

<b>Microsoft Visual Basic Nedir?</b>	<b>Componentler (Kontrol Elemanları)</b>
<b>DEĞİŞKENLER</b>	Textbox (Metin Kutusu)
<b>MsgBox (Mesaj Kutusu)</b>	Label (Etiket)
<b>InputBox (Bilgi Giriş Kutusu)</b>	Command Button (Komut Düğmesi)
<b>Döngüler</b>	CheckBox (İşaret Kutusu)
If ..... Then ..... Else ..... End if	Option Button (Seçenek Düğmesi)
Select Case	Horizontal & Vertical ScrollBar (Kaydırma Çubuğu)
For ..... Next	Shape (Şekil KontrolElemanı)
<b>QBColor ()</b>	Timer Control (Zamanlayıcı)
<b>RGB ()</b>	Frame Control (Çerçeve)
<b>Like</b>	UpDown(Yukarı aşağı arttırma)
<b>Disk ve Dosya Komutları</b>	Line (Çizgi)
CurDir	ComboBox (Açılan Liste)
ChDir	ListBox (Listeleme Kutusu)
ChDrive	DriveListBox (Sürücü Listeleme Kutusu)
MkDir	Directory List Box (Dizin Listeleme Kutusu)
Rmdir	FileList Box (Dosya Listeleme Kutusu)
FileCopy	PictureBox (Resim Kutusu)
FileLen	Image (Resim Gösterme Kontrolü)
FileDateTime	Common Dialog (Diyalog Pencereleeri)
GetAttr	MSMasked (Formatlı Giriş)
SetAttr	SSTab
Name	Richtextbox
Dir	Popup Menü
Kill	MsFlexGrid
End	ProgressBar (İlerleme Çubuğu)
<b>Çizim Yapmak</b>	Slider
Line	DateTimePicker
Circle	MonthView
Pset	Toolbar
<b>Random Dosyalar</b>	Coolbar
<b>Fonksiyonlar</b>	Statusbar
Abs Fonksiyonu	<b>Veri Tabanı Uygulamaları</b>
Asc Fonksiyonu	DBGrid
Chr Fonksiyonu	ADO ile Veri Tabanı
Date Fonksiyonu	DataGrid
DateDiff Fonksiyonu	Data Report
Int Fonksiyonu	<b>SQL Komutları</b>
LCase Fonksiyonu	<b>Formlar</b>
Left Fonksiyonu	<b>MDIFormlar</b>
Len Fonksiyonu	
Ltrim, Rtrim ve Trim Fonksiyonu	
Mid Fonksiyonu	
Now Fonksiyonu	
Round Fonksiyonu	
Space Fonksiyonu	
Shell Fonksiyonu	
Sqr Fonksiyonu	
String Fonksiyonu	
Time Fonksiyonu	
UCase Fonksiyonu	
Val Fonksiyonu	
<b>Dizi Değişkenler</b>	
<b>Dinamik Diziler</b>	
<b>Function Oluşturma</b>	

## Microsoft Visual Basic Nedir?

Visual Basic bir programlama dilidir. Visual Basic'in bir programın yazılıp çalıştırılmasından öte, program geliştirmek için çok sayıda araca sahip olması; kullanıcı arabiriminin tasarlanması, hataların giderilmesi, veritabanı yaratmak, farklı türde programlar üretmek gibi gelişmiş özellikleri, onu bir "programlama ortamı" ya da "program geliştirme ortamı" olarak tanımlamamıza neden olur.

Visual Basic, çok amaçlı ve aynı zamanda kolay bir şekilde program geliştirmenizi sağlayan bir programlama ortamıdır. Visual olmak (görsellik), simgelerle yapılan iletişimi ifade eder. Visual Basic programlarında ortamın tasarımı öncelikli olarak yapılan işlemdir. Kod yazmak ikincil kalır.

Visual Basic ile küçük işlemler için bir program geliştirilebileceği gibi, bütün dünyaya uzanabilecek bir internet uygulaması yada büyük firma düzeyinde kullanılacak gelişmiş bir enterprise (kurumsal) uygulaması geliştirilebilir. Örneğin bir cari hesap, stok, satış, imalat vb. uygulamaları, oyun programı vb.

Visual Basic'in sahip olduğu araçlar:

1. Kullanıcı ara birimi geliştirmek için çok sayıda kontrol,
2. Programı derleme ve hata bulmaya yönelik çok sayıda araç,
3. Değişik veritabanlarına erişim,
4. SQL Server ve diğer enterprise düzeyindeki veritabanlarına erişim,
5. Diğer uygulamaların fonksiyonlarını kullanmayı sağlayan ActiveX teknolojisi,
6. Uygulamanın EXE dosya olarak üretilebilmesi ve dağıtılması,
7. Raporlama araçları,
8. Veri yapısı araçları,
9. Web uygulaması geliştirme araçları,
10. Çok katmanlı programlama ve Microsoft Transaction Server.

## Visual Basic'in Özellikleri

Özellikle Windows ortamında programlama geliştirme aracı olarak kullanılan Visual Basic, diğer klasik programlama dillerine göre farklı özelliklere sahiptir. Visual Basic ile program geliştirmek, öncelikle Windows arabiriminin kullanıcı etkileşiminde kullanılan araçların kullanılması anlamına gelir. Klasik programlama dilleri ise genellikle karakter tabanlı (DOS ve UNIX) işletim sistemlerinde kullanılan özellikle klavye ağırlıklı kullanıcı arabirimine sahiptir. Gelişmiş programlama ortamı olan Visual Basic, Windows uygulaması gibi mönülere, düğmelere sahiptir.

Visual Basic sadece standart kontroller sağlamaz. Aynı zamanda çok sayıda üçüncü parti yazılımcıların da hazırladığı kontrolleri de kullanabilir. Bu kontroller satın alınarak kolayca programlara eklenebilirler. Visual Basic, birçok OCX(OLE Control Extensions) kontrol ve ActiveX kontrolleri destekler.

## Bir Visual Basic Programının Kısımları

Bir program belli kısımlardan oluşur:

- Kullanıcı arabirimi
- Bilgi işleme (hesaplamalar)
- Bilgi saklama (veritabanı işlemleri)
- Raporlama işlemleri

Kullanıcı arabirimi bir programın ekranını oluşturur. Diğer bir deyişle kullanıcıların bilgi girdiği etkileşimli ekranlardır. İyi bir kullanıcı arabirim geliştirmenin yöntemleri şunlardır.:

- Anlaşılabilir, kolay ve çok yönlü kullanılabilir olması
- Veriyi net bir biçimde ifade etmeleri
- Komutları açıkça sergilemeleri,
- Mönüler, komutlar ve düğmeleriyle programın diğer bölümleriyle uyum içinde olmaları,
- Mönü, araç çubukları, fare ve klavye aracılığıyla kullanılabilir olması.

