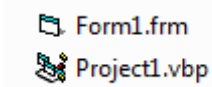
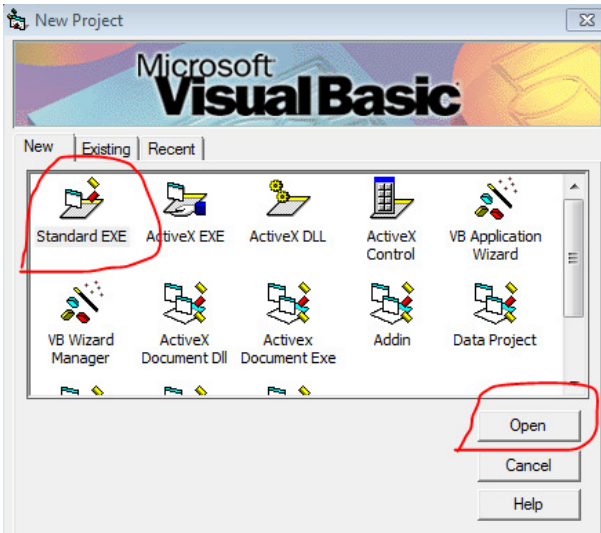


محیط برنامه نویسی Visual Basic(VB)

- ۱- بعد از اجرای برنامه گزینه Standard EXE را انتخاب کرده و open را می زنیم.
- ۲- عادت کنید نخست در مسیری دلخواه پوشه ای برای هر برنامه ایجاد کنید و پروژه را در آن ذخیره کنید. دو فایل با پسوندهای .frm برای فرم و .vbp برای پروژه ذخیره می شوند.



- ۳- مهم ترین عنصر برنامه نویسی VB، کنترل فرم (Form) می باشد. مهم ترین خصوصیات (property) کنترل فرم و اغلب کنترلهای دیگر عبارتند از:



• Name: نام کنترل

• Caption: عنوان ظاهری

• BackColor: رنگ پس زمینه

• ForeColor: رنگ زمینه

• Font: تنظیمات فونت

• Width: پهنا

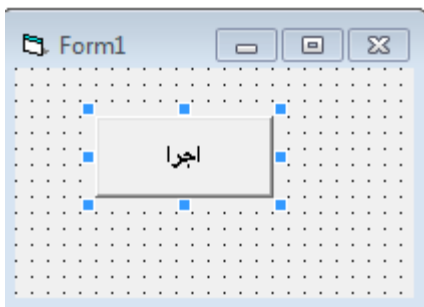
• Height: ارتفاع

• Left: فاصله از چپ

• Top: فاصله از بالا

• Visible: دو حالت دارد؛ True نمایان و False مخفی

• Enabled: دو حالت دارد؛ True فعال و False غیر فعال



- ۴- دستورات برنامه ها اغلب در رویداد کلیک کنترل دکمه فرمان (Command Button) اجرا می شوند. بعد از قرار دادن کنترل دکمه فرمان روی فرم، عنوان (Caption) آنرا تنظیم کرده و سپس با کلیک روی آن وارد بخش برنامه نویسی آن می شویم. بصورت خودکار دستورات زیر نوشته می شوند و دستورات ما بین آنها قرار می گیرند:

```
Private Sub Command1_Click()  
    دستورات برنامه را اینجا می نویسیم  
End Sub
```



- ۵- برای شروع اجرای برنامه کلید F5 را می زنیم یا روی آیکن مثلث (Start) کلیک می کنیم.

آشنایی با چند دستور ساده

Print : برای نمایش داده ها روی فرم که به صورت زیر نوشته می شود:

Print "متن دلخواه"

```
Private Sub Command1_Click() → در یک رویداد دکمه فرمان  
    Print "اولین برنامه من"  
End Sub
```

تمرین: در یک پروژه جدید (ذخیره در پوشه vb2) سه دکمه روی فرم قرار دهید که با کلیک روی هریک به ترتیب نام، رشته و سن خود را روی فرم نشان دهد.

انواع داده ها: برای ذخیره انواع مختلف داده ها در برنامه لازم است با انواع آنها آشنا شویم

متغیر (Variable): مکانی نامدار از حافظه برای نگهداری موقت اطلاعات از نوع معین

دسته	نام	پسوند	محدوده	مصرف حافظه بر حسب بایت	کاربرد
عدد صحیح	Byte		0 → 255	۱	عدد صحیح
	Integer	%	-32768 → +32767	۲	
	Long	&	حدود ۲ میلیارد (+ یا -)	۴	
عدد اعشاری	Single	!	دقت ۶ رقم	۴	عدد اعشاری
	double	#	دقت ۱۰ رقم	۸	
منطقی	Boolean		True , False	۲	درست یا نادرست
تاریخ	Date			۸	تاریخ و زمان
عدد بزرگ	Currency	@		۸	عدد بسیار بزرگ (واحد پول)
رشته ای با طول ثابت	String * تعداد		۶۵۴۰۰ کارکتر	طول رشته	مجموعه ای از کارکترها
رشته ای با طول متغیر	string		تا ۲ میلیارد کارکتر	طول رشته + ۱۰	مجموعه ای از کارکترها
شیئی	Object			۴	ذخیره شیئی
عمومی عددی	Variant			۱۶	هر نوع عددی دلخواه
عمومی متن	Variant			طول رشته + ۲۲	رشته کارکتر

نحوه تعریف متغیر:

نام نوع داده AS نام متغیر Dim

علامت پسوند نام متغیر Dim

مثال: تعریف متغیری از نوع عدد صحیح → `dim a% as integer` یا `Dim a as integer`

نام متغیر	درست یا غلط	دلیل نادرست بودن
ABC	درست	
Ali23	درست	
P2p	درست	
15M	غلط	با عدد شروع شده است
My_Age	درست	
_New	درست	
Ab_	درست	
My var	غلط	فاصله قابل قبول نیست
for	غلط	این کلمه رزرو شده است

قواعد نامگذاری:

- فقط از حروف الفبا (کوچک و بزرگ)، ارقام ۰ تا ۹ و علامت زیر خط (_) استفاده شود و علائم دیگر و فاصله مجاز نیست.
- با رقم شروع نیشود
- از کلمات خاص و ریزو شده زبان برنامه نویسی استفاده نیشود.
- حداکثر طول ۲۵۵ کارکتر

مثال: برای ذخیره داده های زیر متغیر مناسب (نام و نوع) تعریف کنید:

کاربرد	محدوده مقادیر	تعریف متغیر
--------	---------------	-------------

Dim Mowjodi AS Long	عدد دلخواه	موجودی حساب
Dim Days AS Byte	۱ تا ۳۱	تعداد روزهای یک ماه
Dim Nomre AS single	۰ تا ۲۰ اعشاری	نمره این درس
Dim MyName AS string*30	رشته ای با طول ۳۰	نام خود
Dim Pelak AS string*11	رشته ای با طول ۱۱	پلاک ماشین
Dim melli AS Long	عدد بزرگ	شماره ملی
Dim Tel as Long	۶۳*****	تلفن منزل بدون پیش شماره
Dim Mobile as String *11	رشته ای با طول ۱۱	موبایل
Dim Lamp as Boolean	روشن(True) یا خاموش(False)	وضعیت یک لامپ

مقدار = [نوع] نام *Const*
Const pi AS single = 3.14
Const N = 20

ثابت (Constant): مکانی از حافظه برای ذخیره اطلاعات که در طول اجرای برنامه ثابت بوده و تغییر نمی کند. نحوه تعریف بصورت زیر است (عبارت داخل [] اختیاری است):
نکات

- مقادیر رشته ای داخل علامت " " نوشته می شوند. مثال: S = "Mohamad"
- مقادیر تاریخ داخل علامت # # نوشته می شوند. مثال: D = #23/1/2013#

مثال	مفهوم و کاربرد	عملگر
$2^3 = 8$	توان	^
$6 * 4 = 24$	ضرب	*
$14 / 4 = 3.5$	تقسیم	/
$5 + 6 = 11$	جمع	+
$8 - 3 = 5$	تفریق	-
$14 \setminus 4 = 3$	تقسیم صحیح	\
$14 \bmod 4 = 2$	باقیمانده	Mod
"hello" & "Ali" = "helloAli"	ترکیب یا الحاق رشته ها	& یا +

آشنایی با عملگرهای محاسباتی و رشته ای (پردازش داده ها)

اولویت عملگرها: ترتیب اجرای انواع عملگرها در یک عبارت ریاضی بصورت زیر است:

$$A = 5 + 6 * 3 + 8 \bmod 3 = 5 + [18] + [2] = [23] + [2] = 25$$

عملگر	ترتیب اجرا
عبارت داخل پرانتز ()	0
^	۱
- (تفریق یکانی مانند -5)	۲
*, /	۳
\	۴
Mod	5
+, -	6

دستور انتساب: اساسی ترین دستور است و مقداری را در متغیری قرار می دهد.

