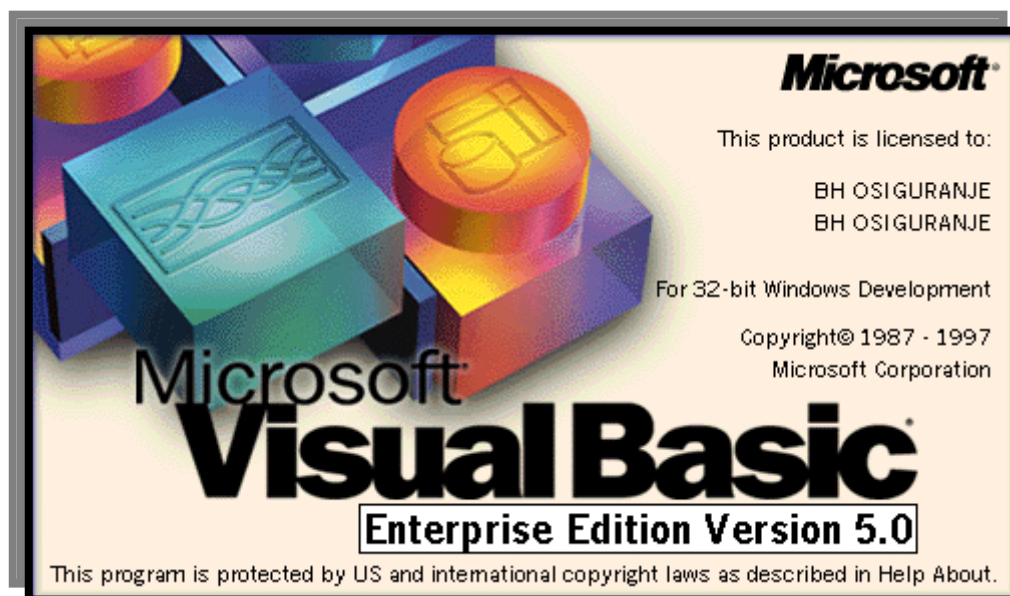


**Univerzitet «Džemal Bijedić» Mostar**  
Studij informatike

Skripta za predmet : **Uvod u programiranje**  
Programski jezik : **Visual Basic 5.0**  
Autor : demonstrator vježbi - **Gondžo Haris**

---

# Visual Basic 5.0



## Programski jezici

Programski jezik je sredstvo pomoću kojeg čovjek(programer) komunicira s računarom i kreira software. To je vještačka tvorevina sastavljena od simbola grupisanih u riječi. Svaki programski jezik mora zadovoljiti dva osnovna zahtjeva:

1. Da je razumljiv za čovjeka
2. Mogućnost prevođenja u oblik razumljiv računaru
- 3.

### Mašinski jezik

Mašinski jezik je niz jedinica i nula određene i konačne dužine. U prvoj fazi razvoja računara programiranje se vršilo isključivo mašinskim jezikom, što je bilo izuzetno teško, dugotrajno i mukotržno. Program napisan u mašinskom jeziku za računar je potpuno prihvatljiv, ali za čovjeka je totalno neprihvatljiv, jer se iz njega ne može direktno uočiti koji se problem riješava i po kom se algoritmu radi. A da bi se bolje razumio program napisan u mašinskom jeziku prevodi se iz binarnog u simbolički oblik, i to pomoću programa koji je namjenjen za to – **disasemblera**. Ali i poslije toga ostaje dosta komplikovan. Kada govorimo o programiranju u mašinskom jeziku potrebno je reći da je to dio prošlosti.

### Simbolički mašinski jezik

Simbolički mašinski jezik predstavlja veliki korak naprijed u odnosu na mašinski jezik. Osnovna karakteristika Simboličkog mašinskog jezika je uvođenje simboličkih naziva za svaki kod operacije. Tako program postaje pregledniji a samim tim lakše je uočiti greške. Prevođenje programa na mašinski jezik vrši – **assembler**, Program namjenjen za to.




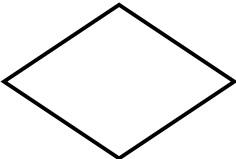
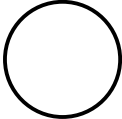
### Viši programski jezici

Viši programski jezici imaju sljedeće prednosti: olakšano programiranje, skraćeno vrijeme obuke u programiranju, nije nužno detaljno poznavanje hardwera računara, prenosivost programa(program se može premjestiti na drugi računar i izvršiti bez ikakvih izmjena. U grupu viših programskih jezika spadaju npr: Basic, Fortran, C jezik, Pascal ...

Program napisan u nekom od viših programskih jezika ne može se izvršiti direktno na računaru sve dok se izvorni program ne prevede na mašinski jezik. Proces prevođenja vrši poseban program(prevodilac) koji se naziva **kompajler**. Konačan rezultat kompajlera je izvršni program.

## Pojam algoritma

Niz koraka koji jasno i određeno vode ka rješenju nekog problema, odnosno daju odgovor da rješenja nema naziva se **ALGORITAM**. Za rješavanje problema na računaru neophodno je razviti odgovarajući algoritam. Da bi prikazali algoritam na papiru koristi ćemo za to predviđene simbole.

Simboli		Naziv simbola
	-	Početak/kraj
	-	Ulaz/Izlaz
	-	Obrada
	-	Ispitivanje uslova
	-	Vezni simbol

Čitav postupak rješavanja problema odvija se polazeći od korisničkih zahtjeva, preko definisanog i shvaćenog problema i globalnog algoritma toka. Jedan od standardnih načina zapisivanja algoritma je **diagram toka**. Dijagram toka sastoji se od niza simbola kojima se označavaju pojedine operacije algoritma. Pomoću navedenih simbola može se difinisati i najsloženiji algoritam. Na ovaj način imati ćemo pregledno razrađen cio postupak i tok njegovog izvođenja. Elementi dijagrama toka povezuju se usmjerenim linijama.

## Uvod

Kada govorimo o programskim jezicima, i to o danas aktuelnim, neizbježno je spomenuti i Visual Basic. Basic je programski alat koji je začet još davne 1963.god. na Darmut koleđu, i bio je prvi jezik koji je omogućavao početnicima rješavanje problema na kompjuterima. Skraćenica **B** (Beginners) **A** (All puppose) **S** (Simbolic) **I** (Instructions) **C** (code) u prevodu znači «Programski jezik opće namjene za početnike». Mada se dosta sa razvojem informatike mijenjao i usavršavao ipak je zadržao svoje osnovne karakteristike. Basic je prošao sve faze razvoja koje je morala proći, sa razvojem informatike i svaka druga aplikacija. Od GWBasica, QBasica, Visual Basica 3.0, Visual Basic-a 4.0 itd., pa do sada aktuelne verzije Visual Basic 6.0 (mada će vjerovatno uskoro biti predstavljena i verzija 7.0. Svaka nova verzija je donosila nova poboljšanja. Recimo da je GWBasic (radi pod DOS-om) i danas u planu i programu Osnovnih škola u BiH. Nakon svega programiranje u VB-u je lakše i brže se uči nego recimo u C++ i zato je ovaj programski alat sve popularniji.

## Visual Basic 5.0

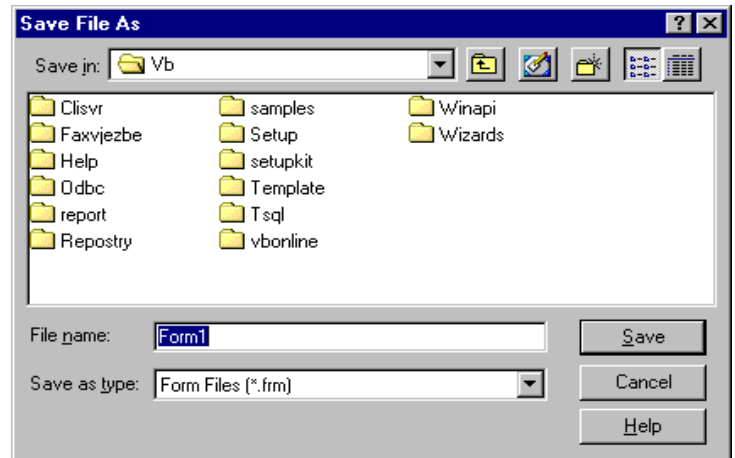
Visual Basic 5.0 je napredni programski alat kojim možete kreirati 32-bitne aplikacije, sa vrlo malo truda. Visual Basic spada u vizuelne programske alate. Njegovo okruženje je gotovo identično kao i kod svake window orijentisane aplikacije, sadrži Menije, Toolbarove, Wizarde. Pri pokretanju aplikacije pojavljuje nam se prozor koji se zove «New Project». Koji ujedno i novina u odnosu na predhodnu verziju. Ovaj prozor nam nudi da izaberemo šta to želimo raditi 7u Visual Basic-u, kreirati novi projekt, otvoriti postojeći ili koristiti predhodno rađeni. Vizualni izgled vaše aplikacije će zavisiiti isključivo od vas, VB nudi neograničenu mogućnost estetskog doradivanja. Objekte, kontrole ćete praktično povlačiti sa vašeg toolbar-a na formu i pomjerati ih na formi gdje želite. Predpostavimo da sa uvodnog prozora izaberemo opciju «Standard EXE» i dobiti ćemo kreiran «project1» u koji nam Visual Basic automatski uključuje Formu na kojoj ćemo smiještati kontrole. Par riječi o formi.



Forma je objekat na koji ćete smiještati vaše kontrole , tj. one kontrole (komandne dugmiće, menije, naslove, textbox-ove ... ) koje će vam biti potrebne za rješavanje zadanog problema. Prva forma imati će ime «Form1» što znači da će sve sljedeće forme imati oznake poput «Form2», «Form3» itd. U jedan projekt možemo uključiti neograničen broj formi.

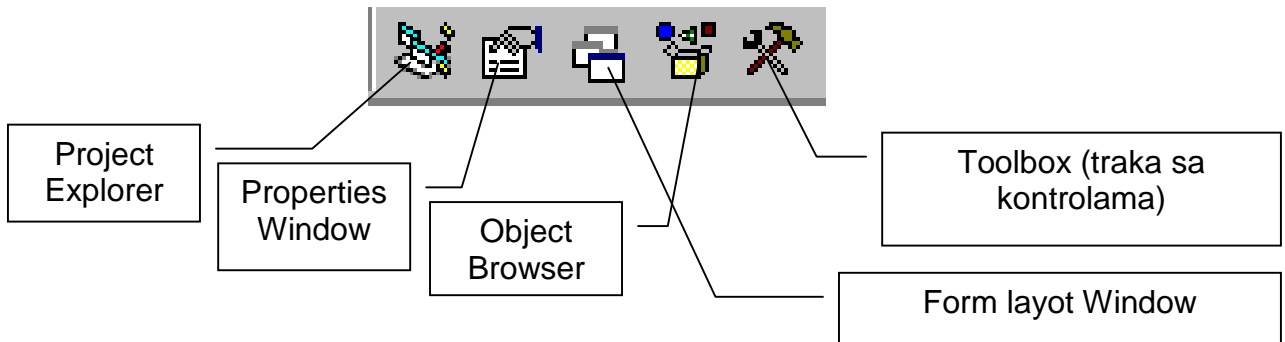
## Snimanje projekta

Snimiti svoj rad je osnovne operacija u svakom programu pa tako i u Visual Basic-u. Iz menija «File» izaberimo opciju «Save» i dodjelimo ime našem prvom projektu. Visual Basic će u zadani folder snimiti posebno projekt i posebno formu. U padajućem meniju «Save in» određujemo gdje će projekt biti snimljen, a u polju «File name» kako će se zvati projekt ili forma zavisno šta snimamo. Ostaje nam još samo klik na komandni dugmić «Save» i naš rad je snimljen.



