme
    cmdDugme1.Visible=True            *otkrij prvo dugme
    cmdDugme2.Visible=False           *sakrij drugo dugme
End Sub
```

Zadatak 4. (probajte prvo sami)

- Boju forme treba staviti na svijetlo plavu (kada klineš na BackColor i izabereš boju, pojavi se broj &H00FFFF80 – zapiši to i zapiši koji je za bijelu boju jer će ti trebati.
- Treba na formu staviti 2 upravljačka gumba, jedan nema naslov, drugi ima *Klikni ovdje*.
- Kada klikneš na drugi upravljački gumb (*Klikni ovdje*), forma ide na cijeli zaslon, a kad klikneš na prvi upravljački gumb (koji nema naslov), forma se vrati u normalu.
- Kada mišem pređeš preko drugog upravljačkog gumba (*Klikni ovdje*) forma mijenja boju u bijelu, a kad pređeš mišem preko forme vraća se u svijetlo plavu.

Da bi si olakšali zadatak, možemo uvesti konstante (**Constant**). Konstante su veličine kojima ćemo mi zadati neku vrijednost i kroz cijeli zadatak kada koristimo tu konstantu vrijednost ostaje ista. U matematici bi to bio π . Tako ćemo mi na **početku** uvesti konstantu svijetloplava i bijela.

Sintaksa: `Const svijetloplava=&H00FFFF80`
`Const bijela=&H00FFFFFF`

I umjesto ovog programskog koda za promjenu boje forme:

```
Private Sub Command4_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Form1.BackColor = &HFFFFFF
End Sub
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Form1.BackColor = &HFFFF80
End Sub
```

Koristiti ovaj programski kod:

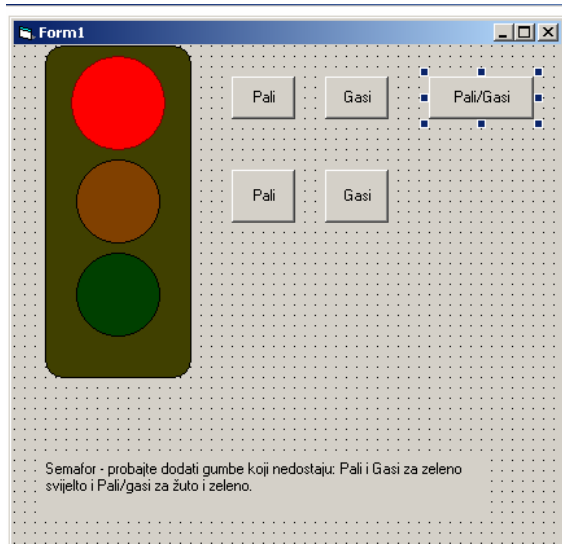
```
Private Sub Command4_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Form1.BackColor = bijela
End Sub
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Form1.BackColor = svijetloplava
End Sub
```

Zadatak 5. (probajte prvo sami)

- Umetnite krug (Shape) na formu i obojite ga u tamnocrvenu
- Postavite na formu 2 upravljačka gumba. Jednom stavite naslov Pali, drugom Gasi.
- Koristiti konstante, prvu nazovi tamnocrvena, drugu svijetlocrvena.
- Klikom na gumb pali krug postaje svijetlocrven, a klikom na gumb gasi postaje tamnocrven

Zadatak 6.

- Nastavite prethodni zadatak tako da dobijete semafor.
- Umetnite još 2 kruga ispod prethodnog na formu i obojite jednog u žutu, a drugog u tamnozeleno
- Postavite na formu po 2 upravljačka gumba za svaki krug. Jednom stavite naslov Pali, drugom Gasi.
- Koristiti konstante, dajte im imena koja imaju smisla.
- Klikom na gumb pali krug postaje upaljen (svijetao), a klikom na gumb gasi postaje ugašen (taman)
- Uokvirite krugove tako da izgleda kao semafor. Pogledaj sliku.



LABEL

Label je grafički objekt namijenjen prikazu teksta kojeg korisnik programa ne može izravno promijeniti. Najčešće se koristi za prikaz stanja programa tijekom izvođenja, kao i za označavanje ostalih objekata, kao npr. TextBox, Picture,... koji nemaju svojstvo Caption.

NEKA SVOJSTVA OBJEKATA LABEL

NAME (ime varijable koje čuva objekt Label. Ne može se mijenjati tijekom izvođenja programa)

FONTSIZE (veličina slova)

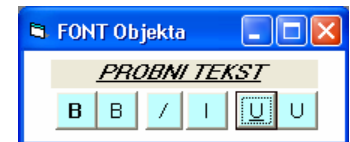
ALIGNMENT (poravnanje teksta: 0-lijevo, 1-desno, 2-centar)

BACKCOLOR (vraća ili postavlja boju pozadine objekta)

BACKSTYLE (vraća ili postavlja vrijednost kojom određujemo je li pozadina objekta transparentna (0-prozirna), ili neprozirna (1-boja je označena sa BackColor))

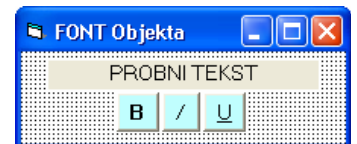
Zadatak 7.

- Promijeni naslov obrasca u: FONT objekta (pogledaj sliku)
- Na obrazac postaviti 6 kontrolnih gumba, nazivi su redom: cmdPodebljaj, cmdNePodebljaj, cmdNakosi, cmdNeNakosi, cmdPodcrtaj, cmdNePodcrtaj, i naslova: **B**, **B**, **/**, **I**, **U**, **U**
- Na obrazac umetnuti Label naslova PROBNI TEKST
- Kad se klikne gumb **B** tekst u Label-u se treba podebljati, tako i za **I** nakositi i za **U** podcrtati. Kad se klikne na preostala tri gumba treba se poništiti podebljano, nakošeno ili podcrtano.



Sintaksa (dio):