مقدار = نام متغیر

مقدار ۵ در متغیر ذخیره می شود $\rightarrow A = 5$

دستورات ورودی: خواندن داده ها

۱- استفاده از کنترل TextBox: اگر این کنترل را روی فرم قرار دهیم با دستور زیر داده وارد شده در آن را می خوانیم:

Dim A As Integer

مقدار وارد شده در کادر متن را در متغیر قرار می دهد $\rightarrow A = \text{Text1.Text}$

مثال: شعاع دایره را بخوانید و محیط و مساحت آنرا بدست آورده و چاپ کنید.
حل:

۱- در محیط برنامه پروژه جدیدی ایجاد کرده و ذخیره کنید

۲- یک کنترل دکمه فرمان (Comand) و کادر متن (TextBox) روی فرم قرار دهید

۳- روی دکمه کلیک کنید تا وارد بخش برنامه نویسی شوید و در آن دستورات زیر را بنویسید

```
Const pi = 3.14  $\rightarrow \pi$  عدد
Dim r as single  $\rightarrow$  شعاع
Dim s as single  $\rightarrow$  مساحت
Dim m as single  $\rightarrow$  محیط
r = Text1.Text  $\rightarrow$  خواندن داده وارد شده برای شعاع
S = pi * r^2  $\rightarrow$  محاسبه مساحت
M = 2*pi*r  $\rightarrow$  محاسبه محیط
Print "mosahat = "; s  $\rightarrow$  نمایش مساحت
print "mohit =" ; m  $\rightarrow$  نمایش محیط
```

مثال: زمان را برحسب ثانیه خوانده و تعداد ساعت و دقیقه و ثانیه آنرا محاسبه و نمایش دهید

Private Sub Command1_Click()

```
Dim T As Long  $\rightarrow$  زمان ورودی
Dim h As Integer  $\rightarrow$  ساعت
Dim m As Integer  $\rightarrow$  دقیقه
Dim s As Integer  $\rightarrow$  ثانیه
Dim r As Integer  $\rightarrow$  متغیر کمکی
T = Text1.Text  $\rightarrow$  ورودی
h = T / 3600  $\rightarrow$  محاسبه ساعت
r = T Mod 3600  $\rightarrow$  محاسبه باقیمانده از ساعت
m = r / 60  $\rightarrow$  محاسبه دقیقه
s = r Mod 60  $\rightarrow$  محاسبه باقیمانده از دقیقه = ثانیه
Print h & ":" & m & ":" & s
```

End Sub

نکته: برای نوشتن توضیحات در برنامه علامت (') را نوشته سپس توضیحات را می نویسیم:

Dim T As Long

زمان ورودی

مراحل نوشتن برنامه

- ۱- تعریف متغیرهای لازم (نام و نوع بر اساس مقادیری که ذخیره می کنند)
- ۲- تعیین مقادیر ورودی
- ۳- دستورات پردازشی برنامه (محاسبات و)
- ۴- چاپ نتایج

روشهای ورود اطلاعات به برنامه

۱- با دستور انتساب مثل: $S = 150000$

۲- استفاده از کنترل کادرمتن (TextBox)

- قرار دادن کنترل روی فرم و نامگذاری در صورت نیاز
- دستور خواندن محتویات آن (با فرض اینکه نام آن Text1 باشد):

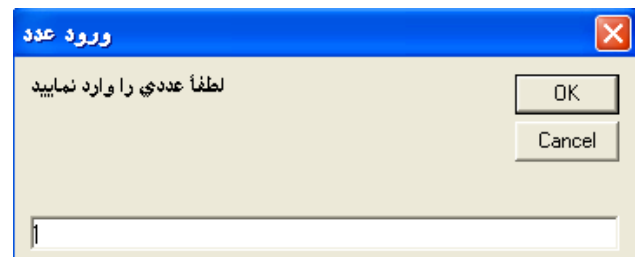
`Text1.Text = نام متغیر`

مانند: `S = Text1.Text` یعنی مقدار وارد شده در کادر آن در متغیر s ذخیره شود.

۳- استفاده از دستور inputbox که در پنجره ظاهر شده اطلاعات وارد می شود
("عنوان پنجره"، "پیامی که در پنجره ظاهر شود") = `inputBox(نام متغیر`

`Dim a as integer`

`A= inputBox("ورود عدد", "لطفاً عددی را وارد کنید")`



***نکته:** با توجه به اینکه خروجی این دستور بصورت رشته ای است. پس برای جلوگیری از خطا باید خروجی آنرا در متغیر رشته ای ذخیره کرده و سپس توسط دستور زیر آنرا به عدد تبدیل نماییم.

`Dim s as string`

`Dim a as integer`

`S= inputbox("لطفاً عددی را وارد کنید")`

`A= val(s) → تبدیل رشته به عدد`

روشهای نمایش نتیجه

۱- دستور `print`

۲- نمایش نتیجه در یک `textbox`

`Text1.Text = مقدار نتیجه`

۳- نمایش نتیجه در یک `Label`

`Label1.caption = مقدار نتیجه`

توجه: کنترل `Label` فقط جهت نمایش متن در آن بکار می رود و نمی توان مقداری در آن وارد کرد. متن مورد نظر توسط مشخصه `caption` تعیین می شود.

برخی مشخصه های مهم در این کنترل:

- `Caption` برای متن
- `Alignment` برای تعیین ترازبندی متن داخل آن
- `Autosize`: اگر مقدار `true` به آن نسبت دهیم خود را به اندازه متن داخل آن بزرگ (کوچک) می کند
- `Wordwrap`: اگر مقدار `true` به آن نسبت دهیم اگر متن در آن جا نشود، متن را شکسته و ادامه در خط بعدی می آید.

کاربرد label: بیشتر برای نمایش عناوین بکار می رود. مثلاً نام، نام خانوادگی، نام پدر و...

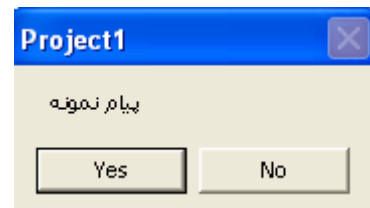
۴- روش دیگر نمایش اطلاعات استفاده از دستور MsgBox است.

متغیر نتیجه MsgBox

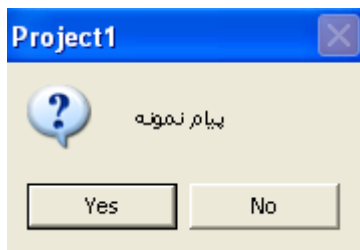
نکته: مقدار نتیجه بهتر است به شکل رشته ای درآید برای اینکار از دستور زیر استفاده می شود:

تبدیل عدد به رشته → (متغیر غیر رشته ای) = CStr = متغیر رشته ای

سوال: چگونه می توان نوع دکمه های ظاهر شده در کادر پیام MsgBox را مشخص کرد؟
جواب: بعد از پیام مربوطه یک کاما(,) قرار داده و سپس می نویسیم VB..... و با انتخاب مورد دلخواه دکمه ها تعیین می شوند.



MsgBox "پیام نمونه", vbYesNo



سوال: چگونه می توان آیکن دلخواهی را در پنجره قرار داد؟
جواب: بعد از تعیین نوع دکمه ها علامت + قرار داده و سپس VB.... و یکی از گزینه ها را مطابق جدول صفحه ۶۱ کتاب را انتخاب می کنیم.

MsgBox "پیام نمونه", vbYesNo + vbQuestion

مثال: برنامه ای بنویسید که قد یک نفر را بر حسب فوت و اینچ بخواند و تعیین کند قد او بر حسب سانتی متر چقدر است؟

راهنمایی: هر اینچ = ۲.۵۴ سانتی متر و هر فوت برابر ۱۲ اینچ است.