## İyi Bir Programın Nitelikleri

Programlar belli bir işin bilgisayar tarafından yerine getirilmesini sağlarken; programların belli niteliklere sahip olması gerekir. İyi program niteliklerine sahip olmak, aynı zamanda program geliştirme sürecinin de amaçlarındandır. İyi bir programın nitelikleri:

- Görsellik
- Kolaylık
- Doğruluk
- Hızlı
- İyi bir belgeleme
- Kolayca değiştirilebilme, güncellenebilme
- Etkin bir kodlama
- Etkin bir işletim (daha az sistem kaynağına gereksinim duyması)

Bunun dışında programın analiz çalışmalarının ve kodlarının iyi bir şekilde belgelenmesi gerekir (söz uçar, yazı kalır). Belgeleme belli bir sistem içinde bir kitapçık olarak hazırlanmalı ve kod içinde de açıklamalara çok önem verilmelidir.

## Çalışma Ortamı

Visual Basic 6.0 ortamı özellikle program birimlerinin düzenlenmesi için çok sayıda bölüme sahiptir. Bu bölümler aracılığıyla programın tasarımı yapılır ve kod yazılır. Çalışma ortamının hemen her alanında farenin sağ tuşuna basılarak içerik mönüleri elde edilir. Bu mönüler bulunulan yere göre içeriği değişebilen özel mönülerdir. Bütün olanaklar grafik yardımı ile programın arabiriminin geliştirilmesini, çalıştırılmasını ve hata düzeltmelerinin yapılmasını sağlar.

Visual Basic programı bir proje olarak geliştirilir. Bu proje içinde formlar, kontroller, modüller ve komponent' ler bulunur. Formlar üzerine kontroller yerleştirilir. Ardından kontrollerle ilişkili olarak olay yordamları yazılır. İstenirse genel yordamların yer aldığı modüller yazılır ve proje çalıştırılır.

## Project ve Formlar

Yeni bir Visual Basic programına bir proje olarak başlanır. Varsayılan proje türü standart EXE' dir. Proje penceresiyle birlikte bir form ekrana gelir. Projenin başlangıç adı Project1, formun başlangıç adı ise Form1'dir. Her Visual Basic uygulamasında bir form bulunur. Form üzerine denetimler ve grafikler eklenerek uygulama geliştirilir. Her bir form aynı zamanda form tasarımının yapıldığı bir alandır. Formlar uygulamanın kullanıcı arabirimini oluşturur. Visual Basic kontrollerinden istenilen kontroller form üzerine fare ile yerleştirilerek uygulamanın arabirimi oluşturulur. Bu arabirim üzerinde metin kutuları (textbox), komut düğmeleri(command buttons) gibi veri işlem elemanları yer alır. Ardından istenilen olaylara göre kodlar yazılarak programın yazılması sağlanır.

## Araç Kutusu (Toolbox)

Görsel programlama ortamlarında görsel kontrollerin çizilmesi için zengin bir araç kutusu kullanılır. Visual Basic araç kutusunda programın tasarımı sırasında kullanılmak üzere çok sayıda aracı içerir. Bu araçlar kontrol (control) olarak adlandırılır. Standart olarak Visual Basic açıldığında ekrana gelen kontroller aşağıda gösterilmiştir. Ayrıca diğer kontrollerden de eklemek isterseniz; toolbox üzerinde farenin sağ tuşuna basarak Component mönüsünden seçebilirsiniz.

### Standart Kontrol İşlevi

<b>PictureBox</b> .....	<b>Resim kutusu.</b> Bitmap, Icon, GIF resimlerin gösterilmesini sağlar
<b>Label</b> .....	<b>Etiket.</b> Sadece bilgi vermek yada görüntü olarak form üzerinde durur. Genellikle metin kutularının adlarını göstermek için kullanılır.
<b>TextBox</b> .....	<b>Metin Kutusu.</b> Veri girişi için kullanılan bir kontroldür. Hemen hemen bütün programlarda kullanıcılardan veri girişi, değiştirme yapması gereken alanlar gerekir. İşte metin kutusu bu amaçla kullanılır.
<b>Frame</b> .....	<b>Çerçeve.</b> Kontrolleri gruplamak için kullanılır.
<b>CommandButton</b> .....	<b>Komut düğmesi.</b> Bir işlemin başlatılmasını sağlar. Genellikle fare üzerine tıklanarak kullanılır.
<b>CheckBox</b> .....	<b>İşaret Kutusu.</b> Bir seçim yapmayı sağlar.
<b>OptionBox</b> .....	<b>Seçenek Kutusu.</b> Seçim yapmayı sağlar.
<b>ComboBox</b> .....	<b>Açılan Kutu.</b>
<b>ListBox</b> .....	<b>Liste.</b> Birçok elemanı liste biçiminde göstermek için kullanılır.

<b>HscrollBar</b> .....	<b>Yatay kaydırma çubuğu.</b> Yatay olarak kullanılan kaydırma çubuğu
<b>VscrollBar</b> .....	<b>Dikey kaydırma çubuğu.</b> Dikey olarak kullanılan kaydırma çubuğu
<b>Timer</b> .....	<b>Zaman saati.</b> Zamanın takip edilmesini sağlar.
<b>DriveListBox</b> .....	<b>Sürücü listesi.</b> Sistemdeki sürücüleri listeler.
<b>DirListBox</b> .....	<b>Dizin listesi.</b> Sistemdeki dizinleri (klasörleri) listeler.
<b>FileListBox</b> .....	<b>Dosya listesi.</b> Bir dizin içindeki dosyaları listeler.
<b>Shape</b> .....	<b>Şekil.</b> Şekil çizmeyi sağlar.
<b>Line</b> .....	<b>Çizgi.</b> Çizgi çizmek için kullanılır.
<b>Image</b> .....	<b>Resim gösterme.</b> Sadece resimleri göstermek için kullanılır.
<b>Data</b> .....	<b>Data.</b> Verileri göstermek için kullanılır.
<b>OLE</b> .....	<b>OLE</b>

## MÖNÜLER

### File Mönüsü

<b>New Project</b> .....	Yeni bir projeye başlamayı sağlar.
<b>Open Project</b> .....	Daha önce kaydedilmiş bir projeyi açar.
<b>Add Project</b> .....	Mevcut projeye yeni projeler ekler.
<b>Remove Project</b> .....	Projeyi ve elemanlarını ortadan kaldırır.
<b>Save Project Group</b> .....	Projeleri grup kayıt eder.
<b>Save Project Group As</b> .....	Projelere yeni ad vererek kaydeder.
<b>Save Form1</b> .....	Aktif durumdaki formu kaydeder.
<b>Save Form1 As</b> .....	Aktif durumdaki formu değişik adla kaydeder.
<b>Save Selection</b> .....	Seçili kaydet.
<b>Save Chance Script</b> .....	Transact-SQL deyimlerinin kayıt edilmesini sağlar.
<b>Print</b> .....	Form görüntülerinin veya kod sayfalarının yazdırılmasını sağlar.
<b>Print Setup</b> .....	Yazıcı ayarları.
<b>Make Project1.exe</b> .....	Projenin EXE dosyası haline getirilmesini sağlar.
<b>Make Project Group</b> .....	Projeleri bir grup olarak birleştirir.
<b>Exit</b> .....	Visual Basic'ten çıkış.