## Toolbar-ovi

Trake sa alatima su standardni dijelovi aplikacija. Tako i Visual Basic, posjeduje trake sa alatima i preko menija «View» i opcije «Toolbars» uključujemo i isključujemo određene trake iz našeg projekta. «Standard» traka će vam biti neophodna za rad.



Toolbox je traka koja sadrži kontrole koje ćete uključivati u vaš projekt i to je traka koja će vam najvjerojatnije trebati najviše.

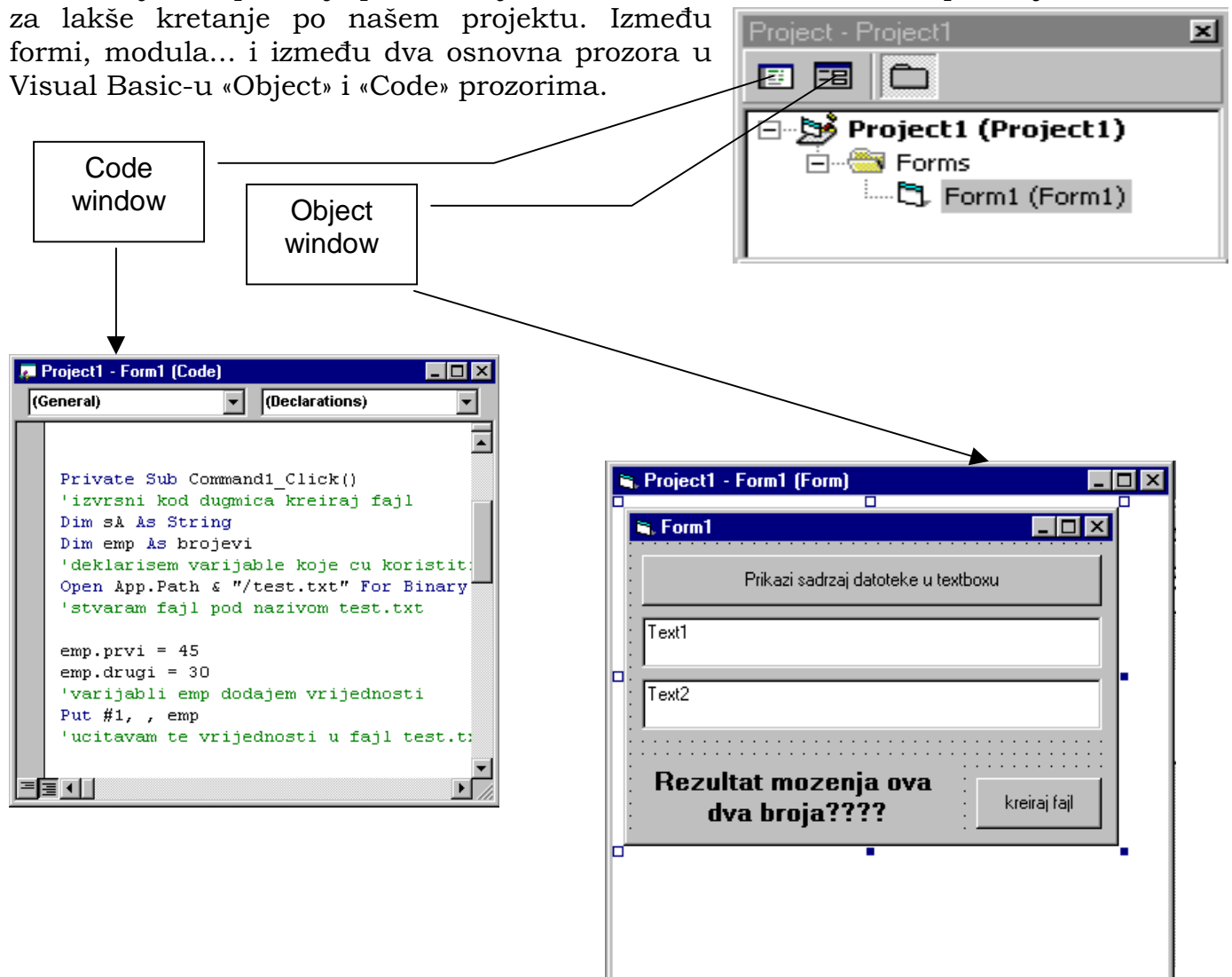


Pointer, PictureBox, Label, TextBox, CommandButton, CheckBox, OptionButton, CommoBox, ListBox, VerticalScrollBar, HorizontalScrollBar, Timer, DriveListBox, DirListBox, FileListBox, Shape, Line, Image, Data, Ole, su kontrone koje su uključene u standardni exe projekat, ali traka nije ograničena samo na ove kontrole. Smiještanje

kontrola na objekat je objašnjeno u poglavlju «Uključivanje novih kontrola u vaš projekat».

## Projekt explorer

Projekt explorer je prozor smješten obično sa desne strane aplikacije i služi nam za lakše kretanje po našem projektu. Između formi, modula... i između dva osnovna prozora u Visual Basic-u «Object» i «Code» prozorima.



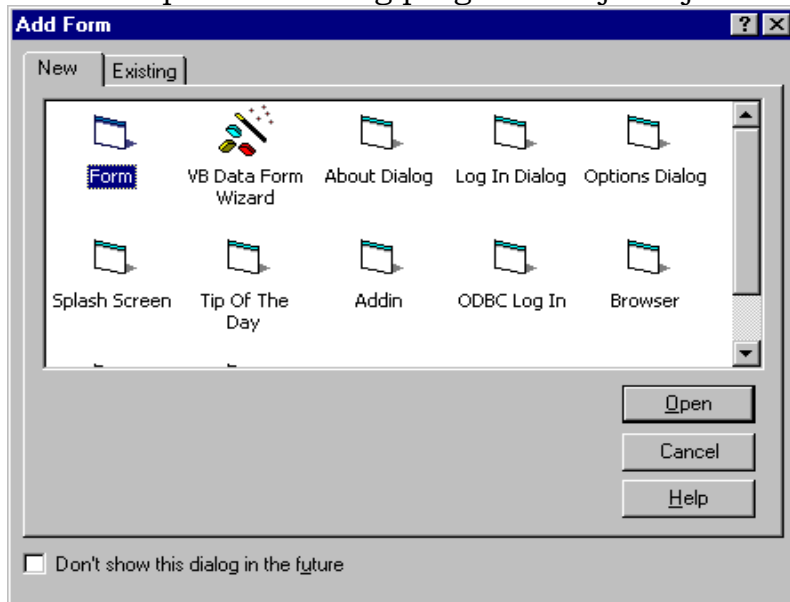
na slici imamo prikazana i «code window» koji će nam trebati za pisanje koda tj. određivanje ponašanja kontrola i objekata, i «objekt window» prozor koji nem je potreban da vizuelno uređujemo aplikaciju.



formu. Koraci poput, programiranje te kontrole ne spadaju o ovo poglavlje a i razlikuju se od kontrole do kontrole. Ali većina kontrola kada ih selektujete u prozoru «properties» ima opciju «custom» pod kojim ćete moći podesiti većinu parametara kontrole bez koda.

### Insertovanje nove forme u vaš projekt

Aplikacije rađene u Visual Basic-u mogu imati neograničen broj formi, zavisno od vrste i potreba samog programa kojim riješavamo određen problem.



odaberete koji formu želite.

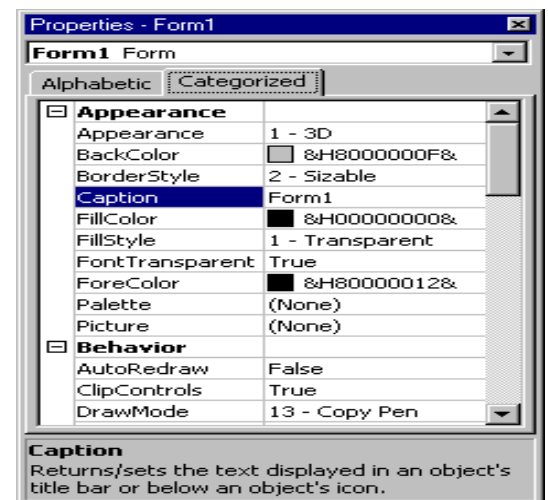
Visual Basic vam nudi već gotove predloške formi koje su urađene po ugledu na klasične windows prozore. Npr. «About Dialog», «Log In Dialog», «Options Dialog!», «Splash Screen» itd. Između ovih predložaka postoji i «VB Data Form Wizard». To je čarobnjak koji vas kroz nekoliko koraka vodi ka automatskom povezivanje vaše forme sa bazom podataka. Sljedeći postupak vam pokazuje kako se insertuje nova forma u vaš projekt. Pod menijem «project» kliknite na opciju «Add Form». I dobićete prozor ka o na slici. Vaše je još samo da

### Svojstva objekata – properties window

Visual Basic vam omogućava da bez pisanja suvišnih redova koda odredite svojstva objekata. Za primjer uzeti ćemo običnu formu. Pozor je podjeljen na dva dijela. Gornji dio gdje možete birati objekt za koji želite odrediti svojstva i donji dio gdje možete pregledati i editirati svojstva objekata. Navesti ćemo, kao primjer nekoliko svojstava koja je moguće podešavati u ovom prozoru.