```
Private Sub cmdPodebljaj_Click()  
    Label1.FontBold = True  
End Sub  
Private Sub cmdNePodebljaj_Click()  
    Label1.FontBold = False  
End Sub
```



Uz pomoć logičkog operatora NOT negira se postojeće stanje podebljavanja, ukošavanja ili podcrtavanja. Uz pomoć njega zadatak se puno brže i jednostavnije riješi. Jer, ako je podebljano, onda NOT podebljano, ako nije podebljano, onda je NOT nije podebljano (podebljano). Time nam ne trebaju 6 upravljačkih gumba i 6 redova kodova, nego samo tri.

Sintaksa za Bold (isto je za nakošeno i podcrtano):

```
Private Sub cmdPodebljaj_Click()  
    Label1.FontBold = NOT(Label1.FontBold)  
End Sub
```

Možemo dodati programski kad se klikne na slovo **B** podeblja se tekst i promijeni izgled slova **B** i obratno. Tako i za nakošeno i za podcrtano. Probajte u istom programu koji ste radili, umetnuti kod:

Sintaksa: `cmdPodebljaj.FontBold=NOT(cmdPodebljaj.FontBold)`

Zadatak 8.

Prethodnom programu treba dodati Labelu gdje će pisati **Veličina slova** (Font), **TextBox** gdje će se upisivati željena veličina slova, i upravljački gumb koji će imati naslov **Promijeni**, a koji postavlja probni tekst na upisanu veličinu. Vidi sliku.



Zadatak 9.

Ispod prethodnog dodaj tri upravljačka gumba. Prvi ima naslov **L**, kad se na njega klikne probni tekst se lijevo poravna. Drugi ima naslov **S**, kad se na njega klikne, tekst se poravna po sredini i treći ima naslov **D**, a kad se na njega klikne, tekst se poravna desno.

Zadatak 10.

Napravi program koji će sate, minute i sekunde pretvarati u sekunde. Forma treba sadržavati pet Labela, *Unesi sate:*, *Unesi minute:*, *Unesi sekunde*, *Broj sekundi:* i jednu praznu u koju će se ispisivati sekunde. Centrirajte naslove, promijenite boju slova i podebljajte ih te promijenite veličinu fonta na 16. Te tri **TextBox-a** u koje će korisnik upisivati tražene podatke. Njima promijenite veličinu slova na 17, podebljajte i centrirajte tekst i upravljački gumb naslova *Pretvori* koji će na klik vršiti pretvaranje.

Zadatak 11.

Napravi program koji će zbrajati dva broja. Forma treba sadržavati četiri Labela, *Unesi prvi broj:*, *+*, *Unesi drugi broj:*, *Zbroj iznosi:* i jednu praznu u koju će se ispisivati rezultat. Centrirajte naslove. Te dva **TextBox-a** u koje će korisnik upisivati tražene podatke. Njima promijenite veličinu slova na 14, podebljajte i centrirajte tekst i upravljački gumb naslova *Izračunaj* koji će na klik vršiti izračun.