### Edit Mönüsü

<b>Undo</b> .....	Son yapılan işlemi geri alır.
<b>Redo</b> .....	Geri alınan son işlemi tekrarlar.
<b>Cut</b> .....	Seçili olanları keserek panoya atar.
<b>Copy</b> .....	Seçili olanları kopyalayarak panoya atar.
<b>Paste</b> .....	Daha önce kesilmiş yada kopyalanmış olanları yapıştırır.
<b>Paste Link</b> .....	Daha önce kopyalanmış olanları bağlantılarını oluşturur.
<b>Remove</b> .....	Seçilen elemanları siler.
<b>Delete</b> .....	Seçilen elemanları siler.
<b>Delete table from Database</b> ....	Tabloyu veritabanından siler.
<b>Select All</b> .....	Form üzerindeki yada kod penceresindeki bütün elemanlar seçilir.
<b>Select All Columns</b> .....	Bütün kolonları seçer.
<b>Table</b> .....	Tablo komutlarını gösteren bir alt mönü.
<b>Find</b> .....	Kod penceresinde istenilen bilgiyi bulur.
<b>Find Next</b> .....	Bir sonrakini bulur.
<b>Replace</b> .....	Bulunan bir sözcüğü değiştirmeyi sağlar.
<b>Indent</b> .....	Satırı içerden başlatır.
<b>Outdent</b> .....	Dışarı alır.
<b>Insert File</b> .....	Koda bir dosyadan ekleme sağlar.
<b>List properties / Methods</b> .....	Yazılan nesnelerin özelliklerinin açılan bir liste olarak ekrana getirilmesini sağlar.
<b>List Constants</b> .....	Bir özelliğin alacağı değerleri listeler.
<b>Quick Info</b> .....	Bir deyim kullanım biçimini gösterir.
<b>Parameter Info</b> .....	Bir deyim parametrelerini gösterir.

## View Mönüsü

<b>Code</b> .....	Kod penceresini görüntüler.
<b>Object</b> .....	Formu görüntüler.
<b>Definition</b> .....	Kod.
<b>Last Position</b> .....	Bir önceki konuma gidilmesini sağlar.
<b>Object Browser</b> .....	Mevcut proje yada Visual Basic nesnelere seçilmesini sağlayan görsel yardımcı.
<b>Immediate Window</b> .....	Programın kesilmesi yada hata bulunduğu (Debug durumunda) istenilen değişkenlerin değerlerinin izlenebildiği pencere.
<b>Locals Window</b> .....	Yine debug durumunda yerel değişkenlerin değerlerinin izlenmesini sağlar
<b>Watch Window</b> .....	Bir ifadenin durumunun izlenmesini sağlar.
<b>Call Stacks</b> .....	Çağırılmış ama tamamlanmamış procedure'leri gösterir.
<b>Project Explorer</b> .....	Proje içinde yer alan elemanlar görüntülenir.
<b>Properties Window</b> .....	Özellikler penceresi görüntülenir.
<b>Form Layout Window</b> .....	Formun ekrandaki görüntüsünü gösterir.
<b>Property Pages</b> .....	Özellikler tablosunu gösterir.
<b>Table</b> .....	Veritabanı, query gibi ortamlarda seçilen penceresinin açılıp kapanmasını sağlar.
<b>Zoom</b> .....	Büyütme.
<b>Show Panes</b> .....	Bölmeleri göster.
<b>Toolbox</b> .....	Araç kutusunun görüntülenmesi / görüntülenmemesi sağlanır.
<b>Data View Window</b> .....	
<b>Color Palette</b> .....	Form üzerinde renklendirme yapmak için kullanılan renk paleti ekrana gelir.
<b>Toolbars</b> .....	Araç çubuklarının görüntülenmesi için kullanılır.
<b>Visual Component Manager</b> ..	

## Project Mönüsü

<b>Add Form</b> .....	Projeye yeni bir form ekler.
<b>Add MDI Form</b> .....	Projeye yeni bir MDI form ekler.
<b>Add Module</b> .....	Projeye yeni bir modül ekler.
<b>Add Class Module</b> .....	Projeye yeni bir modül ekler.
<b>Add User Control</b> .....	Projeye yeni bir kullanıcı kontrolü ekler.
<b>Add Property Page</b> .....	Projeye yeni bir property ekler.
<b>Add User Document</b> .....	Projeye yeni bir user document ekler.
<b>Add ActiveX Designer</b> .....	Projeye yeni bir ActiveX Designer ekler.
<b>Add File</b> .....	Projeye dosya ekler.
<b>Remove &lt;form&gt;</b> .....	İstenilen eleman kaldırılır.
<b>References</b> .....	Çeşitli kütüphanelerin yüklenmesi sağlanır.
<b>Components</b> .....	Araç kutusuna eklenecek bileşenleri gösterir.
<b>&lt;Proje&gt; Properties</b> .....	Projenin özellikleri düzenlenir.

## Format Mönüsü

<b>Align</b> .....	Form üzerindeki birden çok elemanın birbirine hizalanmasını sağlar.
<b>Make Same Size</b> .....	Form üzerindeki birden çok elemanın birbirine göre aynı boyuta getirilmesini sağlar.
<b>Size to Grid</b> .....	Form üzerindeki elemanların form üzerindeki ızgaraya göre boyutlarını ayarlar.
<b>Horizontal Spacing</b> .....	Form üzerinde birden çok eleman arasındaki yatay boşluğu ayarlamak için kullanılır.
<b>Vertical Spacing</b> .....	Form üzerinde birden çok eleman arasındaki dikey boşluğu ayarlamak için kullanılır.
<b>Center in Form</b> .....	Form üzerindeki elemanların forma dikey yada yatay olarak ortalanmasını sağlar.
<b>Order</b> .....	Form üzerinde üst üste duran elemanların sırasını belirler.
<b>Lock Controls</b> .....	Form üzerindeki elemanların kilitlenerek yerlerinin oynamasını sağlar.