- ⇒ *Caption* – natpis u naslovnoj traci forme
- ⇒ *BackColor* – boja forme
- ⇒ *BorderStyle* – stil granica forme
- ⇒ *Icon* – ikonica u naslovnoj traci prozora
- ⇒ *Font* – font forme
- ⇒ *Picture* – slika koja će biti učitana na formu

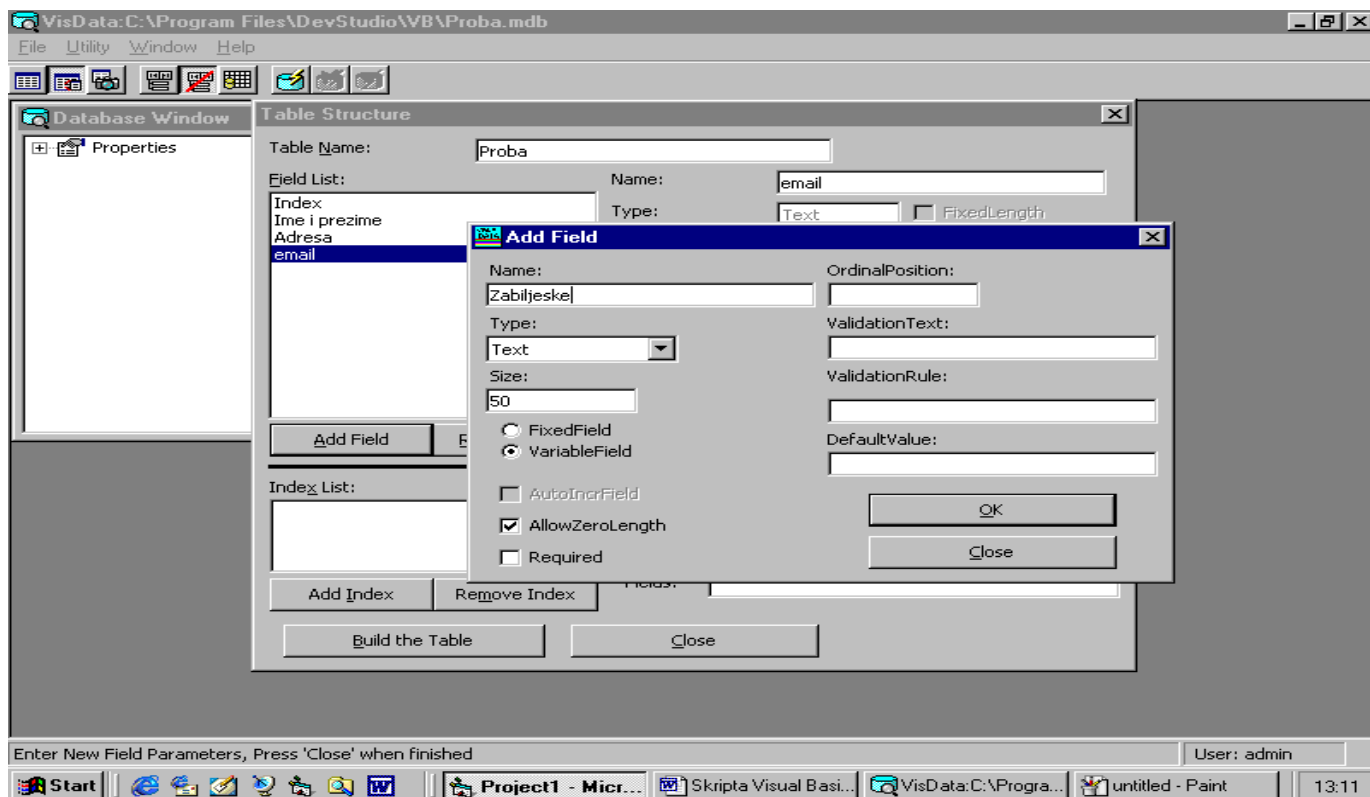
*Properties window* se uključuje u vaš projekat preko menija «view» ili prečicom «F4».



## Visual Data menager

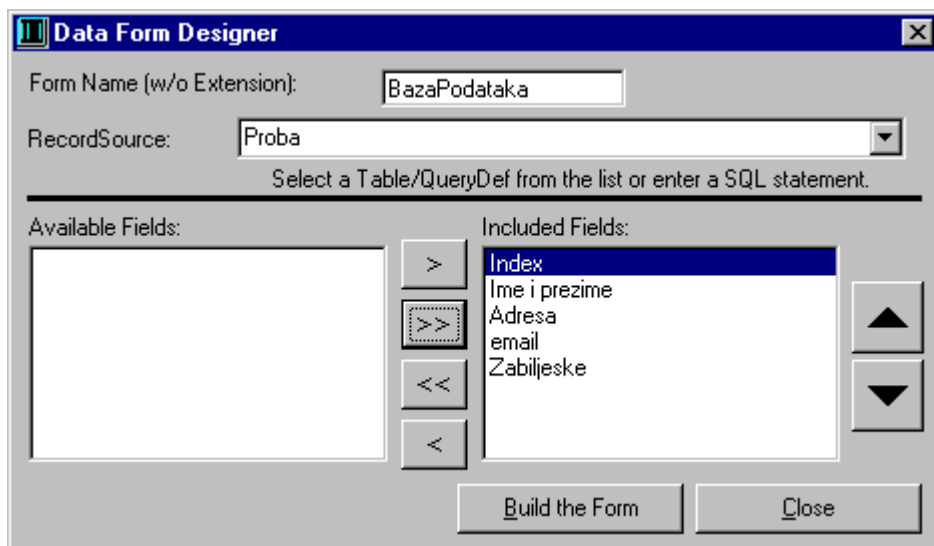
Rad sa bazama podataka je dio poboljšanja ove verzije Visual Basic-a. Visual Basic je sada u mogućnosti da pomoću svog podprograma Visual Data Menager-a uređuje i kreira baze podataka različitih formata. Visual Data Menager se pkrije pod menijem «Add-ins». Ovaj program je pomoćni program Visual Basic-a i direktno je povezan sa Visual Basic-om. Opsati ćemo postupak kreiranja nove baze podataka u ovom programu i automatsko kreiranje forme za upravljanje tom bazom podataka.

U meniju «File» odaberite opciju «New»a zatim «Microsoft Access» i verziju 7.0. Data Menager će vas pitati gdje i pod kojim imenom da snimi vašu bazu. Kada ste odredili ime baze, u našem slučaju «Proba» i mjesto gdje će se pohraniti, u «database window» kliknite desnim klikom na natpis «properties» i odaberite opsiju «New Table». Dobiti ćete prozor gdje se od vas očekuje da kreirate tabelarnu strukturu vaše baze podataka. Kao što se na slici vidi naša pokazna baza podataka nosi ime «Proba» a sadrži polja kao što su «index, Adresa. itd. Kada upišete ime tabele klikom na dugmić «Add Field» dodajete polja toj tabeli. Kada ste dodali sva potrebna polja vašoj bazi podataka ostaje da



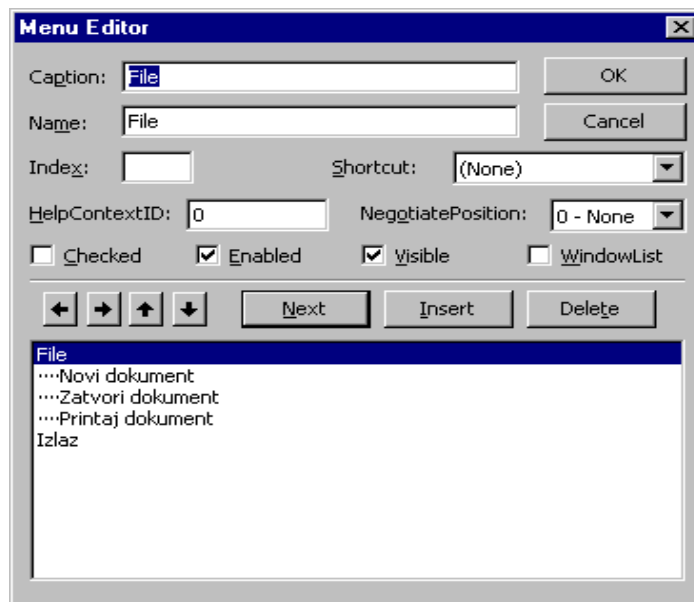
klikom na dugmić «Build the Table» izgradite tabelu. Sada u «Database window» imate i natpis «Proba» koji predstavlja vašu bazu podataka, sa svim poljima i njihovim karakteristikama. Sada nam ostaje poslednji korak koji će automatski sa već urađenim kodom sagraditi novu formu u vašem projektu. U meniju «utility» izaberite opciju «Data Form Disigner». dobiti ćete prozor kao na slici. U polje «Form Name» unesite ime forme,

U polju «recordSource» izaberite vašu bazu podataka. I iz dobijenih «Available Fields» tj. dostupnih polja prebacite strelicama u «included Fields» tj. uključena polja, polja koja želite da su na vašoj formi. Kada je i taj korak završen ostaje nam da kliknemo na komandni dugmić «Build the Form». Visual Data Menager će kreirati novu formu za upravljanje vašom bazom podataka bez potrebe da vi trošite vrijeme i pišete kod. Osim ovih postupaka koje smo naveli Visual Data Menager nam nudi i druge operacije nad bazama podataka.



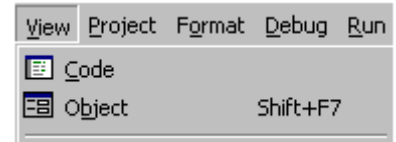
## Menu Editor

Dodavanje standardnih windows menija je riješeno u Visual Basic-u preko opcije «menu Editor» koja se nalazi u meniju «Tools». U ovom prozoru potrebno je da unesete naslov vašeg menija ( na slici «file»), naziv koji će Visual Basic koristiti kada budete pisali kod za njega. Položaj menija tj. da li će biti samostalan (na slici «Izlaz» i «File») ili će biti pod nekim drugim (na slici «Novi dokument» ili Zatvori dokument») koji su pod meniji menija «File». tu operaciju položaja novog menija ćete podesiti strelicama pored dugmića «Next». Možete podesiti još nekoliko svojstava menija ka što su prečice, vidljivost itd. Ako je to sve što ste htjeli kliknite na dugmić «OK» i imate menije integrisane u vašu formu. Ostaje vam da napišete kod na svaku opciju posebno. Prozor za pisanje koda dobijate kada kliknete jednom na opciju za koju želite pisati kod.



## Prozori za pisanje koda i vizuelno uređivanje

Za lagodniji način rada u Visual Basic-u potrebno je da se znate prebacivati između dva osnovna prozora. To su «Code Window» i «Object Window». «Code Window» je prozor u kojem udeđujemo i pišemo kod za objekte koji su na formi. Do njega dolazimo preko menija «View» i opcije «Code», ili jednostavnim dvoklikom na objekat za koji želimo pisati kod. Drugi slučaj je vjerovatno prihvatljiviji za početnike jer automatski ispiše recimo standardni dio koda koji u većini slučajeva moramo ispisati sami. «Object Window» je prozor za Vizuelno uređivanje naših aplikacija. Do ovog prozora dolazimo preko menija «View» i opcije «Object»

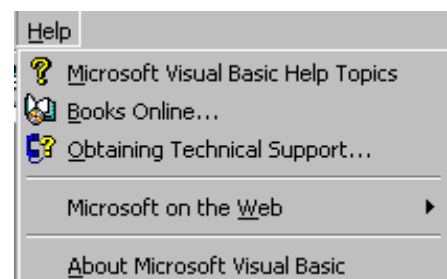
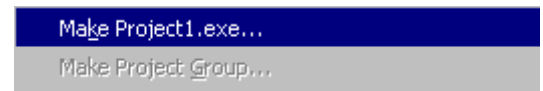


## Pokretanje programa i kompajliranje u exe datoteku

Da bi vidjeli kako naš program radi moramo ga pokrenuti i ispraviti eventualne greške. Pokretanje programa se vrši preko opcije «Start» koja se nalazi u meniju «Run» ili prečicom «F5». Prekidanje izvršenja programa vrši se klikom na opciju «End» u istom menju. Eventualne greške će se uočiti prilikom pokretanja programa. Visual Basic će prekinuti izvršenje programa i označiti dio koda koji sadrži grešku. Iako smo vidjeli naš program na djelu, još uvijek nemamo izvršni program tj. exe datoteku na disku. Opcija koja nam to omogućava nalazim se u meniju «File» i naziva se « Make ..... . exe». Ostaje vam da kažete gdje želite da snimate exe datoteku i pod kojim imenom. Exe datoteka se može prenositi sa na druge računare ali na kojim ima instaliran VB 5.0 ili ako ponesete nekoliko sistemskih datoteka bez kojih vaš program neće raditi na drugim računarima.