حل:

```
Dim i As Single
Dim f As Integer
Dim a As Single, b As Single
i = Text1.Text 'خواندن اینچ
f = Text2.Text 'خواندن فوت
a = f * 12 + i 'تبدیل به اینچ
b = a * 2.54 'تبدیل به سانتی متر
Text3.Text = b
```

۱- ایجاد پروژه جدید

۲- ذخیره پروژه

۳- قرار دادن کنترل های لازم

تمرین ۱: تعداد روز را بخوانید و تعداد سال، ماه و روز آنرا بدست آورید
تمرین ۲: قد بر حسب سانتی متر را خوانده و مقدار فوت و اینچ آنرا محاسبه کنید
تمرین ۳: مسافت بین دو شهر (برحسب کیلومتر) و سرعت ماشین (برحسب کیلومتر بر ساعت) را از ورودی خوانده و تعیین نمایید چه مدت طول می کشد تا به مقصد برسید؟ (برحسب ساعت و دقیقه) ----- راهنمایی :
سرعت/مسافت = زمان

ساختارهای تصمیم گیری و دستورات کنترلی

۱- دستور شرطی if

در برنامه نویسی مواردی پیش می آید که بخواهیم دستور یا دستوراتی، هنگامی که شرط خاصی برقرار است، توسط برنامه به اجرا در آید. این مورد در زندگی روزمره نیز دیده می شود؛ به عنوان مثال " اگر فردا برف نیارد، مسابقه فوتبال برگزار می گردد." شرط مورد نظر نیاریدن برف است و عملی که قرار است انجام شود برگزاری مسابقه می باشد. در برنامه نویسی ویژوال بیسیک این شرط به صورت زیر نوشته می شود:

دستور مورد نظر Then شرط مورد نظر If

مثال:

مسابقه فوتبال برگزار می گردد Then فردا برف نیارد If

If شرط مورد نظر Then

دستور شماره ۱

دستور شماره ۲

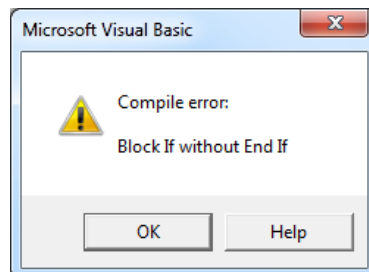
دستور شماره ۳

..... دستور شماره

End if

گردد.

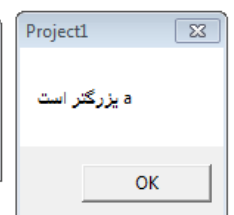
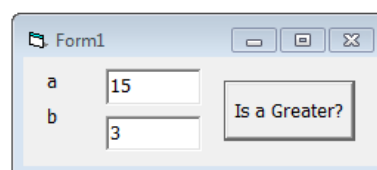
که حالت معمولی و ساده دستور می باشد که در این حالت اگر شرط برقرار باشد، دستور مورد نظر انجام می شود و در غیر این صورت اتفاقی نمی افتد. اگر بخواهیم در صورت برقراری شرط، چندین دستور اجرا شوند، از شکل زیر استفاده می کنیم:



توجه کنید که در پایان دستور ها باید کلمه End if را برای اتمام کار بنویسیم در غیر این صورت برنامه با پیغام خطا مواجه می

مثال: دو عدد بخوانید و مشخص کنید که آیا اولی بزرگتر است؟

```
Private Sub Command1_Click()  
    Dim a As Integer, b As Integer  
    a = Text1.Text  
    b = Text2.Text  
    If (a > b) Then MsgBox "a بزرگتر است"  
End Sub
```



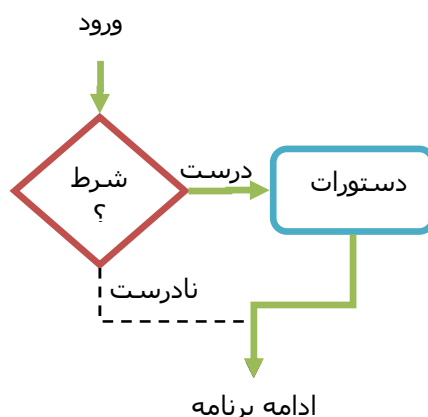
If (شرط) then

دستورات

End if

یا

If (شرط) then دستور



۲-دستور شرطی کامل بصورت If-else

گاهی اوقات نیاز داریم که در صورت برقرار بودن شرط خاصی یک سری از دستورات اجرا و در صورت برقرار نبودن آن شرط، دسته ای دیگر از دستورات اجرا گردند. به عنوان مثال "اگر فردا برفنبارد، مسابقه فوتبال برگزار می گردد، در غیر اینصورت مسابقه برگزار نخواهد شد."

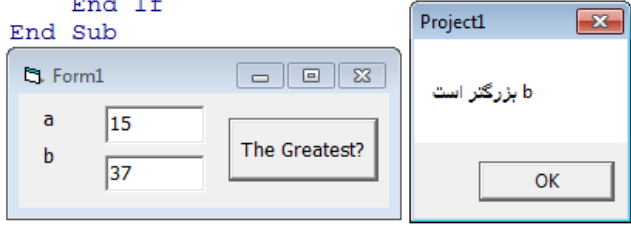
<p><i>If</i> شرط مورد نظر <i>Then</i></p> <p>دستور اول</p> <p><i>Else</i></p> <p>دستور دوم</p> <p><i>End if</i></p>	<p><i>If</i> فردا برف نبارد <i>Then</i></p> <p>مسابقه فوتبال برگزار می گردد</p> <p><i>Else</i></p> <p>مسابقه فوتبال برگزار نمی گردد</p> <p><i>End if</i></p>
---	--

👉 در این حالت از دستور، اگر شرط درست باشد، دستورات ۱ و در غیر اینصورت دستورات ۲ اجرا می شوند. دقت کنید از بین دستورات ۱ و ۲ فقط یکی اجرا خواهد شد.

مثال: دو عدد بخوانید و مشخص کنید که آیا اولی بزرگتر است یا دومی؟

```

Private Sub Command1_Click()
    Dim a As Integer, b As Integer
    a = Text1.Text
    b = Text2.Text
    If (a > b) Then
        MsgBox "a بزرگتر است"
    Else
        MsgBox "b بزرگتر است"
    End If
End Sub
                
```



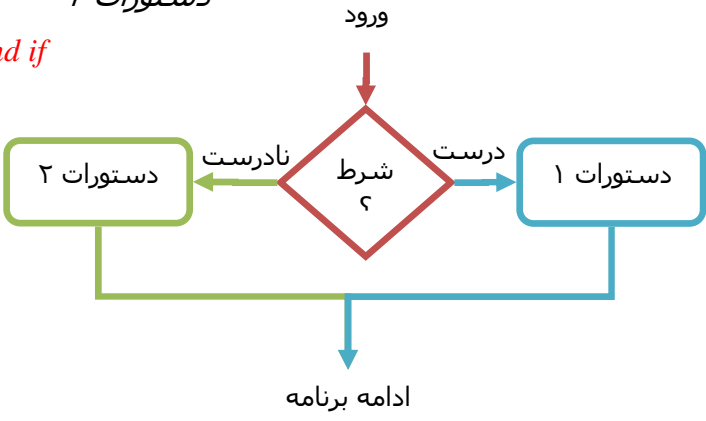
then (شرط) If

دستورات ۱

Else

دستورات ۲

End if



۳- **if...else بصورت ترکیبی:** هر گاه بخواهیم همزمان چندین شرط را بررسی کرده و بر اساس آنها دستورات متناظر را اجرا نماییم از این حالت استفاده می کنیم:

If (شرط ۱) *then*

دستورات ۱

Elseif (شرط ۲) *then*

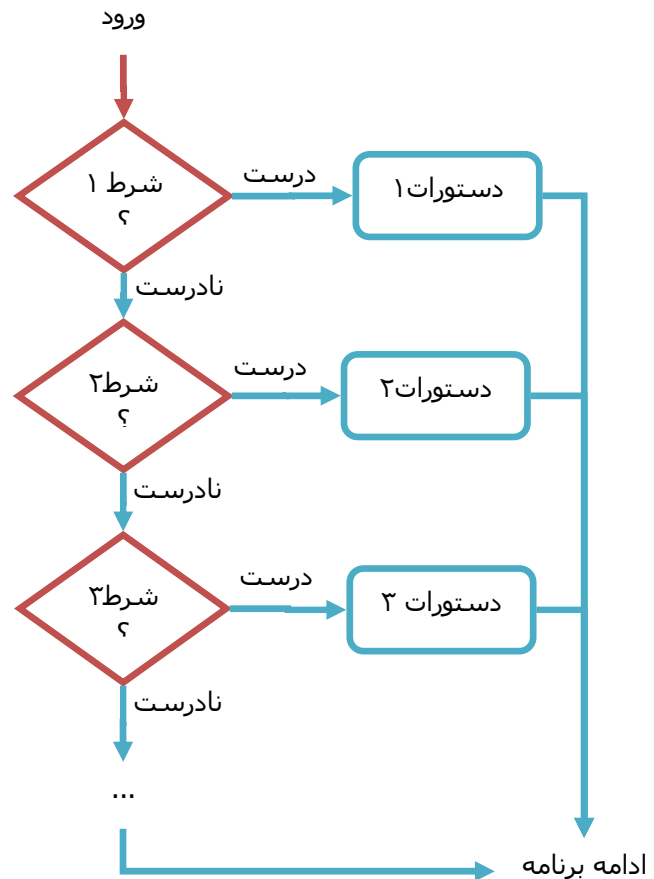
دستورات ۲

Elseif (شرط ۳) *then*

دستورات ۳

...