## Debug Mönüsü

<b>Step Into</b> .....	Programın satır satır çalışmasını sağlar. F8 tuşu.
<b>Step Over</b> .....	Programın procedure olarak çalışmasını sağlar.
<b>Step Out</b> .....	Programın
<b>Run to Cursor</b> .....	Programı bulunulan satıra kadar çalıştırır.
<b>Add Watch</b> .....	Program çalışırken değişkenlerin değerlerini kayıt eder ve izlenmesini sağlar.
<b>Edit Watch</b> .....	İstenilen izleme satırını düzenler.
<b>Quick Watch</b> .....	Değişkenlerin seçildikleri andaki değerleri görüntülenir.
<b>Toggle Breakpoint</b> .....	Bir satıra kesme noktası işareti vererek programın bu satıra durması sağlanır.
<b>Clear All Breakpoints</b> .....	Bütün kesme noktalarını kaldırır.
<b>Set Next Statements</b> .....	jkh
<b>Show Next Statements</b> .....	

## Run Mönüsü

<b>Start</b> .....	Programı çalıştırır. F5 tuşu.
<b>Start With Full Compile</b> .....	Programı derler / çalıştırır.
<b>Break</b> .....	Programı keser.
<b>End</b> .....	Programı sonlandırır.
<b>Restart</b> .....	Programı yeniden başlatır.

## Query Mönüsü

<b>Run</b> .....	Sorguyu çalıştırır.
<b>Clear Results</b> .....	Sorguyu temizler.
<b>Verify SQL Syntax</b> .....	SQL deyimini denetler.
<b>Group By</b> .....	Guruplar.
<b>Change Type</b> .....	Sorgu tipini değiştirir.
<b>Add to Output</b> .....	Seçilen kolonu giriş bilgisine ekler.
<b>Sort Ascending</b> .....	Artan sırada sıralar.
<b>Sort Descending</b> .....	Azalan sırada sıralar.
<b>Remove Filter</b> .....	Filtreyi kaldırır.
<b>Select All Rows From ...</b> .....	Bütün satırları seçer.

## Diagram Mönüsü

<b>New Text Annotation</b> .....	Metin açıklama ekler.
<b>Set Text Font</b> .....	Metin yazı tiplerini düzenler.
<b>Add related Tables</b> .....	İlişkili tablolar ekler.
<b>Show Relationship Labels</b> ....	İlişki etiketlerini ekler.
<b>Modify Custom View</b> .....	Özel görünümü değiştirir.
<b>View Pages Breaks</b> .....	Sayfa aralarını (sonlarını) göster.
<b>Recalculate Page Breaks</b> .....	Sayfa aralarını (sonlarını) yeniden hesaplar.
<b>Arrange Selection</b> .....	Yeniden düzenler.
<b>Arrange Tables</b> .....	Tabloları düzenler.
<b>Autosize Selected Tables</b> .....	Tabloları yeniden boyutlandırır.

## Tools Mönüsü

<b>Add Procedure</b> .....	Kod penceresinde yeni bir procedure yaratmak için kullanılır.
<b>Procedure Attribute</b> .....	Özellik ve metotları düzenler.
<b>Menu Editör</b> .....	Bir mөнüyü tasarlayan iletişim kutusu.,
<b>Options</b> .....	Visual Basic ortamını düzenlemek için kullanılır.

## Add-Ins Mönüsü

<b>Visual Data Manager</b> .....	Veritabanı düzenleme aracı.
<b>Add-In Manager</b> .....	Ek programları ekleme/kaldırma.

## Window Mönüsü

<b>Tile Horizontally</b> .....	Yatay döşeme.
<b>Tile Vertical</b> .....	Dikey döşeme.
<b>Cascade</b> .....	Basamaklama.
<b>Arrange Icons</b>	

## Help Mönüsü

<b>Contents</b> .....	İçerikler.
<b>Indx</b> .....	İndeks.
<b>Search</b> .....	Arama.
<b>Technical Support</b> .....	Teknik destek seçenekleri.
<b>Microsoft on the Web</b> .....	Microsoft'un Web sitesine bağlantı.
<b>About Microsoft Visual Basic</b>	Visual Basic programınız hakkında bilgiler.

## Options Mönüsü

### Editör Sekmesi

<b>Auto Syntax Check</b> .....	Yazılan kodların otomatik olarak yazım denetiminin yapılmasını sağlar.
<b>Require Variable Declaration</b>	Kod içinde kullanılan bütün değişkenlerin tanımlanmasını zorunlu kılar.
<b>Auto List Members</b> .....	Nesnelerin özelliklerinin otomatik olarak listelenmesini sağlar.
<b>Auto Quick Info</b> .....	Bir deyim yazılması durumunda parametrelerinin ekrana gelmesini sağlar.
<b>Auto Data Tips</b> .....	Program kesildikten sonra (debug modda) fare ile üzerine gidilen değişkenin değerini gösterir.
<b>Auto Indent</b> .....	İçerden başlamayı sağlar.
<b>Drag-and-Drop Text Editing</b> .	Kod satırları üzerinde fare ile seçip sürükleme işleminin yapılmasını sağlar.
<b>Default to Full Module View</b> .	Bir procedure yada bütün procedure'ların görülmesini sağlar.
<b>Procedure Separator</b> .....	Procedure'lar arası ayraç.

### Editör Format Sekmesi

<b>Normal Text</b> .....	Normal kod satırlarının rengini düzenler.
<b>Selection Text</b> .....	Seçilen kod satırlarının rengini düzenler.
<b>Syntax Error Text</b> .....	Sözdizimi hatası olan kod satırlarının rengini düzenler.
<b>Foreground</b> .....	Yazı rengi.
<b>Background</b> .....	Zemin rengi.
<b>Indicator</b> .....	Göstergenin rengi.