NAPOMENA: Ovdje ne možemo napisati samo `Text1+Text2` jer je svojstvo tekst tipa String i kao rezultat dobijemo spojena dva teksta. Umjesto toga ćemo koristiti funkciju **VAL**.

FUNKCIJA VAL vraća broj u znakovnom nizu i prepoznaje samo točku kao znak koji odvaja cijele brojeve od razlomljenih.

Sintaksa: `Label5.Caption = Val(Text1) + Val(Text2)`

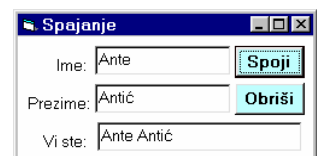
Zadatak 12.

Treba izračunati opseg i površinu, ako korisnik unese

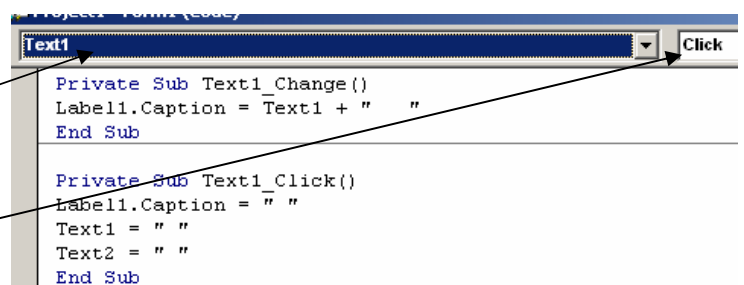
- stranicu kvadrata
- stranice pravokutnika
- stranice trokuta i visinu na stranicu b
- Izračunati opseg, broj dijagonala iz jednog vrha i ukupan broj dijagonala, te zbroj unutrašnjih kutova pravilnog mnogokuta, ako korisnik unese broj stranica i duljinu stranice pravilnog mnogokuta.

Zadatak 13.

Treba spojiti ime i prezime nakon što ih korisnik unese pojedinačno. Vidi sliku.



Staviti prazna mjesta između imena i prezimena (" "). Možete napraviti kao na slici upravljačke gumbke koji će spajati ime i prezime, te brisati ILI umjesto gumba *Spoji* možete koristiti događaj (Event) **Change** za **TextBox**. Dva puta kliknemo na **TextBox** ILI U prozoru za upis koda na padajućem lijevom izborniku izaberite **Text1**, a u desnom **Change**. Napisati što će se desiti u naslovu labela kad se nešto promijeni u **TextBox-u**. Umjesto gumba *Obriši* možete koristiti događaj **Click** za prvi **TextBox**. Za prikaz rezultata koristite Labelu.



Zadatak 14.

Kao u prethodnom zadatku treba spojiti ime i prezime, ali ih treba napisati velikim tiskanim slovima. Koristiti UCase (za pisanje velikih slova). *LCase (za pisanje malih slova)

TIMER

Omogućava izvođenje vremenski uvjetovanih dijelova programa.

NEKA SVOJSTVA OBJEKTA TIMER

NAME (ime varijable koje čuva objekt tipa Timer. Ne može se mijenjati za vrijeme izvođenja programa).

ENABLED (logička vrijednost (True ili False). Vraća ili postavlja mogućnost pokretanja vremenski uvjetovanih događaja)

INTERVAL (cijeli broj između 1 i 65 635. Vraća ili postavlja broj milisekundi koji treba proći između dva dijela vremenski uvjetovanog potprograma. Svaki poziv predstavlja događaj Timer)

DOGAĐAJ TIMER događa se kad istekne navedeni broj milisekundi. 100 milisekundi = 1 sekunda

Zadatak 15.

Treba napisati koliko je sati (vrijeme) i da se vide sekunde koje prolaze i koji je datum. Na formu stavi dva objekta tipa Label i imenuj ih sa LabelVrijeme i LabelDatum. Postavi i jedan Timer imena Sat. Objektu sat postavi svojstvo interval na 100.

Sintaksa:

```
Private Sub Sat_Timer()  
    LabelVrijeme=Time  
    LabelDatum=Date  
End Sub
```

MSGBOX

Dijaloški okvir pomoću kojeg upozoravamo korisnika na neke promjene u programu ili pogrešno unesene podatke. Poruku pišemo pod dvostrukim navodnicima, stavljamo zarez pa pišemo ili izaberemo iz padajućeg izbornika *vbOKOnly* – postavlja gumb OK. Opet zarez i u navodnike stavljamo ime tj. Naslov poruke. Vidi sliku.

Sintaksa:

```
MsgBox "Nešto ste zabrljali!", vbOKOnly, "Opomena"
```



Možemo ovaj prozor staviti da se javlja kad nam Visual Basic javlja pogrešku. Prije pisanja MsgBox-a treba napisati `Exit Sub`

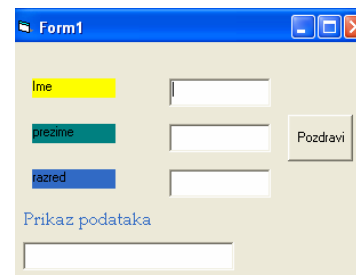
Sintaksa:

```
On Error GoTo pogreska *To pišemo na početku programskog koda  
pogreska: *Nakon Exit Sub, na samom kraju koda  
MsgBox "Nešto ste zabrljali!", vbOKOnly, "Opomena"
```

IF – THEN – ELSE

Uvjetno grananje, ovisno izvođenje grupe naredbi ovisno o vrijednosti uvjeta. Uvjet je obavezan. Može imati vrijednost True ili False.

```
Sintaksa:      If      UVJET      then  NIZNAREDBI
                ElseIf   NIZNAREDBI-n
                Else     NIZNAREDBI
                End If
```



Zadatak 16.

Napravi program koji će vas, nakon unosa vašeg imena pozdraviti. Forma treba izgledati kao na slici. Ako niste unijeli podatke treba vam se pojaviti prozor koji vas sa porukom upozorava. Poruka je: *Morate upisati ime.*

Koga pozdravljam? Naziv prozora je *Poruka*. (MsgBox).

Sintaksa:

```
If Text1.Text = "" Then      *Ako nije uneseno ništa onda
MsgBox " Morate upisati ime, Koga Pozdravljam ???" *Prozor
Else                          *Inače
Text4.Text = " Dobar dan " & Text1      *U objektu piše Dobar dan
End If
End Sub
```

Zadatak 17.

Nakon unosa broja program ga testira da li je pozitivan ili negativan te rezultat ispisuje. Ako je unesena 0 piše *Broj nije ni pozitivan ni negativan, unijeli ste nulu.*