End if



مثال: سه عدد ضرایب معادله درجه ۲ را بخوانید و ریشه های آنرا با MsgBox نمایش دهید ($ax^2+bx+c=0$)

Dim a As Single, b As Single, c As Single, delta As Single, x1 As Single, x2 As Single

a = Text1.Text

b = Text2.Text

c = Text3.Text

*delta = b ^ 2 - 4 * a * c*

If (*delta* < 0) *Then*

MsgBox "no roots!", vbCritical

Elseif (*delta* = 0) *Then*

*x1 = -b / (2 * a)*

MsgBox "x1 = x2 = " & CStr(x1), vbInformation

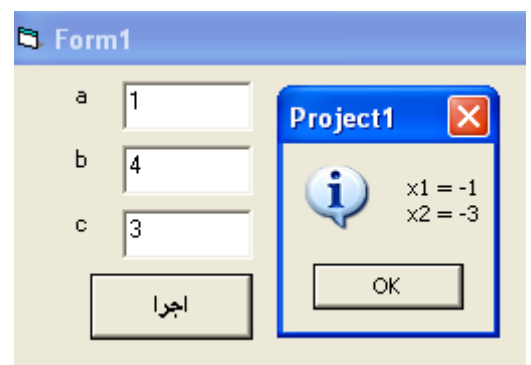
Else

*x1 = (-b + Sqr(delta)) / (2 * a)*

*x2 = (-b - Sqr(delta)) / (2 * a)*

MsgBox "x1 = " & CStr(x1) & vbCrLf & "x2 = " & CStr(x2), vbInformation

End If



تمرین: یک بازی حدس عدد طراحی کنید. کامپیوتر عددی در نظر میگیرد و شما با ۱۰ بار سوال آنرا حدس می زنید. کامپیوتر فقط با «کوچکتر» یا «بزرگتر» جواب می دهد. اگر تعداد حدس از ۱۰ بیشتر شد، پیغام باخت را با msgbox نمایش دهد.

```

Dim Number As Integer → عدد اصلی سیستم
Dim Guess As Integer → شمارش تعداد حدسها
Dim x As Integer → عدد حدسی ما
Private Sub cmdGuess_Click() → دکمه حدس
    x = Val(Text1.Text) → خواندن حدس خود
    If (Number > x) Then → مقایسه حدس خود با عدد کامپیوتر
        Print CStr(Guess) & " " & CStr(x) & " " & "بزرگتر"
    ElseIf (Number < x) Then
        Print CStr(Guess) & " " & CStr(x) & " " & "کوچکتر"
    Else
        Print CStr(Guess) & " " & CStr(x) & " " & "آفرین"
    End If
    cmdGuess.Enabled = False → غیر فعال کردن دکمه حدس
    Guess = Guess + 1 → شمارش تعداد حدسهای خود
    If (Guess > 10) Then → اگر تعداد حدس از ۱۰ بیشتر شود
        MsgBox "تو باختی"
        cmdGuess.Enabled = False
    End If
End Sub

```

دکمه شروع مجدد → Private Sub cmdNew_Click()

```

Randomize → آماده سازی برای عدد تصادفی
Number = 1 + 100 * Rnd() → عدد تصادفی کمتر از ۱۰۰
Guess = 1 → تعداد حدس
Form1.Cls → پاک کردن فرم
Print "حدس عدد نتیجه" → چاپ پیام خروجی
Print "====="
cmdGuess.Enabled = True → فعال کردن دکمه حدس
End Sub

```

در این تمرین از توابع زیر استفاده شده است:

- Rnd() برای تولید عدد تصادفی بین صفر و یک
- CStr() برای تبدیل عدد به رشته کارکتر

Private Sub Command1_Click()

```

Dim nomre As Single
nomre = Val(Text1.Text)
If (nomre > 20) Then
    Label2.Caption = "نمره اشتباه است"
ElseIf (nomre > 18) Then
    Label2.Caption = "آفرین"
ElseIf (nomre >= 15) Then
    Label2.Caption = "خوب است"
ElseIf (nomre >= 12) Then
    Label2.Caption = "متوسط"
ElseIf (nomre >= 10) Then
    Label2.Caption = "ضعیف"
Else
    Label2.Caption = "خیلی ضعیف!"
End If
End Sub

```

تمرین: نمره دانش آموز را خوانده و بر اساس آن پیام مناسب چاپ کنید:

- نمره بین ۱۸ تا ۲۰: آفرین
- نمره بین ۱۵ تا ۱۸: خوب است
- نمره بین ۱۲ تا ۱۵: متوسط
- نمره بین ۱۰ تا ۱۲: ضعیف
- نمره کمتر از ۱۰: خیلی ضعیف
- غیر از موارد بالا: نمره اشتباه

تمرین: عدد یک رقمی را خوانده سپس آنرا به حروف در یک MsgBox نشان دهید.



آشنایی کنترل دکمه رادیویی RadioButton

این کنترل نیز شبیه کادر انتخاب است. ولی یک تفاوت اساسی دارند.

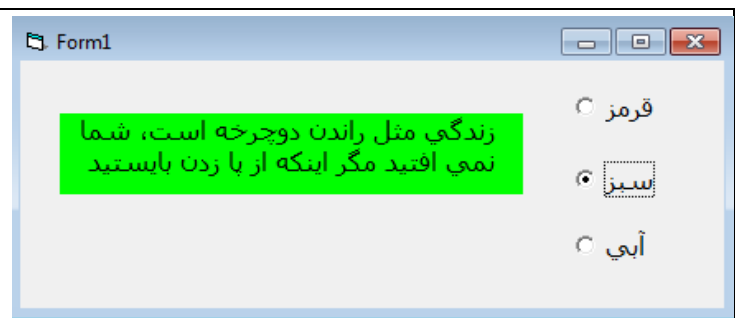
تفاوت مهم: از مجموعه چند کادر انتخاب (تیک دار کردن) میتوان هریک یا همه را انتخاب نمود ولی از بین چند دکمه رادیویی فقط و فقط حداکثر یکی (یا هیچی) قابل انتخاب است.

مثال: سه دکمه رادیویی اضافه نمایید که بتوان در اثر انتخاب آنها رنگ پس زمینه برچسب داده شده را به سه رنگ قرمز، آبی و زرد تغییر داد.

```
Private Sub Option1_Click() → قرمز
    Label1.BackColor = vbRed
End Sub
```

```
Private Sub Option2_Click() → سبز
    Label1.BackColor = vbGreen
End Sub
```

```
Private Sub Option3_Click() → آبی
    Label1.BackColor = vbBlue
End Sub
```



دستور select case

برای پرهیز از if..else های طولانی و تو در تو و جلوگیری از نا خوانا شدن برنامه از دستور ساده تری به نام select case استفاده می گردد. ساختار آن بصورت زیر است:

```
Select case عبارت
Case ۱ مقدار
    دستورات ۱
Case ۲ مقدار
    دستورات ۲
Case ۳ مقدار
    دستورات ۳
Case ۴ مقدار
    دستورات ۴
.....
Case else
    اگر مقدار عبارت با هیچکدام برابر نبود
    دستورات متفرقه
End select
```

نحوه اجرا: نخست مقدار عبارت داده شده محاسبه شده و بر اساس مقدار آن، یکی از مقادیر جلو case که با آن برابر باشد، دستورات داخل آن اجرا شده و کار آن تمام می شود. مثال: خواندن یک عدد و نمایش نام روز هفته (۰: شنبه، ۱: یکشنبه و ... ۶: جمعه)