### General Sekmesi

<b>Show Grid</b> .....	Form üzerinde kareleme yapılarak kontrollerin hizalanmasına yardımcı olur.
<b>Align Controls to Grid</b> .....	Form üzerindeki kontrollerin hizalarının gridlere dayanmasını sağlar.
<b>Break on All Errors</b> .....	Derleyici herhangi bir hata ile karşılaştığında programı keser.
<b>Break in Class Module</b> .....	Class modulünde programın kesilmesi.
<b>Break on Unhandled Errors</b> ...	Bulunamayan hatalar oluştuğunda da programın kesilmesini sağlar.
<b>Compile on Demand</b> .....	İstendiğinde derleme.
<b>Background Compile</b> .....	Derleme arka planda yapılır.
<b>Show Tooltips</b> .....	Fare işaretçisinin herhangi bir kontrol elemanı üzerine gelmesi ile o kontrol hakkında bilgi verici bir metnin görüntülenmesi sağlanır.
<b>Collapse Proj.Hide Window</b> ...	Pencerelerin diğerleri arasına yuvalanarak açılmasını sağlar. İşaretlenmezse pencere diğerlerinin üzerinde bir yerde açılır.

### Docking Sekmesi

<b>Immediate Window</b> .....	Anlık görünüm penceresi.
<b>Locals Window</b> .....	Yerel değişkenleri penceresi
<b>Watch Window</b> .....	İzleme penceresi.
<b>Project Explorer</b> .....	projenin elemanlarını gösteren pencere.
<b>Properties Window</b> .....	Özellikler penceresi.
<b>Object Browser</b> .....	Nesne gösterici.
<b>Form Layout</b> .....	Formun ekrandaki görünümü.

<b>Toolbox</b> .....	Araç kutusu.
<b>Color Palette</b> .....	Renk Paleti.

## Environment Sekmesi

**Visual Basic başladığında**  
**Prompt for project** .....

Visual Basic'in açılmasının ardından hangi tip proje yaratılacağı sorulur.  
 AutoLoad dosyasına kayıt edilmiş dosyalardan oluşan bir proje başlatılır.

**Bir program başladığında**  
**Save Changes** .....

Program çalıştırılmadan önce yapılan değişiklikler kayıt edilir.  
 Program çalıştırılmadan önce yapılan değişiklikler kayıt edilmesi için soru sorulur.

**Don't Save Changes** .....

Program çalıştırılmadan önce yapılan değişiklikler kayıt edilmez.

### Şablonları göster

**Forms** .....

**MDI Forms** .....

**Modules** .....

**Class Modules** .....

**User Controls** .....

**Property Pages** .....

**User Document** .....

**Template Directory** .....

Formlar.

MDI Formlar.

Modüller.

Class Modüller.

Kullanıcı Kontrol Modülleri.

Özellik Sayfaları.

Kullanıcı Dokument Modülleri.

Şablonların saklanacağı klasör.

## Matematiksel İşaretler

^	Üs alma
*	Çarpma
/	Bölme
\	Tam Bölme (Bölme sonucunun tam kısmını alır)
+	Toplama
-	Çıkarma
=	Eşittir
<, <=	Küçük, küçük eşit
>, >=	Büyük, büyük eşit
<>	Eşitdeğil

## İlk Programım

Bir form açılır. Kod sayfasını açmak için formun üzerine çift tıklanır, Project Explorer penceresinden View Code tıklanır veya F7 tuşuna basılır.

Bir yordamın başlama ve bitirme deyimleri vardır.

**Private Sub Form\_Load()**

**End Sub**

### ***SHOW***

Sadece Form\_Load yordamında kullanılır ve form yüklendiği anda ekrana yazdığımız yazının gelmesi için kullanılır.

### ***PRINT***

Ekrana sabit bir bilgiyi yazmak için kullanılır.

### **ÖRNEK**

- 1- Show  
Print "Microsoft Visual Basic Dersine Hoş Geldiniz"
  - a. Fontbold=False
  - b. Fontitalic=True
2. Print "Visual Basic Öğretmeni"
3. Print "Oguz KUTLU"
  - a. Fontsize=24
  - b. Fontname="Times New Roman"
4. Print "PRESTIGE EDUCATION CENTER"
  - a. Fontunderline=True
  - b. Fontname="Arial"
5. Print "Bilgisayar Teknik Eğitim Merkezi"  
Forecolor=RGB(214,124,35)
1. Print "İngilizce Eğitim Merkezi"

**NOT :** Dikkat edileceği gibi kodlar yordam içerisinde yukarıdan aşağıya doğru ilerler. Yukarıda yapılan bir yazı büyüklüğü veya yazının rengi komutu aşağı kodlarda değiştirilmezse aynen uygulanır. Aynı kodları bir Command butonunun Click yordamına yazarsak baştaki Show komutunu yazmamıza gerek kalmaz.

Yazdırdığımız yazıları formumuzun herhangi bir noktasına yazdırabiliriz. Bunun için kullanılacak deyim **TWIP** komutudur. Twip komutu aslında Formun bir özelliği olan **Form1.scalewidth** ve **Form1.scaleheight** komutlarını kullanarak yapar.

- 1 inch = 1440 Twip
- 1 inch = 2,54 cm.
- 1 Twip = 0,0018 cm.

Şimdi yukarıdaki örneğimize şu satırları ekleyelim.

- a. CurrentX = 2500
- b. CurrentY = 2500

Formun en üstteki mavi bant hariç twip cinsinden ekran boyutları yaklaşık olarak

	640 X 480	800 x 600	1024 X 768
Form1.scalewidth	9600	12000	16000
Form1.scaleheight	7200	9000	12000

## DEĞİŞKENLER

Visual Basic program içersine yazılacak olan değişkenleri tanımlama zorunluluğu vermemektedir, ama büyük programlarda tanımlanmayan değişkenler bir sorun yaratabilirler. Bunun için Visual Basic'in tanıdığı değişkenler aşağıda gösterilmiştir. Bir procedure içinde tanımlanan değişkene local (yerel) değişken denir. Bu değişken bu procedure aktif olduğu zaman aktif olur. Bir değişkenin tüm modüllerde de geçerli kılmak için değişken tanımı Declarations kesiminde tanımlanır. Bunun için değişkeni PUBLIC deyimi kullanılarak tanımlamak gerekir. Yerel değişkenleri de STATİC deyimi ile tanımlarsak uygulama boyunca aktif olurlar.

**Public (Değişken) As (Değişken Tipi)** ` Yalnız Declarations kısmında tanımlanabilir.

**Static (Değişken) As (Değişken Tipi)**

**Dim (Değişken) As (Değişken Tipi)**

Değişkenlerin özellikleri

1. Bir harf ile başlar
2. En çok 255 karakterden oluşur.
3. İşaretler içermez.
4. Yazıldığı yordam içinde tek olmalı
5. Değişken adı olarak Visual Basic in kullandığı herhangi bir kod kullanılamaz.
6. Eğer değişken Declatations yordamına yazılırsa tüm procedure içinde geçerli olur.