```
Sintaksa      Private Sub Text1_Change()
                If Val(Text1) > 0 Then
                    Label1.Caption = "Broj je pozitivan!"
                Else
                    Label1.Caption = "Broj je negativan!"
                End If
                If Val(Text1) = 0 Then
                    Label1.Caption = "Broj nije ni pozitivan ni
                    negativan! Unijeli ste 0!"
                End If
                End Sub
```

Zadatak 18.

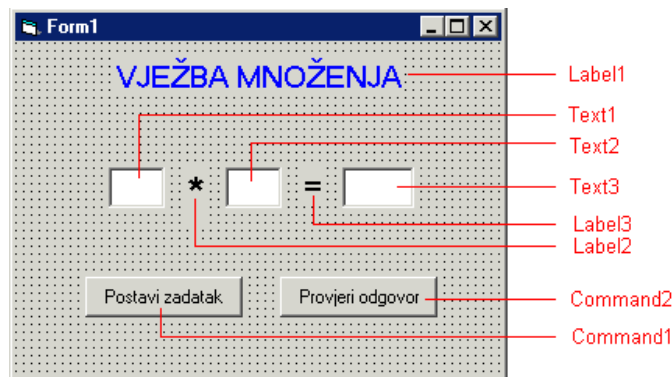
Treba napraviti program koji će kad korisnik unese broj reći da li je jednoznamenkast, dvoznamenkasti ili troznamenkasti. Naglasiti korisniku da unese broj do 1000. Ako korisnik unese veći broj neka se pojavi prozor (MsgBox) naziva *Napomena* koji će ga upozoriti da broj treba biti do 1000 i da ponovo unese broj.

Zadatak 19.

Napiši program VJEŽBA MNOŽENJA koji će na formi imati ove objekte:

- Label1 (svojstvo **Caption** = "VJEŽBA MNOŽENJA")
- Label2 (svojstvo **Caption** = "*" - znak množenja!)
- Label3 (svojstvo **Caption** = "=" - znak jednakosti!)

- **Text1** (za upis prvog generiranog broja)
- **Text2** (za upis drugog generiranog broja)
- **Text3** (za prihvatanje odgovora korisnika)
- **Command1** (svojstvo **Caption** = "Postavi zadatak")
- **Command2** (svojstvo **Caption** = "Provjeri odgovor")



Kada korisnik klikne na dugme **Command1** program treba generirati dva slučajna broja između 0 i 10 i upisati ih u **Text1** i **Text2**. Nakon što korisnik upiše svoj odgovor u **Text3** i klikne na dugme **Command2**, program provjerava da li je odgovor ispravan i o tome daje obavijest u **MsgBoxu**.

FUNKCIJA RND[Broj]

Vraća slučajno odabrani broj tipa **Single** iz raspona [0,1). Ako je broj manji od nule, vraća se stalno isti slučajni broj, ako je veći od nule vraća se sljedeći slučajni broj iz slijeda. Ako je broj jednak nuli vraća se zadnji proizvedeni slučajni broj. Prije pozivanja funkcije **RND** treba pozvati funkciju **Randomize** kako bi se inicijalizirao generator slučajnih brojeva sa satom sustava.

NAPOMENA: Pošto je funkcija **RND** iz raspona [0,1) treba množiti sa 10 ako želimo jednoznamenaste brojeve, sa 100 ako želimo i dvoznamenkaste brojeve itd.

PROGRAMSKI KOD	OBJAŠNJENJE PROGRAMA
<pre>Private Sub Command1_Click() Randomize Timer a = Int(Rnd * 10) b = Int(Rnd * 10) Text1.Text = a Text2.Text = b Text3.Text = "" End Sub</pre>	<pre>'uključenje generatora slučajnih brojeva 'generiranje prvog slučajnog broja (između 1 i 10) 'generiranje drugog slučajnog broja (između 1 i 10) 'upis prvog slučajnog broja u Text1 'upis drugog slučajnog broja u Text2 'brisanje starog odgovora u Text3</pre>
<pre>Private Sub Command2_Click() a = Val(Text1.Text) b = Val(Text2.Text) odgovor = Val(Text3.Text) c = a * b If odgovor = c Then MsgBox "Odgovor je točan!" Else MsgBox "Odgovor NIJE točan!" End If End Sub</pre>	<pre>'prihvatanje prvog zadanog broja iz Text1 'prihvatanje drugog zadanog broja iz Text2 'prihvatanje odgovora korisnika u varijablu odgovor 'izračun točnog odgovora za daljnju provjeru: 'ako je odgovor = c (korisnik točno izračunao) ' ispiši poruku "Odgovor je točan!" 'ako je odgovor <> c (korisnik netočno izračunao) ' ispiši poruku "Odgovor NIJE točan!" 'kraj logičke provjere</pre>