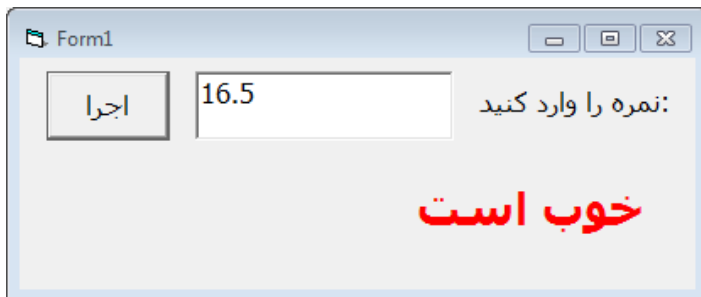
```
Dim n As Integer
n = Text1.Text
Select Case n
Case 0
Text2.Text = "شنبه"
Case 1
Text2.Text = "یکشنبه"
Case 2
Text2.Text = "دوشنبه"
Case 3
Text2.Text = "سه شنبه"
Case 4
Text2.Text = "چهارشنبه"
Case 5
Text2.Text = "پنج شنبه"
Case 6
Text2.Text = "جمعه"
Case Else
Text2.Text = "خطای ورودی"
End Select
```



حالت دیگر دستور select استفاده از قالب زیر برای مقایسه در یک محدوده است:

عبارت Select case
مقدار ۱ علامت مقایسه IS
دستورات ۱
مقدار ۲ علامت مقایسه IS
دستورات ۲
مقدار ۳ علامت مقایسه IS
دستورات ۳
مقدار ۴ علامت مقایسه IS
دستورات ۴
.....
End select

مثال: برنامه بررسی وضعیت نمره



Private Sub Command1_Click()

```
Dim nomre As Single
nomre = Val(Text1.Text)
select Case nomre
case IS > 20
Label2.Caption = "نمره اشتباه است"
case IS >= 18
Label2.Caption = "آفرین"
case IS >= 15
Label2.Caption = "خوب است"
case IS >= 12
Label2.Caption = "متوسط"
case IS >= 10
Label2.Caption = "ضعیف"
case IS < 10
Label2.Caption = "له شت داهه شئی!!"
End select
End Sub
```

برای بررسی محدوده خاص می توان از قالب زیر نیز در دستور select استفاده کرد:

عبارت Select case

مقدار دوم **TO** مقدار اول

دستورات ۱

مقدار دوم **TO** مقدار اول

دستورات ۲

مقدار دوم **TO** مقدار اول

دستورات ۳

مقدار دوم **TO** مقدار اول

دستورات ۴

.....

End select

مثال: شمارش تعداد رقم های یک عدد

Private Sub Command1_Click()

Dim n As long

n = Val(Text1.Text)

select Case n

case 0 TO 9

Label2.Caption = "یک رقمی"

case 10 To 99

Label2.Caption = "دو رقمی"

case 100 To 999

Label2.Caption = "سه رقمی"

case 1000 To 9999

Label2.Caption = "چهار رقمی"

case 10000 To 99999

Label2.Caption = "پنج رقمی"

case 100000 To 999999

Label2.Caption = "شش رقمی"

Case Else

Label2.Caption = "عدد خیلی بزرگ است"

End select

End Sub

✓ تابع IIF(): اگر بخواهیم بر اساس درست یا نادرست بودن شرطی، یکی از دو حالت را انتخاب نماییم می توانیم به جای دستور if-else از این تابع استفاده کنیم:

(مقدار در صورت نادرست بودن , مقدار در صورت درست بودن , شرط) = IIF(متغیر)

روش (۱)	روش (۲)	مثال: بدست آوردن بیشترین مقدار بین دو عدد روش (۱): به کمک ساختار if-else روش (۲): با استفاده از تابع IIF()
<pre>If (a>b) then max = a Else max= b end if</pre>	<pre>max=IIF(a>b , a , b)</pre>	

تمرین: دستور زیر چکاری انجام می دهد؟

$S = \text{IIF}(S < 0, -S, S)$

✓ تابع Choose(): حالت ساده شده select-case است. اگر بخواهیم بر اساس اولین مقدار یکی از مقادیر بعدی را انتخاب نماییم، از این تابع کمک میگیریم:

(..., مقدار انتخابی ۳, مقدار انتخابی ۲, مقدار انتخابی ۱, مقدار مقایسه) = Choose(متغیر)

نکته: مقدار مقایسه باید از عدد یک شروع شود

مثال: بر اساس عدد ورودی (از ۱ تا ۷) نام روز هفته را در یک برچسب نشان دهید.

Dim n as integer

n= Text1.text

Label1.caption = choose(n, 'شنبه', 'یکشنبه', 'دو شنبه', ...)

✓ تابع switch(): لیستی از جفت عبارتهای شرط و مقدار را بررسی می نماید. اگر یکی از عبارتها درست باشد، مقدار متناظر آن برگردانده می شود:

(..., مقدار انتخابی ۳, عبارت ۳, مقدار انتخابی ۲, عبارت ۲, مقدار انتخابی ۱, عبارت ۱) = Choose(متغیر)

مثال: بر اساس عدد ورودی (از ۱ تا ۷) نام روز هفته را در یک برچسب نشان دهید.

Dim n as integer

n= Text1.text

Label1.caption = switch(n=1 , 'شنبه' , n=2 , 'یکشنبه' , n=3 , 'دو شنبه' , ...)

آشنایی با کنترل کادر انتخاب Checkbox

برای انتخاب چندین حالت همزمان بکار می رود

مثال: حالت‌های **Bold**, *Italic*, Underline متن یک label را به کمک کنترل‌های کادر انتخاب تعیین نمایید.

```
Private Sub Check1_Click() → Bold
```

```
Label1.FontBold = Check1.Value
```

```
End Sub
```

```
Private Sub Check2_Click() → italic
```

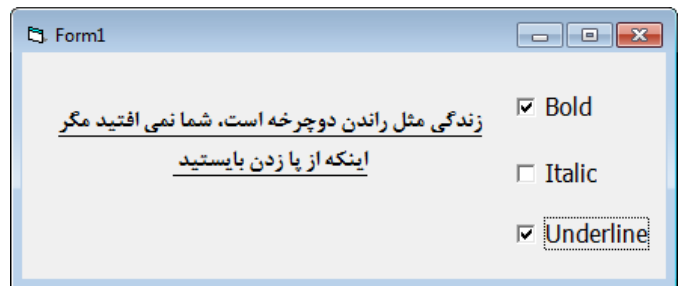
```
Label1.FontItalic = Check2.Value
```

```
End Sub
```

```
Private Sub Check3_Click() → underline
```

```
Label1.FontUnderline = Check3.Value
```

```
End Sub
```



حلقه های تکرار

۱- حلقه **For**: این حلقه برای کنترل تکرار دستورات به تعداد دفعات معین بکار می رود:

[اندازه قدم حلقه step مقدار نهایی to مقدار شروع = متغیر For دستورات متغیر Next

- مقدار شروع: مقدار اولیه متغیر
- مقدار نهایی: آخرین مقداری که متغیر می تواند به خود بگیرد
- اندازه قدم حلقه: میزان افزایش (کاهش) متغیر بعد از هر بار اجرا

توجه: اگر اندازه قدم حلقه برابر ۱ باشد نوشتن 1 step لازم نیست. مثلاً نمایش اعداد ۱ تا ۱۰۰:

```
For I = 1 to 100
    Print I;
Next I
```

نحوه اجرا:

- ۱) متغیر از مقدار اولیه شروع می کند
- ۲) اگر مقدار متغیر از مقدار نهایی بزرگتر شد (در حالت اندازه قدم مثبت) یا از مقدار نهایی کوچکتر شد (در حالت اندازه قدم منفی)، اتمام کار حلقه
- ۳) اجرای دستورات
- ۴) تغییر متغیر به اندازه قدم داده شده
- ۵) برگرد به مرحله ۲

مثال ۱: عددی بخوانید و

تمام اعداد فرد کوچکتر از آن را به دو صورت صعودی و نزولی چاپ کنید

```
Private Sub Command1_Click() → صعودی
    Dim n As Integer, I As Integer
    n = Text1.Text
    Print
    Print "صعودی: "
    For I = 1 To n Step 2
        Print I;
    Next I
End Sub
```

مثال ۲: عددی بخوانید و تمام مقسوم علیه های آن را چاپ کنید

```
Private Sub Command2_Click() → نزولی
    Dim n As Integer, I As Integer
    n = Text1.Text
    Print
    Print "نزولی: "
    For I = n To 1 Step -2
        Print I;
    Next I
End Sub
```

```
Dim n as integer, I as integer
N= text1.text
For I = 1 to n
    If (n mod I = 0) then
        Print I;
    End if
Next i
```

مثال ۳: دو عدد m,n را

خوانده و m^n (توان: m را n مرتبه در خود ضرب کنید) را به کمک ضرب حساب کرده و نمایش دهید.

```
Dim n as integer, m as integer, I as integer, P as Long
n= Text1.text
m = Text2.text
p= 1 → طرف نتیجه
For I = 1 to n → به تعداد عدد n
    P = p*m → در خود ضرب شود m
Next i
Text3.text = p
```

تمرین ۱: عدد شماره ماه را از Text1 خوانده و نام آن ماه (مثلاً عدد ۷ برابر ماه مهر است) را در Text2 بنویسید (به کمک Select).