Değişken Tipi	Veri Tipi	Alabileceği En Büyük Değer	Boyutu (Byte)	Değişken Tipi Karakter
Byte	İşaretsiz tam sayı	0 - 255	1	Yok
Integer	Tamsayı	± 32.767	2	%
Long	Uzun tamsayı	± 2.147.483.647	4	&
Single	Sayısal değişken	± 3,4 X 10 <sup>38</sup>	4	!
Double	Sayısal değişken	± 1,7 X 10 <sup>308</sup>	8	#
Currency	Para birimi	± 922..337.203.685.447 kuruş bölümünde 4 hane gösterir	8	@
Boolean	Evet-Hayır Olumlu-Olumsuz		2	Yok
Date	Tarih formatlarında	01/01/100 – 31/12/9999	8	Yok
String	Sayısal içeriği olmayan değişken	65.000 karakter	Her karakter 1 Byte	\$
Variant	Sayısal içeriği olmayan değişken		16 + Her karakter 1 Byte	Yok
Object	Resim ve image'ler için kullanılır.		4	Yok

### Sabit Değişken

Değişkenlerin aksine sabitler (constant), sahip oldukları değerler ile kullandığımız bilgi tanımlamalardır. Özellikle belli değerlerin kullanımını sağlar. Kodlamayı ve kodun okunabilirliğini kolaylaştırır. Bir sabitin uygulama boyunca geçerli olması için Declarations kesiminde tanımlanmalı ve Const deyiminden önce Public deyimi kullanılır.

### ÖRNEK

Const Pi = 3.14159265

Const Katsayı = 2,5

Const KDV = 1,17

## ***String***

Karakterlerden oluşan veriler String olarak tanımlanır. Örneğin müşteri adı gibi bir bilgi String olarak tanımlanmalıdır. String veriler sayıları içerebilir ancak sayısal işleme giremezler. String değişkenlere değer atamak için çift tırnak karakteri kullanılır. Bir String değişkene belli bir uzunluk da verilebilir. Bunun için asteriks (\*) karakterinden sonra uzunluk birimi girilir.

Dim s As String

S = "Oguz KUTLU"

Dim Adı As String \* 20

### **ÖRNEK**

1. 1.1

Show  
Dim a As String  
Dim b As String  
Dim c As String

a = 168  
b = 122  
c = a + b  
Print c

Show  
a\$ = 168  
b\$ = 122  
c\$ = a + b  
Print c

Sonuç = 168122 olur.

2. 2.1

Show  
Dim a As Integer  
Dim b As Integer  
Dim c As Integer

a = 168  
b = 122  
c = a + b  
Print c

Show  
a% = 168  
b% = 122  
c% = a + b  
Print c

Sonuç = 290 olur.

Sonuç = 290 olur.

Bazen programlarımızda değişkenlerin tiplerini değiştirmemiz gerekebilir. Bunun için aşağıdaki kodlar kullanılır.

CByte	Değişkeni Byte tipine çevirir
CInt	Değişkeni Integer tipine çevirir
CLng	Değişkeni Long tipine çevirir
CSng	Değişkeni Single tipine çevirir
CDbl	Değişkeni Double tipine çevirir
CCur	Değişkeni Currency tipine çevirir
CBool	Değişkeni Boolean tipine çevirir. Bu özellik pek kullanılmaz.
CDate	Değişkeni Date tipine çevirir
String	Değişkeni String tipine çevirir
CVar	Değişkeni Variant tipine çevirir

Kullanım şekli

CCur (Değişken adı)

### **ÖRNEK**

Dim a As Single,

....

....

a = CDbl (a)

***NOT :*** Değişken tiplerini çevirirken dikkat edilecek en önemli nokta özellikle büyük değişkenlerin küçüklere çevrilmesi sırasında veri kaybının olmaması veya virgüllü sayıların tam sayılara dönüştürülmesi sırasında oluşabilecek veri kayıpları.

## VarType () Fonksiyonu

Tipi belirtilmeyen deęişkenlere programın iřletimi sırasında Variant deęişken tipini alır. VarType fonksiyonu ile Variant bir deęişkenin hangi tipte bilgi içerdęi öğrenilir.

### VarType (Variant Deęişken Adı)

Veri Tipi	Sayısal Deęer
Boş – Empty	0
Null	1
Integer	2
Long	3
Single	4
Double	5
Currency	6
Date	7
String	8
Object	9
Error	10
Boolean	11
Variant	12
Data Object	13

### **ÖRNEK**

```
Private Sub Command1_Click()
```

```
Dim ad
ad = "Oguz KUTLU"
FontSize = 20
ForeColor = RGB(12, 35, 142)
CurrentX = 800
CurrentY = 800
veritipi = VarType(ad)
Print veritipi
```

```
ad = 3.254
FontSize = 30
ForeColor = RGB(101, 13, 142)
CurrentX = 800
veritipi = VarType(ad)
Print veritipi
```

```
End Sub
```

## ***MsgBox (Mesaj Kutusu)***

Bu deyim özellikle kullanıcıya belli bir mesaj verme ve kullanıcıdan belli bir yanıt almak için kullanılır. MsgBox deyimi ile bir mesaj görüntülenir. Bu mesaj sadece mesajda olabilir veya üzerinde onay düğmeleri bulunabilir. Örneęin kullanıcının bir dosyayı kayıt etmesi gerektięinde "Dosya Kaydedilsin mi?" gibi bir soru olabilir ve msgbox üzerinde de Yes No gibi yanıtlar alınabilir.

### **MsgBox ("mesaj", simgeler, "başlık")**

### **ÖRNEK**

1. MsgBox "Visual Basic Dersine Hořgeldiniz"
2. MsgBox "Prestige Educations Center"

## **NOT**

MsgBox deyimini Form Load yordamına yazılırsa form ekrana yüklendiği anda mesaj penceresi çıkar.

Windows uyumlu programlarda kullanıcıyı uyarmak veya mesaj vermek üzere ekrana getirilen mesaj kutularında Tamam düğmesinden başka düğmeler ve işaretler bulunur. Örneğin Excel ile çalışırken aktif belge kaydedilmeden kapatılmak istenirse, ekrana bir uyarı mesajı çıkar ve size seçme olanağı sunar. Bunun için iki seçenek vardır ve bilgisayarın sizin hangi tuşa bastığınızı anlaması için bir değişkene atamanız gerekmektedir.

## **ÖRNEK**

Dim a

a = MsgBox("Visual Basic Dersine Hoşgeldiniz")

Görüldüğü gibi herhangi bir simge belirtilmezse TAMAM düğmesi belirir. Diğer düğmeleri çıkartmak için aşağıdaki kodlar kullanılır.