## Help

Pomoć koju nudi Visual Basic je vrlo iscrpna i protkana primjerima ali iskoristiva naravno samo za one koji nabadaju engleski. U 6.0 verziji programeri iz Microsofta su za taj segment ovog programskog jezika rezervisali čak dva CD-a, koji sadržavaju mnoštvo primjera i objašnjenja. Osim helpa koji dolazi uz programski paket Microsoft nudi i podršku preko interneta, i Microsoft Books Online. Ova posljednja pomoć je prava mala enciklopedija VB-a. U webolikoj aplikaciji imate potpunu knjigu koja vas vodi kroz Visual Basic 5.0, sa laganim primjerima i dobro riješenom potragom za onim što vam treba.



**Osnovni tipovi podataka:**

BOOLEAN - logički tip podataka, može imati vrijednosti TRUE ili FALSE.

BYTE - numericki cjelobrojni tip podataka, cije se vrijednosti mogu kretati između 0 i 255

CURRENCY - numericki decimalni, vrlo precizni tip podataka, koji smješta 15 cifara lijevo od decimalne tačke i 4 cifre desno od decimalne tačke. Raspon vrijednosti je od -922,337,203,685,477.5808 do 922,337,203,685,477.5807.

DATE - tip podataka koji reprezentuje datume i vremena. Datumi se mogu kretati u rasponu 01.01.100 do 21.12.9999, a vrijeme od 00:00:00 to 23:59:59.

DOUBLE - ovaj tip predstavlja numeričke decimalne podake dvostruke preciznosti. Podaci se mogu kretati u rasponu od -1.79769313486232E308 do -4.94065645841247E-324 za negativne vrijednosti i 4.94065645841247E-324 do 1.79769313486232E308 za pozitivne vrijednosti.

INTEGER - tip koji predstavlja cjelobrojne numericke podatke. Podaci se mogu kretati u rasponu -32,768 do 32,767.

LONG - tip koji predstavlja cjelobrojne numericke podatke dvostruke duzine. Moze se kretati u rasponu -2,147,483,648 do 2,147,483,647.

OBJECT - tip podakak koji u sebi moze smještati objekete.

SINGLE - ovaj tip predstavlja numeričke decimalne podake dvostruke preciznosti. Podaci se mogu kretati u rasponu od -3.402823E38 do -1.401298E-45 z anegativne vrijednosti i od 1.401298E-45 do 3.402823E38 za pozitivne vrijednosti.

STRING - Tip koji u sebi smješta karakterne podatke, nizove karaktera. Max. duzina stringa moze biti oko 2 milijarde karaktera.

VARIANT - Tip koji u sebe može smjestiti **sve ostale tipova podataka**, uključujući i vrijednosti predstavljene ključnim riječima **Empty**, **Error**, **Nothing**, i **Null**.

**OPERATORI U VB5**

Redosljed prioriteta:

<u>Aritmetički</u>	<u>Poređenje</u>	<u>Logički</u>
Stepenovanje(^)	Jednako (=)	<b>Not</b>
Negacija (-)	Razlicito (<>)	<b>And</b>
Množenje, dijeljenje (*, /)	Manje od (<)	<b>Or</b>
Cijelobrojno dijeljenje (\)	Veće od (>)	<b>Xor</b>
Modus ( <b>Mod</b> )	Manje ili jednako (<=)	<b>Eqv</b>
Sabiranje, oduzimanje (+, -)	Veće ili jednako (>=)	<b>Imp</b>
Spajanje stringova (&)		<b>Like, Is</b>

**Imenovanje kontrola:**

<u>Tip kontrole</u>	<u>Prefix</u>	<u>Primjer</u>
3D Panel	pnl	pnlGroup
Animated button	ani	aniMailBox
Check box	chk	chkReadOnly
Combo box, drop-down list box	cbo	cboEnglish
Command button	cmd	cmdExit
Common dialog	dlg	dlgFileOpen
Communications	com	comFax
Data control	dat	datBiblio
Data-bound combo box	dbcbo	dbcboLanguage
Data-bound grid	dbgrd	dbgrdQueryResult
Data-bound list box	dblst	dblstJobType
Directory list box	dir	dirSource
Drive list box	drv	drvTarget
File list box	fil	filSource
Form	frm	frmEntry
Frame	fra	fraLanguage
Gauge	gau	gauStatus
Graph	gra	graRevenue
Grid	grd	grdPrices
Horizontal scroll bar	hsb	hsbVolume

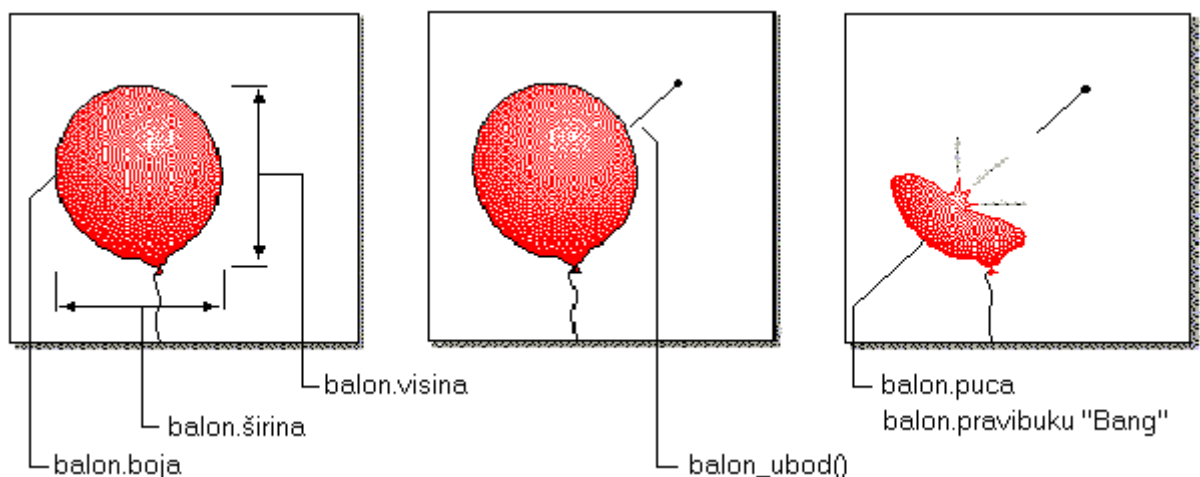
Image	img	imgIcon
Key status	key	keyCaps
Label	lbl	lblHelpMessage
Line	lin	linVertical
List box	lst	lstPolicyCodes
MAPI message	mpm	mpmSentMessage
MAPI session	mps	mpsSession
MCI	mci	mciVideo
MDI child form	mdi	mdiNote
Menu	mnu	mnuFileOpen
MS Flex grid	msg	msgClients
MS Tab	mst	mstFirst
OLE	ole	oleWorksheet
Outline	out	outOrgChart
Pen BEdit	bed	bedFirstName
Pen HEdit	hed	hedSignature
Pen ink	ink	inkMap
Picture	pic	picVGA
Picture clip	clp	clpToolbar
Report	rpt	rptQtr1Earnings
Shape	shp	shpCircle
Spin	spn	spnPages
Text box	txt	txtLastName
Timer	tmr	tmrAlarm
UpDown	upd	updDirection
Vertical scroll bar	vsb	vsbRate
Slider	sld	sldScale
ImageList	ils	ilsAllIcons
TreeView	tre	treOrganization
Toolbar	tlb	tlbActions
TabStrip	tab	tabOptions
StatusBar	sta	staDateTime
List View	lvw	lvwHeadings
ProgressBar	prg	prgLoadFile
RichTextBox	rtf	rtfReport

### Imenovanje varijabli

Tip podataka	Prefix	Primjer
Boolean	bln	blnFound
Byte	byt	bytRasterData
Collection object	col	colWidgets
Currency	cur	curRevenue
Date (Time)	dtm	dtmStart

Double	dbl	dblTolerance
Error	err	errOrderNum
Integer	int	intQuantity
Long	lng	lngDistance
Object	obj	objCurrent
Single	sng	sngAverage
String	str	strFName
Variant	vnt	vntChecksum

## **OBJEKTI: SVOJSTVA, DOGAĐAJI I METODI**



Sliku ste već vjerovatno vidjeli ali podsjetimo se malo šta se to ovom slikom hoće reći. Prvo da objasnimo šta je to objekat! Objekat može biti komandni dugmić, label, textbox, ili bilo koja druga kontrola, sa ovog stanovišta objekat je takođe i forma. Svaki objekat ima svoja svojstva.

Svojstvo objekta opisuje objekat na neki način. (Boja, font, širina, visina...) Označava se kao **objekat.svojstvo**. Ovaj način označavanja je standardan u Visual Basic-u. A da bi ti objekti bili od neke koristi tj. riješavali zadane probleme, moraju postojati događaji.

Događaj je ono što se intervencijom korisnika ili programa dogodi objektu. (Upis teksta, klik mišem, pomjeranje...). Događaj se opisuje procedurom, i označava kao: **objekat\_događaj()**.

Metod je ona akcija koju na zahtjev programa izvršava objekat (najšesće izvršavanje tadataka koje objekat ima - ako je u pitanju Multimedija objekat, to je sviranje nekog zvuka ili prikazivanje videa). Označava se sa **objekat.metod**.

## FUNKCIJE

Funkcije su posebno izdvojen skup linija koda, koje će se izvršavati više od jednog puta, sa ciljem nekih proračuna i vraćanja rezultata tih proračuna.

Sintaksa deklaracije funkcija:

```
[Public | Private | Friend] [Static] Function_name [(arglist)] [As type]  
    [statements]  
    [name = expression]  
[Exit Function]  
    [statements]  
    [name = expression]  
End Function
```

Primjer funkcije:

### **Private Function Linearna (X As Double) As Double**

```
DIM a As Double  
DIM b As Double
```

```
a = 2  
b = 3
```

```
Linearna = a*X + b
```

### **End Function**

Poziv ove funkcije izgleda ovako:

```
txtIspis.txt = Linearna(5)
```

Rezultat u tekst boxu "txtIspis" bio bi 13.

## PROCEDURE

Procedura je namjenski izdvojen dio programa koji će se izvršavati više puta tokom programa i izvoditi određene programske zadatke. Za razliku od funkcija, procedure nemaju povranu vrijedost.