تمرین ۲: تمام اعداد زوج کوچکتر از ۱۰۰۰ را چاپ کنید (به کمک for)

تمرین ۳: به کمک حلقه for تعداد ۱۰ عدد با دستور inputbox بخوانید و تعداد اعداد زوج و فرد آنها را در دو Text نمایش دهید. (مثلاً: 5,8,7,12,9,11 تعداد زوج = ۲ و تعداد فرد = ۴)

تمرین ۴: عدد اول - خودآزمایی شماره ۴ صفحه ۱۵۲ (عددی بخوانید و تشخیص دهید که اول است یا نه؛ عدد اول یعنی غیر از خودش و یک مقسوم علیه دیگری ندارد)

تمرین ۵: یک label روی فرم قرار دهید و به کمک سه Checkbox حالت فونت (Bold, Italic, Underline) و به کمک سه Radio Button رنگ زمینه آنرا تغییر دهید.

۲- **حلقه While** : برای تکرار دستورات به تعداد نام معلوم تا زمانیکه شرط برقرار باشد، دستورات اجرا می شوند. تفاوت حالت (۱) و (۲) در این است که در حالت دوم حداقل یکبار اجرا می گردد حتی اگر شرط برقرار نباشد.

(1) Do while (شرط) دستورات Loop
(2) Do دستورات Loop while (شرط)

مثال ۳: تعدادی نمره از ورودی بخوانید (با inputbox) و مجموع و میانگین آنها را نمایش دهید. پایان ورودی با عدد صفر است.

```
Dim n As Integer, a As Single, sum As Single
Dim s As String
sum = 0
n = 0
s = InputBox("نمره را وارد نمایید")
a = Val(s)
Do While (a <> 0)
    Print a;
    sum = sum + a
    n = n + 1
    s = InputBox("نمره بعدی را وارد نمایید")
    a = Val(s)
Loop
Text1.Text = sum
Text2.Text = sum / n
```

۳- حلقه های تکرار until

به دو صورت است:

در هر دو حالت تا زمانیکه شرط **نادرست** باشد اجرای دستورات تکرار می گردد و وقتی شرط **درست** شد از حلقه خارج می گردد (بر خلاف while) **مفهوم:** تا رسیدن به شرط مورد نظر دستورات را تکرار کن

تفاوت حالت ۱ و ۲: در حالت (۱) شرط در اول بررسی می گردد و در (۲) در آخر. پس مورد (۲) حداقل یکبار اجرا می گردد.

1) Do until شرط دستورات Loop
2) Do دستورات شرط Loop until

```
I=2
Do until (I > 5)
    Print I
    I=I+1
loop
```

I	خروجی
۲	--
۳	۲
۴	۲و۳
۵	۲و۳و۴
۶	۲و۳و۴و۵
خروج از حلقه	

مثال: خروجی زیر چیست؟
نحوه اجرا:

مثال: برنامه ای بنویسید که به کمک حلقه do until تعدادی عدد بخواند و مجموع و میانگین آنها را چاپ کند. پایان ورودی صفر است.

حل: در رویداد یک دکمه:

```

Dim s as integer → جمع
Dim a as integer → ورودی
Dim m as single → میانگین
Dim n as integer → تعداد
N=0
S=0
Do
    a = inputbox("عدد")
    S=s+a → جمع کردن
    n=n+1 → شمارش اعداد
Loop until (a=0)
m=s/n → میانگین
Print s,m → چاپ مجموع و میانگین

```

خروجی	M میانگین	N تعداد	S مجموع	A عدد
شروع		۰	۰	
		۱	۵	۵
		۲	۱۱	۶
		۳	۲۶	۱۵
		۴	۵۱	۲۵
		۵	۶۱	۱۰
	۱۰.۱۹	۶	۶۱	۰
۶۱ و ۱۰.۱۹ خروج از حلقه				

مثال: عددی بزرگ بخوانید و تعداد و مجموع رقمهای آنرا چاپ کنید.

```

Dim s as integer → جمع رقم
Dim a as long → ورودی
Dim n as integer → تعداد رقم
Dim r as integer → رقم
n=0
S=0
a= inputbox("عدد")
Do until (a=0)
    r = a mod 10 → باقیمانده (اولین رقم)
    s = s + r → جمع کردن
    n = n+1 → شمارش رقمها
    a = a\10 → رقمهای بعدی
Loop
Print s,n → چاپ مجموع و تعداد رقمها

```

خروجی	N تعداد	S مجموع	R رقم	A عدد
شروع	۰	۰		۹۸۵۴۷
	۱	۷	۷	۹۸۵۴
	۲	۱۱	۴	۹۸۵
	۳	۱۶	۵	۹۸
	۴	۲۴	۸	۹
	۵	۳۳	۹	۰
۳۳ و ۵				خروج از حلقه

حلقه های تودرتو: میتوان حلقه ها را بصورت تودرتو نوشت یعنی به ازای هربار اجرای حلقه بیرونی، حلقه داخلی کاملاً اجرا گردد.

Dim I as integer, j as integer

For i=1 to 10

For j= 1 to 10

Print i*j;

Next j

print

Next i

مثال: دستورات زیر
جدول ضرب را روی فرم
چاپ می کند:

```
Dim I as integer, j as integer
```

```
For i=1 to 10
```

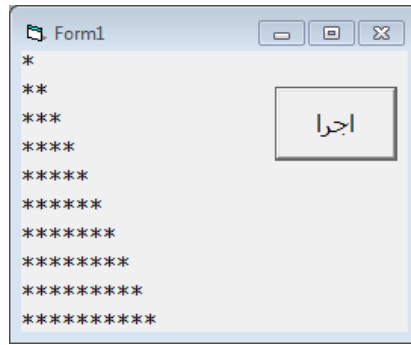
```
For j= 1 to i
```

```
Print "*";
```

```
Next j
```

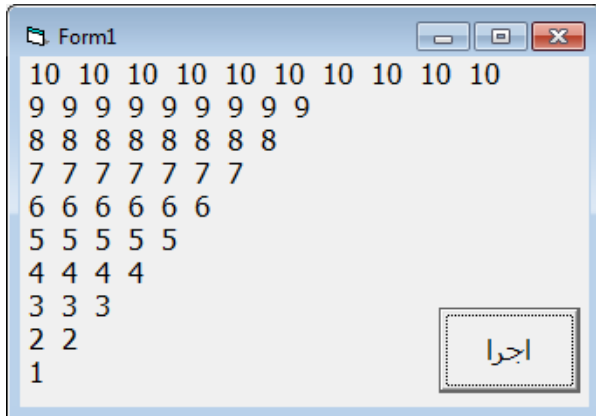
```
print
```

```
Next i
```

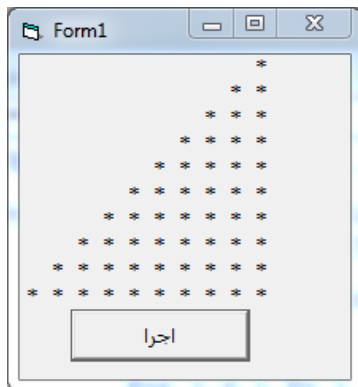


مثال 2: چاپ ستاره در چند خط
بصورت شکل زیر:

تمرین (۱): برنامه ای بنویسید که شکل زیر را
نمایش دهد:



تمرین (۲): برنامه ای بنویسید که شکل زیر را نمایش دهد:



تمرین (۳): برنامه ای بنویسید که یک عدد دودویی حداکثر ۵ رقمی را گرفته و آن را به
مبنای ۱۰ ببرد و نتیجه را در یک Message Box نشان دهد. (در صورت ورود عدد غیر مجاز
پیغام مناسبی نمایش دهد.)