<b>Sİmge</b>	<b>Değeri</b>	<b>Anlamı</b>
vbOKOnly	0	Sadece OK düğmesini gösterir.
vbOKCancel	1	OK ve Cancel düğmesini gösterir.
vbAbortRetryIgnore	2	Durdur, Yeniden dene, Yoksay
vbYesNoCancel	3	Yes, No, Cancel düğmelerini gösterir.
vbYesNo	4	Yes ve No düğmelerini gösterir.
vbRetryCancel	5	Retry ve Cancel düğmelerini gösterir.

## **ÖRNEK**

Dim a

a = MsgBox("Visual Basic Dersine Hoşgeldiniz", vbAbortRetryIgnore)

veya

a = MsgBox("Visual Basic Dersine Hoşgeldiniz", 2)

## **NOT**

Mesaj kutusunun içine iki satır mesaj yazmak için Enter tuşunun karşılına gelen ANSI 10 (Chr(10)) karakterini yazmamız gerekir.

## **ÖRNEK**

Dim a

a = MsgBox("Visual Basic Dersine Hoşgeldiniz" & Chr(10) & "ikinci satıra geçtim", vbYesNo)

Mesaj pencerelerinde bir de uyarı veya dikkat çekici ek ikonlar mevcuttur. Bunlar da aşağıda gösterilmiştir.

<b>Sİmge</b>	<b>Değeri</b>	<b>Anlamı</b>
vbCritical	16	X işareti
vbQuestion	32	Soru işareti
vbExclamation	48	Ünlem
vbInformation	64	i enformasyon işareti

### **ÖRNEK**

Dim a

```
a = MsgBox("Değişiklikler kaydedilsin mi?", vbYesNo + vbCritical)
```

Mesaj pencereleri üzerinde ki düğmelerin bir özelliği de aktif butonu seçme özelliğidir. Bunun için aşağıdaki kodlar yazılır.

<b>Simge</b>	<b>Değeri</b>	<b>Anlamı</b>
vbDefaultButton1	0	1.düğme aktif olacak
vbDefaultButton2	256	2.düğme aktif olacak
vbDefaultButton3	512	3.düğme aktif olacak
vbDefaultButton4	7682	4.düğme aktif olacak

### **ÖRNEK**

Dim a

```
a = MsgBox("Degisiklikler kaydedilsin mi?", vbYesNo + vbCritical + vbDefaultButton2)
```

Son olarak mesaj penceresinin başlığını da değiştirebiliriz. Bunun için yazdığımız kodun en sonuna ekleriz.

### **ÖRNEK**

Dim a

```
a = MsgBox("Degisiklikler kaydedilsin mi?", vbYesNo + vbCritical + vbDefaultButton2, "Kaydet Butonu")
```

Mesaj düğmesinde hangi düğmenin seçildiğine ilişkin bir değer döndürdüğünü belirlemek için aşağıdaki tablodaki değerler kullanılır.

<b>Sabit Bilgi</b>	<b>Değeri</b>	<b>Anlamı</b>
VbOK	1	OK düğmesi seçilmiş.
vbCancel	2	Cancel düğmesi seçilmiş.
vbAbort	3	Abort düğmesi seçilmiş.
vbRetry	4	Retry düğmesi seçilmiş.
vbIgnore	5	Ignore düğmesi seçilmiş.
vbYes	6	Yes düğmesi seçilmiş.
vbNo	7	No düğmesi seçilmiş.

### **ÖRNEK**

```
Private Sub Command1_Click()
```

```
Dim a, b, c
```

```
a = MsgBox("Degisiklikler kaydedilsin mi?", vbYesNo + vbCritical + vbDefaultButton2, "kaydet butonu")
```

```
c = 10
```

```
If a = vbYes Then
```

```
b = c * 10
```

```
Else
```

```
b = c * 2
```

```
End If
```

```
Print b
```

```
End Sub
```

Sonuç = yes butonuna basınca 100,  
no butonuna basınca 20 olur.

veya,

```
Dim a, b, c
```

```
a = MsgBox("Degisiklikler kaydedilsin mi?", 4 + vbCritical + vbDefaultButton2, "kaydet butonu")
```

```
c = 10
```

```
If a = 6 Then
```

```
b = c * 10
```

```
Else
b = c * 2
End If
Print b
```

Sonuç = yes butonuna basınca 100,  
no butonuna basınca 20 olur.

## ***InputBox (Bilgi Giriş Kutusu)***

Giriş kutusu kullanıcıdan bir değeri almayı sağlar. InputBox fonksiyonu bir iletişim kutusu görüntüleyerek kullanıcının bir değer girmesini sağlar. InpuBox fonksiyonu kullanıcıdan aldığı değeri programa geri döndürür.

**Değişken = InputBox ("mesaj", başlık, varsayım, xpoz, ypoz)**

### **ÖRNEK 1**

```
Private Sub Command1_Click()
    Dim ad As String
    ad = InputBox("Şehir Adını Giriniz : ?")
    Print ad
End Sub
```

Görüldüğü gibi ekranın tam ortasında bir bilgi giriş ekranı çıktı ve sol üst köşesinde projenin adı yazıyor. Giriş kutusunun adını değiştirmek için aşağıdaki örnek yazılır.

### **ÖRNEK 2**

```
Private Sub Command1_Click()
    Dim ad As String
    ad = InputBox("Şehir Adını Giriniz : ?", " Giriş Kutusu Başlığı")
    Print ad
End Sub
```

Sol üst köşesinde bu sefer bizim yazdığımız "Giriş Kutusu Başlığı" yazıyor. Giriş kutusunun üzerinde bazen default olarak bir şey yazmasını isteriz, örneğin şehir adlarını giriyoruz ve en çok girilen şehir İstanbul olduğu için onun default olmasını isteyebiliriz. Bunun için aşağıdaki örneği yazarsınız.

### **ÖRNEK 3**

```
Private Sub Command1_Click()
    Dim ad As String
    ad = InputBox("Şehir Adını Giriniz : ?", " Giriş Kutusu Başlığı", "İstanbul")
    Print ad
End Sub
```

Eğer aksi bir durum belirtmediyse açılan InputBox kutusu ekranın tam ortasında açılır. Bazen açılan bu kutuları kullanıcıların ekranı daha iyi görebilmesi için değişik yerlere koyabiliriz. Bunun için aşağıdaki örneği yazarsınız.