Deklaracija procedure:

```
[Private | Public | Friend] [Static] Sub Ime_Procedure [(arglist)]
```

```
    [statements]
    [Exit Sub]
    [statements]
End Sub
```

Primjer procedure:

**Private Static Sub** IspisiKvadrat (KojiBox as Integer, Broj as Double)

```
DIM X As Integer
```

```
X = X + Broj*Broj
```

```
    IF KojiBox = 1 THEN
        txtPolje1.Text = X
    ELSE
        txtPolje2.Text = X
    END IF
```

**End Sub**

Poziv ove procedure bio bi:

**IspisiKvadrat (1, 5)**

A rezultat bi bio ispisivanje broja 25 u txtPolje1.

Ponovni poziv procedure sa parametrima:

**IspisiKvadrat (2, 2)**

ispisao bi broj 29 u txtPolje2.

### **Vidljivost varijabli**

PRIVATE - Ukoliko varijablu, deklariramo kao PRIVATE (podrazumijevano, sa DIM) unutar neke funkcije ili procedure, ona će biti vidljiva i prepoznatljiva samo unutar te procedure.

Ukoliko varijablu deklariramo kao PRIVATE na početku modula, u sekciji "General Declarations", ona će biti vidljiva u svim procedurama i funkcijama koje pripadaju tom modulu.

PUBLIC - Ukoliko varijablu deklariramo kao PUBLIC na početku modula, u sekciji "General Declarations", ona će biti vidljiva u svim procedurama i funkcijama u programu.

STATIC - Ukoliko varijablu deklariramo kao STATIC u okvir neke procedure ili funkcije, njena vrijednost će biti zadržana za ponovnu upotrebu prilikom ponovnog poziva te funkcije ili procedure.

### **Vidljivost i dužina vijeka procedura i funkcija**

PRIVATE - Ukoliko funkciju ili proceduru deklariramo kao PRIVATE, ona će moći biti vidljiva samo unutar modula u kojem se nalazi.

PUBLIC - Ukoliko funkciju ili proceduru deklariramo kao PUBLIC, ona će moći biti vidljiva unutar cijelog programa.

STATIC - ukoliko funkciju ili proceduru deklariramo kao STATIC, sve varijable koje su deklarirane unutar te procedure ili funkcije će zadržati svoju vrijednost i koristiti je prilikom slijedećeg poziva te procedure ili funkcije.

### **Primjeri**

Svaki programski jezik je najbolje učiti kroz konkretne primjere pa je to slučaj i sa Visual Basic-om. Kroz nekoliko primjera, koji su rađeni na vježbama naučiti ćete osnovna pravila programiranja u Visual Basic-u. Prije konkretnih primjera, par stranica neophodnog gradiva. Tipovi podataka su osnova za uspješno programiranje.

### Primjer br. 0 – (Pozdravna poruka)

O našem prvom primjeru tj. programu koji će nam pojasniti osnovne funkcije Objekata koje budemo koristili na kako smo već rekli automatski formiranoj Form1. Koristićemo komandne dugmiće i Label.

Sa toolboxa (s lijeve strane aplikacije) dvostrukim klikom na CommandButton insertujete komandni dugmić na formu, zatim jednostavnim povlačenjem pomjerite dugmić gdje vam se sviđa. Kad ste pomjerali dugmić ponovite postupak još jednom tako da sad imamo dva dugmića na Form1. Zatim ponovite isti postupak ali sa "label" objektom, koji ćete takođe smjestiti gdje vam se sviđa. Sad imamo formu na kojoj imate dva dugmića i label.

**Dvostrukim klikom na formu** dobiti ćemo "code window". Prozor u koji smještamo sljedeći kod i to za objekat na koji smo dvostruko kliknuli.

#### Private Sub Form\_Load()

```
Form1.Caption = "MOJ PRVI
PROGRAM"
'definisem naslov forme
'Caption je svojstvo objekta koje
označava njegov naslov
'u slučaju forme to će biti natpis na
naslovnoj traci prozora
Command1.Caption = "Klikni"
'definisem naslov dugmica
'u slučaju dugmića to će biti natpis na
dugmiću
Command2.Caption = "Ponisti"
'zadajem naslov drugom dugmicu
Label1.Caption = ""
'definisem naslov labela
'u slučaju labela caption će bitinatpis
unutar labela
End Sub
```

#### Dvostruki klik na Command1, i upišite sljedeći kod:

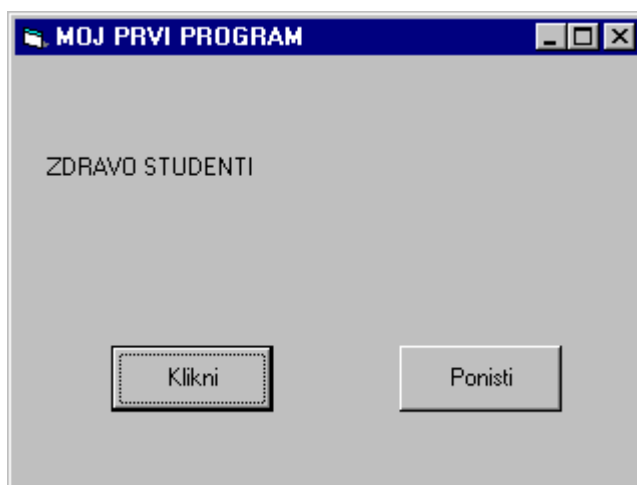
#### Private Sub Command1\_Click()

```
Label1.Caption = "ZDRAVO STUDENTI"
'popunjavam label1 sa prijateljskom
porukom ZDRAVO STUDENTI
End Sub
```

#### Dvostruki klik na Command1, i upišite sljedeći kod:

#### Private Sub Command2\_Click()

```
Label1.Caption = ""
'brisem naslov labela1
End Sub
```



**Primjer br 1.**

U sljedećem primjeru demonstrirati ćemo jednostavne računске operacije sa zadatim vrijednostima varijabli. Najprije kreirajmo novi "standard exe" projekat. Zatim na form1 insertujte šest komandnih dugmića, jedan label, i dvije line )linije kojima ćete odvojiti dijelove aplikacije.

Zatim projektu dodajte sljedeći kod :

'Moja druga aplikacija koja demonstrira izvodjenje osnovnih racunskih operacija  
'za dva vec zadana broja.

**Private Sub Command1\_Click()**

```
Dim a As String
Dim b As String
Dim c As Integer
'deklarisem varijable
a = 10
b = 2
'zadajem vrijednosti varijabli a i b
c = a * b
'izracunavam vrijednost varijable c
mnozenjem
Label1.Caption = c
'popunjavam label1 izracunatom
vrijecnoscu
Form1.Caption = "operacija mnozenja"
'dodajem naslov na formu
```

**End Sub****Private Sub Command2\_Click()**

```
Dim a As String
Dim b As String
Dim c As Integer
'deklarisem varijable
a = 10
b = 2
'zadajem vrijednosti varijabli a i b
c = a / b
'izracunavam vrijednost varijable c
dijeljenjem
Label1.Caption = c
'popunjavam label1 izracunatom
vrijecnoscu
Form1.Caption = "operacija dijeljenja"
'dodajem naslov na formu
```

**End Sub****Private Sub Command3\_Click()**

```
Dim a As String
Dim b As String
Dim c As Integer
'deklarisem varijable
a = 10
b = 2
'zadajem vrijednosti varijabli a i b
c = a - -b
'izracunavam vrijednost varijable c
sabiranjem
Label1.Caption = c
'popunjavam label1 izracunatom
vrijecnoscu
Form1.Caption = "operacija sabiranja"
'dodajem naslov na formu
```

**End Sub****Private Sub Command4\_Click()**

```
Dim a As String
Dim b As String
Dim c As Integer
'deklarisem varijable
a = 10
b = 2
'zadajem vrijednosti varijabli a i b
c = a - b
'izracunavam vrijednost varijable c
oduzimanjem
Label1.Caption = c
'popunjavam label1 izracunatom
vrijecnoscu
Form1.Caption = "operacija oduzimanja"
'dodajem naslov na formu
```

**End Sub**

**Private Sub Command5\_Click()**

Label1.Caption = ""

'ponistavam text ucitan u label1

Form1.Caption = "Ponisteno!!!"

'Dodajem naslov formi

**End Sub**

**Private Sub Command6\_Click()•End**

prekidam rad aplikacije

**End Sub**

**Private Sub Form\_Load()**

Form1.Caption = "MOJ DRUGI  
PROGRAM"

'dodajem naziv formi

Label1.Caption = "Rezultat?"

Command1.Caption = "Sabiranje"

Command2.Caption = "Oduzimanje"

Command3.Caption = "Mnozenje"

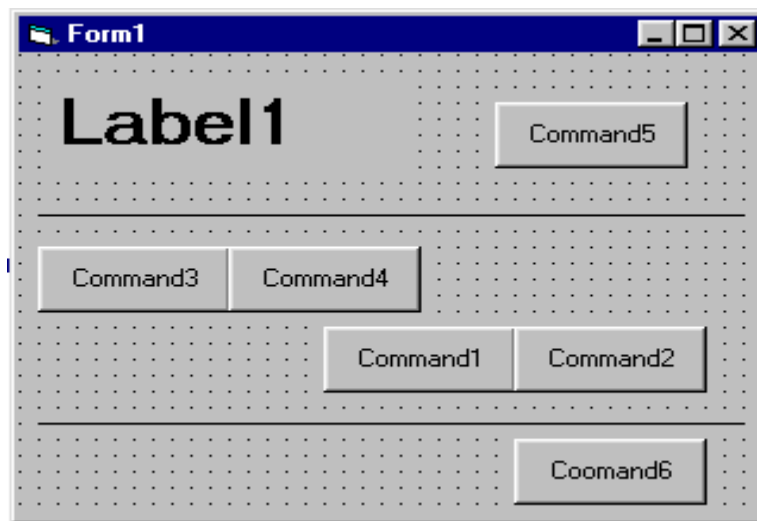
Command4.Caption = "Dijeljenje"

Command5.Caption = "Ponisti"

Command6.Caption = "Izlaz"

'Pridodajem nazive komandnim  
dgmicima

**End Sub**



**Primjer br 2.**

a)

Današnji primjer će se odnositi na svojstva objekata koje koristimo u Visual Basicu a konkretno se tiču vidljivosti i nevidljivosti nekog objekta u datom momentu izazvanom određenim događajem. To svojstva se označava sa riječju "**VISIBLE**" što izvorno engleski znači VIDLJIVOST. Pored ovog postavljenog svojstva pored određenog objekta potrebno je odrediti, razdvojeno znakom jednakosti tačnost ili netačnost ovog svojstva. Za tačnu tvrdnju koristimo "True" a za netčnu "False"

Imamo Formu1 i dva komandna dugmiæa i Label1 koji ima svojstvo "Caption" (naslov) "MALA PROBA", ako želimo da nam **Label1** nakon klika na Command1, nestane tj. da njegovo svojstvo vidljivosti bude netčno kucaćemo iza Command1 dugmića sljedeći kod:

**Private Sub Command1\_Click()**

Label1.Visible = False

**End Sub**

A ako želimo da nam se ponovo pojavi naš natpis kada kliknemo da dugmić Command2, iza Command2 unosimo sljedeći kod:

**Private Sub Command2\_Click()**

Label1.Visible = True

**End Sub**

Vježba: Probati unijeti kod koji će definisati sljedeće radnje i događaje.