آشنایی با چند تابع ریاضی و محاسباتی

- قسمت صحیح عدد را بر می گرداند و برای اعداد منفی به سمت کمتر گرد می کند: (عدد) Int
- قسمت صحیح عدد را بر می گرداند : (عدد) Fix

چند مثال:

$$\text{Int}(6.8) \rightarrow 6, \text{Int}(-6.8) \rightarrow -7 \quad | \quad \text{Fix}(6.8) \rightarrow 6, \text{Fix}(-6.8) \rightarrow -6$$

- $\text{Abs}(\text{عدد}) \rightarrow$ قدر مطلق عدد (مثبت) | $\text{Abs}(-8) \rightarrow 8, \text{Abs}(5) \rightarrow 5$
- $\text{Sqr}(\text{عدد}) \rightarrow$ جذریک عدد | $\text{Sqr}(64) \rightarrow 8$
- $\text{Cos}(\text{زاویه برحسب رادیان}) \rightarrow$ کسینوس زاویه
- $\text{Sin}(\text{زاویه برحسب رادیان}) \rightarrow$ سینوس زاویه
- $\text{Tan}(\text{زاویه برحسب رادیان}) \rightarrow$ تانژانت زاویه

توجه: برای بدست آوردن زاویه بر حسب رادیان از رابطه زیر استفاده کنید:

$$\pi = 3.141593 \quad \text{و} \quad \pi \text{ (مقدار } \pi \text{ زاویه برحسب درجه)} = \text{زاویه برحسب رادیان} / 180$$

مثلاً اگر زاویه ۴۵ درجه باشد، مقدار رادیان برابر است با:

$$\text{Cos}(0.78539825) = 0.7071, \text{Cos}(0.78539825) = (45 * 3.141593) / 180 = 0.78539825 \text{ رادیان}$$

- $\text{Sgn}(\text{عدد}) \rightarrow$ تعیین علامت عدد (مثبت: ۱، منفی: -۱ و صفر: ۰) | $\text{Sgn}(8) \rightarrow 1, \text{Sgn}(-25) \rightarrow -1$
- $\text{Log}(\text{عدد}) \rightarrow$ لگاریتم طبیعی
- $\text{Rnd}() \rightarrow$ تولید عددی تصادفی بین صفر و یک

توجه: برای تولید عدد تصادفی بین a,b از دستور زیر استفاده کنید:

$$a+b*\text{Rnd}()$$

$$1+6*\text{Rnd}() \rightarrow \text{عدد تصادفی بین ۱ تا ۷}$$

مثال (۱): سینوس زاویه های ۰ تا ۳۶۰ درجه را با فاصله ۱۵ درجه روی فرم نمایش دهید.

```
Private Sub Command1_Click()
    Dim z As Integer, rad As Single
    Print " z          sin(z)"
    Print "===== "
    For z = 0 To 360 Step 15
        rad = (z * 3.141593) / 180
        Print z, Sin(rad)
    Next z
End Sub
```

z	sin(z)
0	0
15	0.258819080926347
30	0.500000064237527
45	0.707106838787704
60	0.866025477959537
75	0.965925851143665
90	0.999999999999987

مثال (۲): با در دست داشتن طول سایه (فاصله) یک درخت و زاویه دید نسبت به نوک درخت، ارتفاع آنرا حساب کنید.

```

Private Sub Command1_Click()
    Dim z As Integer, rad As Single
    Dim dist As Single → فاصله
    Dim height As Single → ارتفاع
    dist = Text1.Text
    z = Text2.Text
    rad = (z * 3.141593) / 180 → رادیان
    height = dist * Tan(rad)
    Text3.Text = height
End Sub

```

رشته ها (String): مجموعه ای از هر کارکتر دلخواه شامل حروف، علائم و ارقام و...

متغیر رشته ای بصورت زیر تعریف می گردد: `dim s as string`
و مقدار رشته ای در علامت " " قرار می گیرد مانند: `s="salam"`

اعمال رشته ها

۱. **الحاق** یا به چسباندن دو رشته توسط علامت + یا & استفاده از علامت & بهتر است.
مثال:

```

Dim s1 as string, s2 as string
S1= "salam"
S2="dostan"
S1= s1 & s2 → s1="salamdostan"

```

۲. **مقایسه رشته ها:** ترتیب الفبایی حروف و کارکترها مد نظر است. برای مقایسه از دستور زیر استفاده می شود:

(نحوه مقایسه، رشته دوم، رشته اول) `strComp` = متغیر عددی

نتیجه این تابع عددی است که مقادیر آن بصورت زیر است:

- ۱ : رشته دوم < رشته اول ؛ از نظر الفبایی
- -۱ : رشته دوم > رشته اول ؛ از نظر الفبایی
- ۰ : رشته دوم = رشته اول ؛ از نظر الفبایی

نحوه مقایسه به دو حالت می باشد:

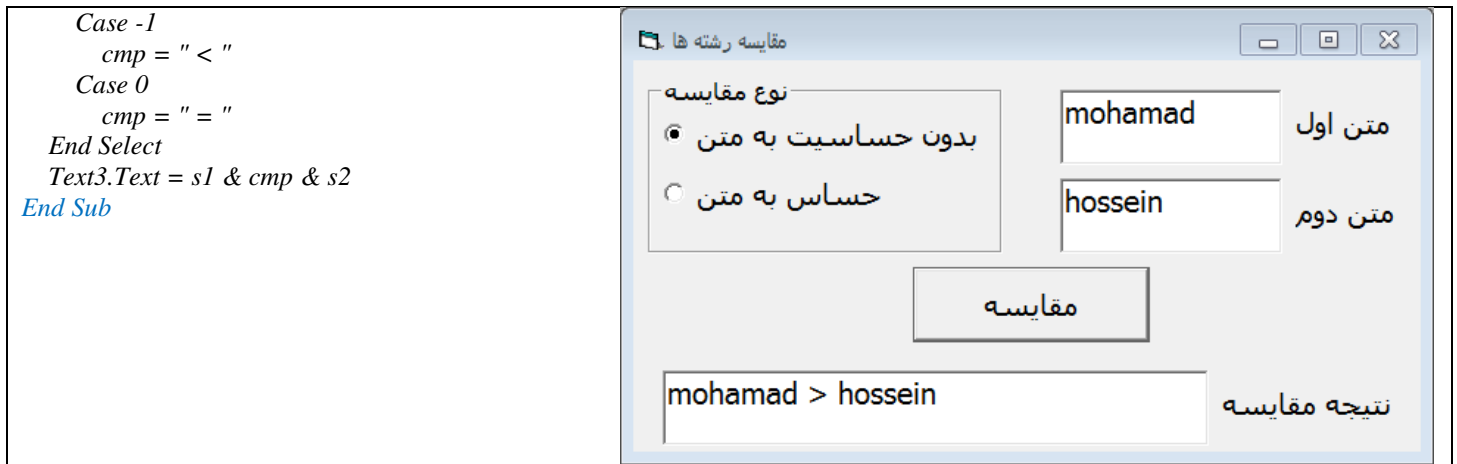
- `VbBinaryCompare`: مقایسه **با لحاظ کردن** حروف کوچک و بزرگ انگلیسی
- `VbTextCompare`: مقایسه **بدون لحاظ کردن** حروف کوچک و بزرگ انگلیسی

مثال: دو رشته را با Text از ورودی بگیرید و در دو حالت مقایسه کنید.

```

Private Sub Command1_Click()
    Dim s1 As String, s2 As String, cmp As String
    Dim r As Integer
    s1 = Text1.Text
    s2 = Text2.Text
    If Option1.Value Then
        r = StrComp(s1, s2, vbTextCompare)
    Else
        r = StrComp(s1, s2, vbBinaryCompare)
    End If
    Select Case r
        Case 1
            cmp = ">"

```



مقایسه با دستور LIKE: این دستور برای بدست آوردن شباهت دو رشته بکار می رود. نحوه کار آن بصورت زیر است:

رشته دوم **Like** رشته اول = متغیر منطقی

جواب آن True: مشابه هم یا False: مشابه هم نیستند

مثال ۱: آیا رشته ورودی با حرف m شروع می شود؟

علامت * یعنی هر تعداد کارکتر دیگری → s1 Like "m*"

مثال ۲: آیا رشته ورودی با حرف m پایان می یابد؟

s1 Like "*m"

مثال ۳: آیا رشته ورودی شامل حرف m است؟

s1 Like "*m*"

علامت مورد استفاده در دستور Like

- علامت * یعنی هر تعداد کارکتر
- علامت ؟ یعنی فقط یک کارکتر
- علامت # یعنی فقط یک رقم عددی
- علامت [] :حروف محدوده خاص مثلاً a تا f بصورت [a-f] نوشته می شود

مثال ۴: آیا رشته ورودی با حرف m شروع و با فقط یک حرف تمام می شود؟

s1 Like "m?"