1. Klik na Label1 uzrokuje nestanak Command2 dugmića.
2. Klik na Command1 uzrokuje nestanak i Label1 i Command2 dugmića.

b)

Drugi današnji primjer æe se odnositi na takođe svojstva objekata i kontrola u Visual Basicu a konkretno na takozvane MsgBox-ove tj message boxove, koji se koriste za komunikaciju sa korisnikom u obliku kratkih kratkih poruka.

Kreiraćemo Formu1 i na njoj komandni dugmić Command1. I sljedećim kodom izgraditi poruku korisniku da ga iskreno pozdravljamo.

**Private Sub Command1\_Click()**

MsgBox "Kako si mi?", vbOKOnly, "Proba"

**End Sub**

vježba: Probati kreirati kratke poruke koje ć govoriti korisniku sljedeće:

1. Upozoravati korisnika da nije dozvoljena operacija koju je izveo.
2. Obavještavati korisnika o autorskim pravima (Autor ovog programa je "Gondžo Haris".)

**Primjer br. 3**

'Aplikacija koja će nam demonstrirati petlje **if, then, else** kroz jednostavan test za aplikaciju ce nam trebati dvije forme i sljedeći objekti raspoređeni na formu po želji: 10 komandnih dugmića, 6 labela, 2 textboxa, na jednu formu i dva komandna dugmića i label na drugu formu. Novu formu ćete dodati preko menija "Project" i opcije "AddForm".

**Private Sub Command1\_Click()**

```
Label3.Caption = "Koliko je 5 + 5"
'Definisem pitanje kao naslov labela3
Command7.Enabled = True
'Cinim provjeru tacnosti prvog pitanja
dostupnom
Text1.Text = ""
'Postavljam svojstvo text objekta text1
na prazno
Text1.SetFocus
'Postavljam kursor u prostor za
kucanje odgovora
End Sub
```

**Private Sub Command10\_Click()**

```
If Text1.Text = "2" Then
'Ako je vijednost svojstva text objekta
text1 jednako 2 onda
Label5.Caption = "Tacno"
'Label5 tj njegovo svojstvo Caption ima
vrijednost Tacno
Text2.Text = Text2.Text + 1
'Brojim bodove u text2
Else
'u drugom slucaju
Label5.Caption = "Netacno"
'Label5 tj njegovo svojstvo Caption ima
vrijednost netacno
Text2.Text = Text2.Text - 1
'Oduzimam bod
End If
'Kraj ako
End Sub
```

**Private Sub Command11\_Click()**

```
If Text1.Text = "15" Then
Label5.Caption = "Tacno"
Text2.Text = Text2.Text + 1
Else
Label5.Caption = "Netacno"
```

```
Text2.Text = Text2.Text - 1
End If
If Text2.Text = "5" Then
'Ako je svojstvo text objekta text2
jednako broju 5 onda
Label5.Caption = " Svaka cast ti si
genije!!!"
'vrijednost svojstva caption labela5
jednaka recenici "svaka ti ...."
Else
'u drugom slucaju
End If
'Kraj ako
If Text2.Text = "4" Then
Label5.Caption = " Skoro savrseno!!!"
Else
End If
If Text2.Text = "3" Then
Label5.Caption = " Dodab ko dobar
dan!!!"
Else
End If
If Text2.Text = "2" Then
Label5.Caption = " Prolazna ocjena!!!"
Else
End If
If Text2.Text = "1" Then
Label5.Caption = " Hajde nek nisi u
minusu!!!"
Else
End If
If Text2.Text = "0" Then
Label5.Caption = " Nula ko da nisi ni
poceo!!!"
Else
End If
If Text2.Text = "-1" Then
Label5.Caption = " Dobro je nek nije -
5!!!"
Else
```

```

End If
If Text2.Text = "-2" Then
Label5.Caption = " E ovo je vec
kriticno!!!"
Else
End If
If Text2.Text = "-3" Then
Label5.Caption = " Katastrofa!!!"
Else
End If
If Text2.Text = "-4" Then
Label5.Caption = " Kataklihma!!!"
Else
End If
If Text2.Text = "-5" Then
Label5.Caption = " Nemam rijeci glup
si ko mazga!!!"
End If
End Sub

```

```

Private Sub Command2_Click()
Label3.Caption = "Koliko je 3 + 2"
Command8.Enabled = True
Text1.Text = ""
Text1.SetFocus
Command7.Enabled = False
End Sub

```

```

Private Sub Command3_Click()
Label3.Caption = "Koliko je 1 + 1"
Command9.Enabled = True
Text1.Text = ""
Text1.SetFocus
Command8.Enabled = False
End Sub

```

```

Private Sub Command4_Click()
Label3.Caption = "Koliko je 3 - 1"
Command10.Enabled = True
Text1.Text = ""
Text1.SetFocus
Command9.Enabled = False
End Sub

```

```

Private Sub Command5_Click()
Label3.Caption = "Koliko je 10 + 5"
Command11.Enabled = True
Text1.Text = ""
Text1.SetFocus

```

```

Command10.Enabled = False
End Sub

```

```

Private Sub Command6_Click()
Form2.Visible = True
End Sub

```

```

'Kod za Form2:
Private Sub Command1_Click()
End
'Prekidam rad aplikacije
End Sub

```

```

Private Sub Command2_Click()
Unload Me
'Povlacim samo form2
End Sub

```

```

Private Sub Command7_Click()
If Text1.Text = "10" Then
Label5.Caption = "Tacno"
Text2.Text = Text2.Text + 1
Else
Label5.Caption = "Netacno"
Text2.Text = Text2.Text - 1
End If
End Sub

```

```

Private Sub Command8_Click()
If Text1.Text = "5" Then
Label5.Caption = "Tacno"
Text2.Text = Text2.Text + 1
Else
Label5.Caption = "Netacno"
Text2.Text = Text2.Text - 1
End If

```

```

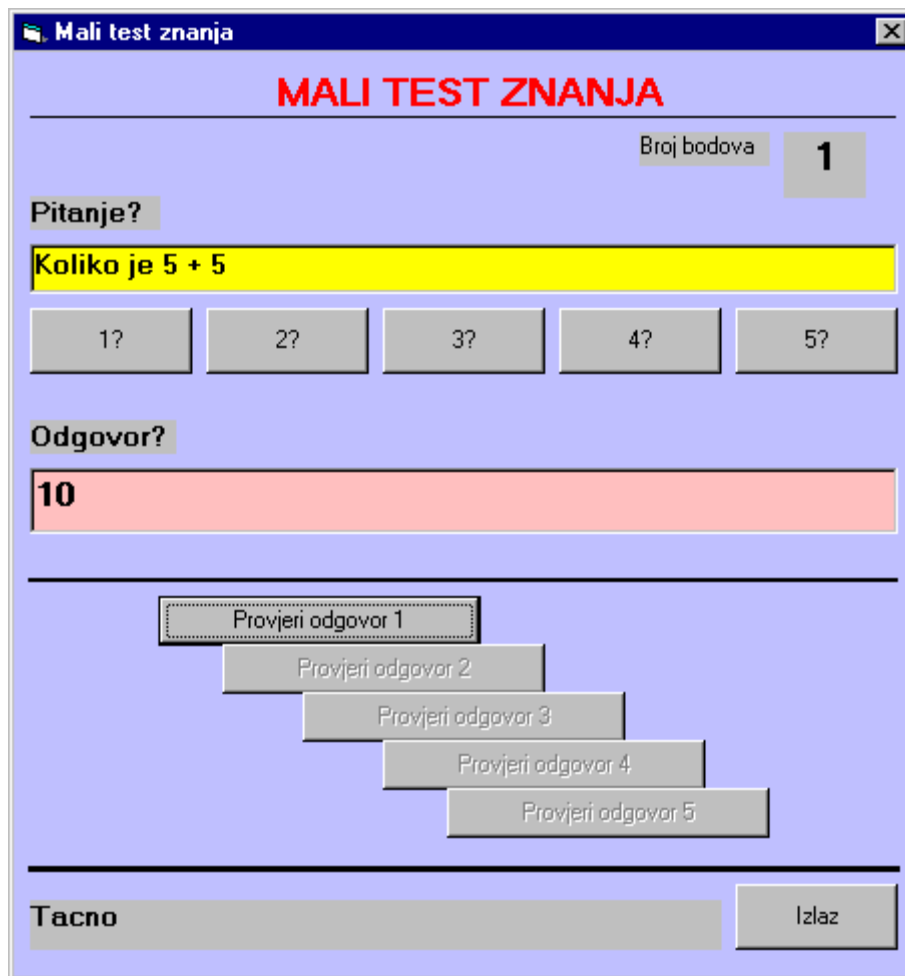
End Sub

```

```

Private Sub Command9_Click()
If Text1.Text = "2" Then
Label5.Caption = "Tacno"
Text2.Text = Text2.Text + 1
Else
Label5.Caption = "Netacno"
Text2.Text = Text2.Text - 1
End If
End Sub

```



#### **Primjer br. 4**

Cilj današnje vježbe je da se bolje upoznamo sa još nekim kontrolama visual basica.

- Timer – nam služi za tempiranje objekata vremenski – recimo nestajanje objekta nakon vremena koje mi odredimo ili mijenjanje nekog svojstva objekta koji izaberemo.
- Check box – nam služi za uključivanje i isključivanje objekata ili drugih kontrola u projektu
- Frame – okvir u koji možemo kreirati odvojene dijelove aplikacije.

#### **Private Sub Check1\_Click()**

```
If Check1.Value = 1 Then
' ako je checkbox uzključen onda
```

```
Frame1.Visible = True
'okvir1 neka je vidljiv
Else
' u drugom slučaju
```

```
Frame1.Visible = False  
'neka je okvir nevidljiv  
End If
```

**End Sub**

```
Private Sub Check2_Click()
```

```
If Check2.Value = 1 Then  
Frame2.Visible = True  
Else  
Frame2.Visible = False  
End If
```

**End Sub****Private Sub Command1\_Click()**

```
Dim fSize As Single  
'deklarišem veličinu fonta kao tip  
podatka single  
fSize = 1  
'početna veličina je jednaka jedinici  
Do  
'učini  
fSize = fSize * 1.2  
'sljedeću stvar: veličinu fonta pomnoži  
sa 1.2  
Me.Font.Size = fSize  
'ovo me znači forma na kojoj želimo  
izvesti vježbu.  
Me.Font = "Arial"  
'Određujem font
```

```
Me.Print "Haris"  
'Naredba print ispisuje naslov na  
formi.  
Loop Until fSize > 100  
'loop znači krug i ovo bi značilo vrti  
isto dok velicina fonta ne pređe 100
```

**End Sub****Private Sub Command2\_Click()**

```
Set Command2.Container = Frame2  
'premjestam objekat u ovom slučaju  
komandni dugmić u drugi okvir
```

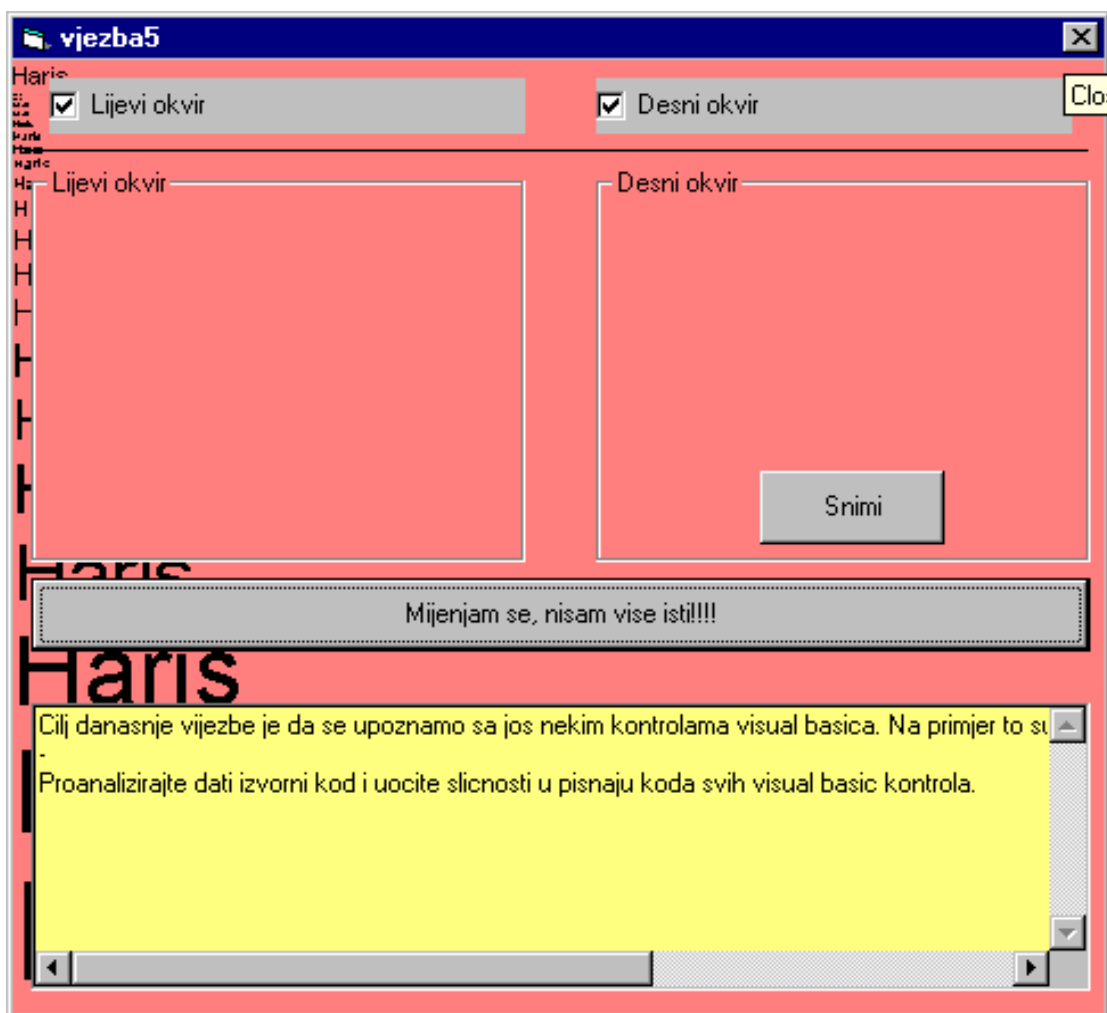
**End Sub****Private Sub Timer1\_Timer()**

```
Command1.Caption = "Mijenjam se,  
nisam vise isti!!!!"  
'definisem sta će se desiti posto timer  
okrene interval koji smo mu zadali
```

**End Sub****Private Sub Timer2\_Timer()**

```
Command1.Caption = "Opet sam isti,  
Klikni na mene!!!!"
```

**End Sub**



### **Promjer br. 5**

Rad sa datotekama, stvaranje datoteke iz programa, citanje podataka iz nje i upisivanje novih podataka u nju.

'stvaram svoj tip podataka

#### **Private Type brojevi**

'Deklarisem svoj tip podataka

prvi As String \* 20

'tip podataka prvi sa dvadeset mijesta

drugi As String \* 20

"tip podataka prvi sa dvadeset mijesta

#### **End Type**

#### **Private Sub Command1\_Click()**

'izvrsni kod dugmica kreiraj fajl

Dim sA As String

Dim emp As brojevi

'deklarisem varijable koje cu koristiti

Open App.Path & "/test.txt" For

Binary As #1

'stvaram fajl pod nazivom test.txt

emp.prvi = 20

emp.drugi = 30

'varijabli emp dodajem vrijednosti

Put #1, , emp

```
'ucitavam te vrijednosti u fajl test.txt
Close #1
'zatvaram fajl
MsgBox "Datoteka je kreirana i zove se
test.txt", vbInformation, "Obavijest"
'porukom obavjestavam korisnika da je
datoteka kreirana
End Sub
```

#### **Private Sub Command2\_Click()**

```
Dim sB As String
Dim sC As String
'Deklarisem varijable
Open App.Path & "/test.txt" For
Binary As #1
'Otvaram datoteku
sB = Space(2)
sC = Space(20)
'kazem varijablama koji dio datoteke
ce ucitati
Get #1, , sB
```

```
Get #1, , sC
'dobavljam iz datoteke podatke pod
vrijednosti varijabli
Text1.Text = sB
Text2.Text = sC
'prikazujem sadrzaj datoteke u dva
textbox-a
Close #1
'zatvaram fajl
End Sub
```

#### **Private Sub Label1\_Click()**

```
Label1 = Text1 * Text2
'mnozim vrijednosti ucitane iz datoteke
u textboxo-va
End Sub
```

#### **Private Sub Form\_Click()**

```
Printer.Print Label1.Caption &
"_____ovo je rezultat"
End Sub
```

### **Primjer br. 6**

Visual Basic je moćan alat za upravljanje bazama podataka, ovaj primjer je više rad sa fajlom nego sa bazom podataka, ali u ovom primjeru posmatračemo ovu tekstualnu datoteku kao bazu podataka. Visual Basic sadrži i čarobnjake koji nam pomažu da kreiramo i upravljamo bazama podataka, o tome je bilo riječi u poglavlju o Visual Data Menager-u.

#### **Option Explicit**

```
'deklarisem glavne varijable, kad
koristim option explicit
'onda ove varijable su validne za cijeli
projekat
Dim baza As info
Dim test As Integer
Dim recordlen As Long
Dim tekucislog As Long
Dim poslednjislog As Long
```

#### **Public Sub Savetekucislog()**

```
'popunjavam tekuci slog podacima
baza.ime_i_prezime =
txttime_i_prezime
baza.adresa = txtadresa
baza.email = txtemail
baza.index = txtindex
'snimam tekuci slog u datoteku
Put #test, tekucislog, baza
```

#### **End Sub**

#### **Public Sub Showtekucislog()**

```

Get #test, tekucislog, baza
'dobavljam podatke iz datoteke

txttime_i_prezime =
Trim(baza.ime_i_prezime)
txtadresa = Trim(baza.adresa)
txtemail = Trim(baza.email)
txtindex = Trim(baza.index)
End Sub
Private Sub cmdDel_Click()
'brisem tekuci slog
Dim dirresult
Dim tmpptest
Dim tmpbaza As info
Dim recnum As Long
Dim tmprecnum As Long
'
'potvdjivanje brisanja
If MsgBox("Dali zaista zelis obrisati
oznaceni slog?", 4) <> 6 Then
txtindex.SetFocus
Exit Sub
End If

If Dir("studenti.tmp") = "studenti.tmp"
Then
Kill "studenti.tmp"
End If

tmpptest = FreeFile
Open "studenti.tmp" For Random As
tmpptest Len = recordlen
recnum = 1
tmprecnum = 1

Do While recnum < poslednjislog + 1
If recnum <> tekucislog Then

Get #test, recnum, tmpbaza
Put #tmpptest, tmprecnum, tmpbaza
tmprecnum = tmprecnum + 1
End If

recnum = recnum + 1
Loop

```

```

'zatvaram datoteku
Close test
Kill "studenti.dat"

Close tmpptest
'reimenujem datoteku studenti.tmp u
studenti.dat
Name "studenti.tmp" As "studenti.dat"
'ponovo je otvaram
test = FreeFile
Open "studenti.dat" For Random As
test Len = recordlen

poslednjislog = poslednjislog - 1

If poslednjislog = 0 Then
poslednjislog = 1

If tekucislog > poslednjislog Then
tekucislog = poslednjislog
End If
End If

Showtekucislog

txtindex.SetFocus
End Sub

Private Sub cmdExit_Click()
End
'zavrsavam rad aplikacije
End Sub

Private Sub cmdNew_Click()
'snimam tekuci slog
Savetekucislog
'dodajem prazan prostor u polja da
dodam novi slog
poslednjislog = poslednjislog + 1
baza.ime_i_prezime = ""
baza.adresa = ""
baza.email = ""
baza.index = ""

Put #test, poslednjislog, baza

```

```
tekucislog = poslednjislog
```

```
Showtekucislog
```

```
txtindex.SetFocus
```

**End Sub**

**Private Sub cmdNext\_Click()**

```
If tekucislog = poslednjislog Then
  MsgBox "Ovo je poslednji slog u bazi!",
  vbOKOnly, "baza studenata"
Else
  Savetekucislog
  tekucislog = tekucislog + 1
  Showtekucislog
End If
```

**End Sub**

Private Sub cmdPrev\_Click()

```
If tekucislog = 1 Then
  MsgBox "Ovo je prvi slog u bazi!",
  vbOKOnly, "baza studenata"
Else
  Savetekucislog
  tekucislog = tekucislog - 1
  Showtekucislog
End If
```

**End Sub**

**Private Sub cmdSearch\_Click()**

```
Dim ime_i_prezimetosearch As String
Dim found As Integer
Dim recnum As Long
Dim tmpbaza As info
```

```
'unosim ime i prezime studenta za
potragu
```

```
ime_i_prezimetosearch =
InputBox("Upisi ime i prezime
studenta kojeg trazis?", "brza potraga
baze")
'ako korisnik nije unio nista prekidam
potragu
```

```
If ime_i_prezimetosearch = "" Then
  txttime_i_prezime.SetFocus
  Exit Sub
End If
```

```
'koristim UCASE funkciju da
konvertujem slova u UPPERCASE
ime_i_prezimetosearch =
UCase(ime_i_prezimetosearch)
```

```
found = False
```

```
'potraga za imenom i prezimenom
studenta
For recnum = 1 To poslednjislog
  Get #test, recnum, tmpbaza
  If ime_i_prezimetosearch =
  UCase(Trim(tmpbaza.ime_i_prezime))
  Then
    found = True
    Exit For

  End If
Next
```

```
'prikazujem pronadjenog studenta
```

```
If found = True Then
  Savetekucislog
  tekucislog = recnum
  Showtekucislog
Else
  MsgBox "student nije pronadjen !",
  vbOKOnly, "baza studenata"
End If
```

```
txttime_i_prezime.SetFocus
```

**End Sub****Private Sub Form\_Load()**

```
'calculate the L.O.F
recordlen = Len(baza)
'get the next file adresa
test = FreeFile
'otvaram datoteku
Open "studenti.dat" For Random As
test Len = recordlen
'vrsim update podataka
tekucislog = 1
'pronlazim poslednji slog
poslednjislog = FileLen("studenti.dat")
/ recordlen

If poslednjislog = 0 Then
poslednjislog = 1
End If
'izvrsavam Showtekucislog proceduru
Showtekucislog
```

**End Sub****Private Sub Form\_Unload(Cancel As Integer)**

```
Close test
End Sub
```

Ovaj primjer sadrži i jedan modul a u njega upišite sljedeći kod.