مثال ۵: آیا رشته ورودی با حرف m شروع و با فقط یک رقم تمام می شود؟

s1 Like "m#"

مثال ۶: آیا رشته ورودی با حرف m شروع و با حروف a تا f تمام می شود؟

s1 Like "m[a-f]" → a,b,c,d,e,f

توابع رشته ای

۱- تابع Len: برای بدست آوردن طول رشته (چند کارکتر؟)

Len ("salam") → 5

Len ("khobi? Chetori?") → 15

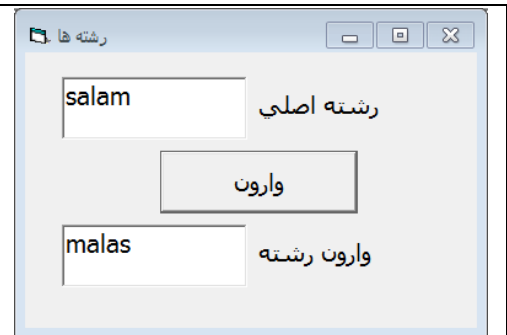
۲- تابع Mid: استخراج حروف و کارکترها از رشته به تعداد دلخواه

(چند تا کارکتر, شروع, رشته) Mid

از دومین حرف به بعد چهار تا بردار \rightarrow "hobi" \rightarrow Mid ("khobi?", 2,4)

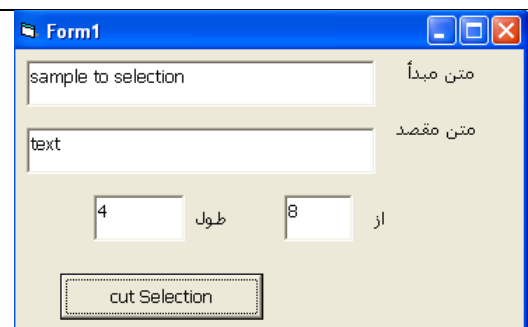
مثال: برنامه ای طراحی کنید که به کمک دو تابع فوق، رشته ورودی را وارون کند. مثلاً salam به malas تبدیل شود.
راه حل: از آخر رشته حرف به حرف برداشته و می نویسیم.

```
Private Sub Command1_Click()  
    Dim s1 As String, pos As Integer  
    s1 = Text1.Text  
    Text2.Text = ""  
    For pos = Len(s1) To 1 Step -1  
        Text2.Text = Text2.Text & Mid(s1, pos, 1)  
    Next pos  
End Sub
```



تمرین : پروژه ای طراحی کنید که با تعیین محدوده شروع و پایان، رشته محدوده را از Text1 برداشته و در Text2 کپی کند. مثال: sample text to selection

```
Private Sub cmCutSel_Click()  
    Dim start As Integer, length As Integer  
    start = Val(Text3.Text)  
    length = Val(Text4.Text)  
    Text2.Text = Mid(Text1.Text, start, length)  
    Text1.Text = Mid(Text1.Text, 1, start - 1) & Mid(Text1.Text, length + start + 1, Len(Text1.Text))  
End Sub
```



نکته: می توان به کمک mid رشته ای را با مقادیر رشته دیگر جایگزین کرد:

رشته جدید = (تعداد , شروع, رشته) Mid

$s = \text{"salam"}$

$\text{Mid}(s, 2, 3) = \text{"ete"} \rightarrow s = \text{"setem"}$

۳- تابع Left: از سمت چپ رشته به تعداد دلخواه بر می دارد:

(تعداد از چپ, رشته) Left

$\text{Left}(\text{"salam"}, 3) \rightarrow \text{"sal"}$

۴- تابع Right: از سمت راست رشته به تعداد دلخواه بر می دارد:

(تعداد از راست, رشته) Right

$\text{Right}(\text{"salam"}, 3) \rightarrow \text{"lam"}$

۵- تابع InStr: برای پیدا کردن رشته ای در رشته دیگر

(رشته فرعی, رشته اصلی, مکان شروع جستجو) InStr

$\text{InStr}(1, \text{"salam"}, \text{"lam"}) \rightarrow$ جواب محل پیدا شدن یعنی ۳ است.

$\text{InStr}(1, \text{"salam"}, \text{"tam"}) \rightarrow$ پیدا نشد و جواب ۰ است

نکته : مکان شروع اگر از اول رشته باشد نیازی به نوشتن آن نیست. و خودش یک در نظر می گیرد

InStr("salam", "lam") → جواب ۳ است

6-تابع InStrRev: برای پیدا کردن رشته ای در رشته دیگر از آخر
(مکان شروع جستجو, رشته فرعی, رشته اصلی)

InStrRev("salam", "lam", len("salam")) →

جواب محل پیدا شدن یعنی ۳ است.

مثال: در یک برنامه کاربرد توابع Instr, instrRev را نشان دهید.

Private Sub cmCutSel_Click()

Dim s1 As String, r As Integer

s1 = Text1.Text

s2 = Text2.Text

If Option1.Value Then

r = InStr(s1, s2)

Else

r = InStrRev(s1, s2)

End If

If r <> 0 Then

Text3.Text = s2 & " وجود دارد در "

Else

Text3.Text = s2 & "' وجود ندارد در '"

End If

End Sub

تابع های حذف فواصل خالی

- LTrim: حذف فاصله چپ
- RTrim: حذف فاصله راست
- Trim: حذف فاصله چپ و راست

تابع Space: تابعی است برای ایجاد فواصل خالی به تعداد دلخواه
مثال: بین دو کلمه this و sample ۱۰ فاصله خالی ایجاد کنید:
جواب:

Print "this" & space(10) & "sample"

تابع String: ایجاد رشته از یک کارکتر خاص به طول دلخواه
مثال: رشته ای از حرف A به طول ۵ ایجاد کنید:

Print string("A",5) → AAAAA

تابع Replace: جایگزینی بخشی از یک رشته با رشته دیگر
(رشته جایگزین, رشته های که باید عوض شود, رشته اصلی)

مثال: در رشته نام خود هرچه حرف m وجود دارد با حرف w عوض کنید

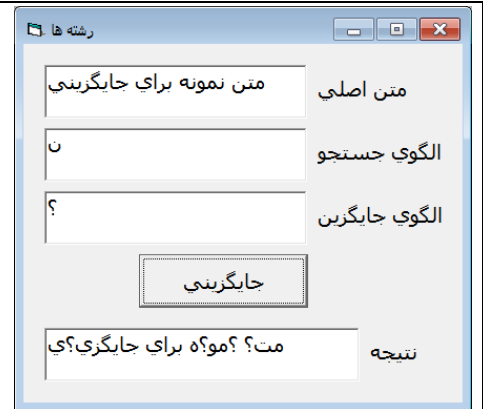
Replace("mohamad", "m", "w") → "wohawad"

مثال: برنامه ای بنویسید که رشته ای را خوانده و بخشی از آن را با رشته ای جدید عوض کند:

```
Private Sub cmCutSel_Click()
```

```
Text4.Text = Replace(Text1.Text, Text2.Text, Text3.Text)
```

```
End Sub
```



توابع تبدیلی

تابع Oct: تبدیل عدد به مبنای ۸

تابع Hex: تبدیل عدد به مبنای ۱۶

مثال:

Oct(459) → "713"

Hex(459) → "1CB"

تابع Asc: تبدیل کارکتر به کد اسکی عددی

تابع Chr: تبدیل کد عددی اسکی به کارکتر

مثال:

Asc("B") → 66

Chr(76) → "L" , Chr(66) → "B"

تابع Str و CStr: تبدیل عدد به رشته

تابع Val: تبدیل رشته به عدد

مثال:

Str(1234) → "1234"

Val("456") → 456

تابع LCase: تبدیل رشته به حروف کوچک

تابع UCase: تبدیل رشته به حروف بزرگ

مثال:

LCase("aLi mohAMad") → "ali mohamad"

UCase("aLi mohAMad") → "ALI MOHAMAD"

تابع strReverse: این تابع معکوس یک رشته بدست می آورد

مثال:

StrReverse("Computer") → "retupmoC"

مثال: خودآزمایی شماره ۵ آخر فصل صفحه ۱۸۳

تمرین منزل (مهم): خودآزمایی های شماره ۳ و ۴ و ۷ و ۸