**Option Explicit****Type info**

```
index As String * 5
ime_i_prezime As String * 50
adresa As String * 50
email As String * 100
```

**End Type**

broj indexa	ime_i_prezime	adresa	email
192	Gondzo Haris	Boracka 32	haris@info.ba

OBRISI   NOVI   PRETH   SLJEDECI   POTRAGA

IZLAZ

**Primjer br. 7**

U ovom primjeru ćemo pokazati kako se može sa malo truda kreirati jednostavan tekstualni procesor. Na slici imate raspored kontrola i objekata.

**Option Explicit****Private Sub Check1\_Click()**

```
'izvorni kod za opciju bold
If Check1.Value = 1 Then
rtfEdit.SelBold = True
Else
rtfEdit.SelBold = False
End If
```

**End Sub****Private Sub Check2\_Click()**

```
'izvorni kod za opciju Italic
If Check2.Value = 1 Then
rtfEdit.SelItalic = True
Else
rtfEdit.SelItalic = False
End If
```

**End Sub****Private Sub Check3\_Click()**

```
'izvorni kod za opciju Underline
If Check3.Value = 1 Then
rtfEdit.SelUnderline = True
Else
rtfEdit.SelUnderline = False
End If
```

**End Sub****Private Sub cmd\_Click()**

```
Dim x As String
'deklarisem varijablu x koja ce nositi
vrijednost unesenu u inputbox
x = InputBox("Koju rijec zelis da dodas
u listu?", "dodavanje u listbox")
'kreiram inputbox
List1.AddItem x
```

```
'Listi dodajem stavku x tj uneseni
string iz inputbox-a
```

**End Sub****Private Sub cmdCut\_Click()**

```
Dim dokument
Dim seltext, duzina
'zadržavam fokus na dokumentu
rtfEdit.SetFocus
'pribavljam radne parametre
dokument = rtfEdit.Text
seltext = rtfEdit.SelStart
duzina = rtfEdit.SelLength
'izrezujem selektovani text u
clipboard
Clipboard.SetText Mid$(dokument,
seltext + 1, duzina)
```

```
dokument = Left$(dokument,
seltext) + _
Mid$(dokument, seltext + duzina
+ 1)
rtfEdit.Text = dokument
rtfEdit.SelStart = seltext
```

**End Sub****Private Sub cmdCopy\_Click()**

```
'zadržavam fokus na dokumentu
rtfEdit.SetFocus
'kopiram selektovani text u
clipboard
Clipboard.SetText rtfEdit.seltext
```

**End Sub****Private Sub cmdPaste\_Click()**

```

'Izvorni kod za lijepljenje text-a iz
clipboard-a
Dim dokument, rezultat
Dim seltext, duzina
'zadržavam fokus na dokumentu
rtfEdit.SetFocus
'uzimam radne parametre
dokument = rtfEdit.Text
seltext = rtfEdit.SelStart
duzina = rtfEdit.SelLength
'Vracam takst iz clipboard-a na
mjesto kursora
rezultat = Clipboard.GetText()
dokument = Left$(dokument,
seltext) + rezultat + _
Mid$(dokument, seltext + duzina
+ 1)
rtfEdit.Text = dokument
rtfEdit.SelStart = seltext +
Len(rezultat)
End Sub
Private Sub cmdEnd_Click()
End
'završavam rad aplikacije
End Sub

```

#### **Private Sub cmdPrint\_Click()**

```

Printer.Print rtfEdit.Text
'stampam sadržaj dokumenta
End Sub

```

#### **Private Sub cmdSave\_Click()**

```

rtfEdit.SaveFile ("c:\test.txt")
'snimam datoteku kao test.txt u
korijenski direktorij
End Sub

```

#### **Private Sub Form\_Load()**

```

Label1.MousePointer = vbSizeAll
'na pokretanju forme definisem
pokazivac misa kada korisnik dodje
iznad labela1

```

#### **End Sub**

#### **Private Sub Timer1\_Timer()**

```

Label1.Caption = "Ako ste dovoljno
zainteresovani"
'dodajem naslov label-u koji ce
nastupiti nakon odredjenog intervala
End Sub

```

#### **Private Sub Timer2\_Timer()**

```

Label1.Caption = "Studij informatike je
pravi studij za VAS"
'dodajem naslov label-u koji ce
nastupiti nakon odredjenog intervala
End Sub

```

#### **Private Sub cmdFind\_Click()**

```

Dim pos As Integer

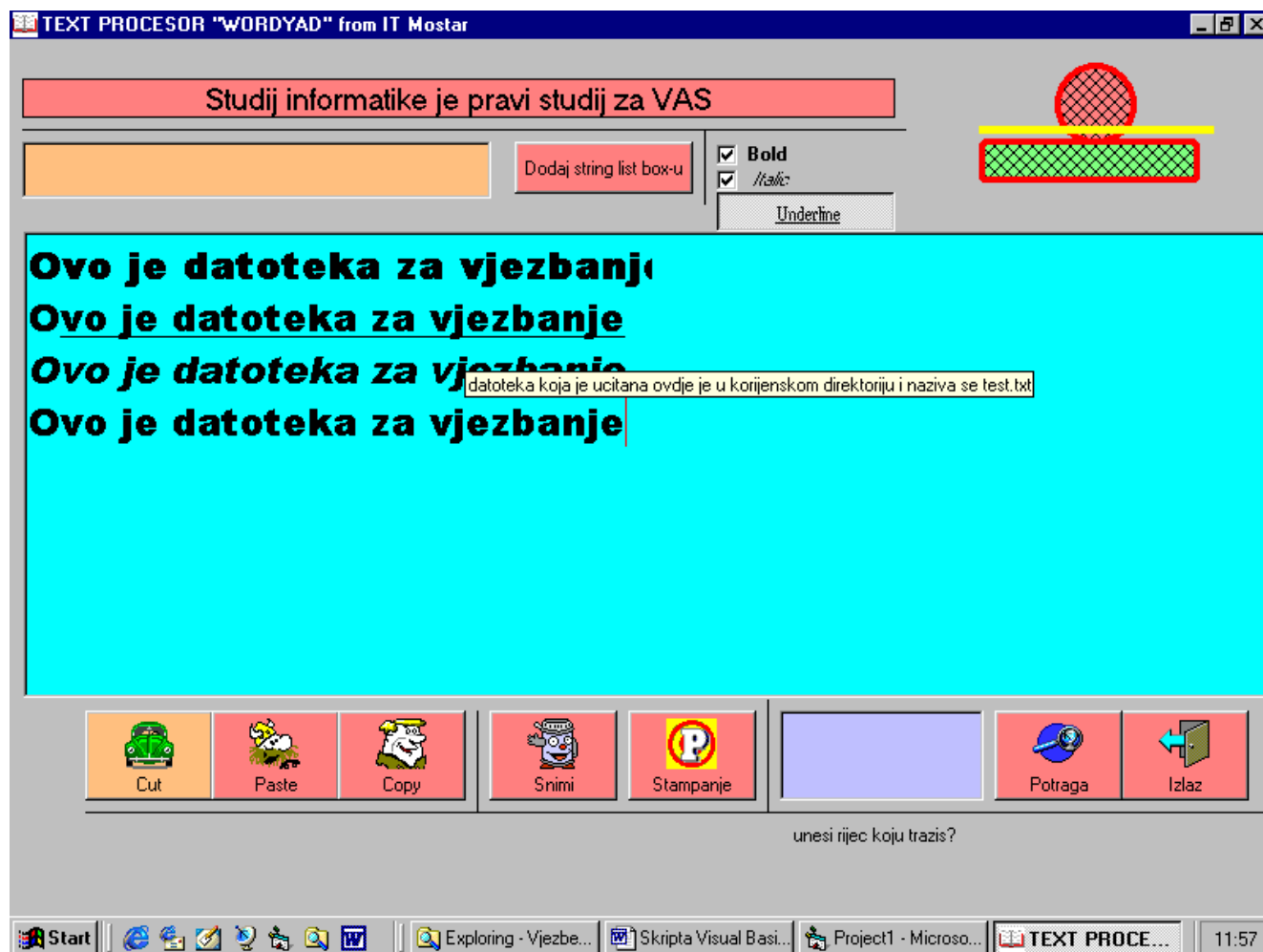
pos = Form1.rtfEdit.SelStart
pos = InStr(pos + 2,
Form1.rtfEdit.Text, txtSearch.Text)
If pos = 0 Then
Form1.rtfEdit.SelStart = 0
Form1.rtfEdit.SelLength = 0
Else
Form1.rtfEdit.SelStart = pos - 1
Form1.rtfEdit.SelLength =
Len(txtSearch.Text)
End If

```

```

Form1.rtfEdit.SetFocus
End Sub

```



### Primjer br. 8

U ovom primjeru vidjeti ćemo kako radi «Select Case», vrlo korisna petlja prilikom programiranja uslova i odabira.

```
Private Sub cmdProvjera_Click()
Dim Number 'Deklarisem varijablu
```

```
Number = txtNumber ' Kazem racunaru da je varijabla Number
'jednaka vrijednost koju korisnik unese u textbox.
```

```
Select Case Number
```

```
Case 1 To 7 ' Broj manji od broja 8.
```

```
lblRezultat.Caption = "Ovaj je broj manji od 8!!!!"
```

Case 8 ' Broj jednak broju 8.

```
lblRezultat.Caption = "Ovaj je broj jednak broju 8!!!! "
```

Case 9

```
lblRezultat.Caption = "Ovaj broj je veci od 8!!!! "
```

Case Is > 100

```
lblRezultat.Caption = "Stvarno si poludio Ovaj je broj veci " & txtNumber / 8 & "  
Putu od broja 8"
```

Case Else ' Druge vrijednosti.

```
lblRezultat.Caption = "Uneseni broj dvocifren i onda logicno veci od 8!!!!"
```

End Select

End Sub