### **ÖRNEK 4**

```
Private Sub Command1_Click()
    Dim ad As String
    ad = InputBox("Şehir Adını Giriniz : ?", " Giriş Kutusu Başlığı", " İstanbul ", 3500, 500)
    Print ad
End Sub
```

### **ÖRNEK 5**

İki tane arka arkaya sayı girilerek girilen iki sayının toplamı MsgBox ile bize bildirilsin. MsgBox penceresinde information da olsun.

```
Private Sub Command1_Click()
    Dim a As Integer
    Dim b As Integer
```

```

Dim c As Integer
Dim d As Integer
a = InputBox("1. sayıyı giriniz")
b = InputBox("2. sayıyı giriniz")
c = a + b
MsgBox("iki sayının toplamı = " & c & ,vbInformation)
End Sub

```

**NOT:** MsgBox yazımında kullanılan & (ampersant) işareti girilen değeri mesajbox kutusuna taşır.

## Döngüler

### ***If ..... Then ..... Else ..... End if***

If deyimi belli bir ifadenin değerine göre bir deyimin işletilmesini sağlar. If deyiminin basitten karmaşığa doğru değişik kalıpları vardır.

1. **if** (koşul) **Then** (işlem)
2. **if** (koşul) **Then**  
    (işlem1)  
**Else**  
    (işlem2)  
**End if**
3. **if** (koşul1) **Then**  
    (işlem1)  
**ElseIf** (koşul2) **Then**  
    (işlem2)  
**Else**  
    (işlem-n)  
**End if**

### **ÖRNEK 1**

İki sayı girin ve Ok basınca iki sayının çarpımını Cancel 'e basınca toplamını alsın ve sonucu MsgBox ile ekranda gösterson.

```

Private Sub Command1_Click()
Dim a As Integer, b As Integer
Dim c As Integer
Dim d As Integer
a = InputBox("1.sayıyı giriniz")
b = InputBox("2.sayıyı giriniz")
c = MsgBox("Sayılar toplansın mı? yoksa çarpılsın mı?", vbOKCancel)
If c = 1 Then
d = a * b
MsgBox ("İki sayının çarpımı =" & d)
End If
If c = 2 Then
d = a + b
MsgBox ("İki sayının toplamı =" & d)
End If
End Sub

```

### **ÖRNEK 2**

Bir öğrencinin aldığı notu girelim ve aldığı not < 50 ise ekrana zayıf not aldı , >=50 ise ekrana geçer not aldı yazsın.

```

Private Sub Command1_Click()
Dim a As Integer
a = InputBox("Öğrencinin notunu giriniz")
If a < 50 Then
MsgBox ("Bu öğrenci zayıf not aldı")
End If

```

```
If a >= 50 Then
    MsgBox ("Bu öğrenci geçer not aldı")
End If
End Sub
```

### **ÖRNEK 2.1**

```
Private Sub Command1_Click()
Dim a As Integer

a = InputBox("Öğrencinin notunu giriniz")

If a < 50 Then
    MsgBox ("Bu öğrenci zayıf not aldı")
Else
    MsgBox ("Bu öğrenci geçer not aldı")
End If

End Sub
```

### **ÖRNEK 3**

Klavyeden girilen 3 sayının en büyük olanını bulmak.

```
Private Sub Command1_Click()
Dim a As Double
Dim b As Double
Dim c As Double

a = InputBox("birinci sayıyı giriniz")
b = InputBox("ikinci sayıyı giriniz")
c = InputBox("üçüncü sayıyı giriniz")

If a > b Then
    If a > c Then MsgBox "En Büyük sayı birinci sayıdır."
    ElseIf c > b Then MsgBox "En Büyük sayı üçüncü sayıdır"
Else
    MsgBox "En büyük sayı ikinci sayıdır."
End If
End Sub
```

### **ÖRNEK 4**

Elektrik faturası hesaplayan bir program 150 kw kadar 44.000 TL. 150 – 300 kw arası için 66.000 TL. 300 den sonrası için 75.000 TL. olsun.

```
Private Sub Command1_Click()
Dim tüketim As Double
Dim para As Double

tüketim = InputBox("Tüketim Değerini Girin")
If tüketim > 0 And tüketim < 151 Then
    para = tüketim * 44000
ElseIf tüketim > 150 And tüketim < 301 Then
    para = (150 * 44000) + ((tüketim - 30) * 66000)
ElseIf tüketim > 301 Then
    para = (150 * 44000) + (150 * 66000) + ((tüketim - 300) * 75000)
End If
MsgBox ("ödeyeceğiniz para =" & para)
End Sub
```

### **ÖRNEK 5**

3 bu örnek de 1.vize ,2. vize ve final notları girilecek ve puanı 50'den büyükse msgbox la geçtiniz değilse bütünleme notunu istet ve sonuc yine 50 den küçükse kalsın değilse geçsin. Vizenin %40 Finalin % 60

Dim a As Integer ,b As Integer ,c As Integer, d As Double, e As Double, f As Double, g As Integer, but As Double

a:

```
a = InputBox("Birinci vize notu giriniz")
  If a < 0 Or a > 100 Then
    MsgBox ("Lütfen 0 - 100 arasında bir sayı giriniz")
    GoTo a
```

Else

b:

```
b = InputBox("ikinci vize notu giriniz")
  If b < 0 Or b > 100 Then
    MsgBox ("Lütfen 0 - 100 arasında bir sayı giriniz")
    GoTo b
```

Else

c:

```
c = InputBox("Final notunu giriniz")
  If c < 0 Or c > 100 Then
    MsgBox ("Lütfen 0 - 100 arasında bir sayı giriniz")
    GoTo c
```

Else

```
    d = ((a + b) / 2) * 0.4
    e = (c * 0.6) + d
```

If e > 50 Then

```
MsgBox ("Geçtiniz")
```

```
FontBold = True
```

```
FontSize = 12
```

```
FontName = "tahoma"
```

```
ForeColor = RGB(15, 65, 85)
```

```
Print "Ortalama Notunuz =" & e
```

Else

' her if muhakkak end if le kapatılmalı iç içe çok sayıda if konabilir. if tek satırda bitirilebilir.

```
MsgBox ("Kaldınız")
```

f:

```
f = InputBox("Bütünleme notunu giriniz")
  If f < 0 Or f > 100 Then
    MsgBox ("Lütfen 0 - 100 arasında bir sayı giriniz")
    GoTo f
```

Else

```
    but = (f * 0.6) + d
```

```
    If but > 50 Then
```

```
      MsgBox ("geçtiniz")
```

```
    Else
```

```
      MsgBox ("artık sizi kimse kurtaramaz")
```

```
    End If
```

```
    Print "Son notunuz =" & but
```

```
End If
```

```
End If
```

```
End If
```

```
End If
```

```
End If
```

### **ÖRNEK 6**

```
Private Sub Command1_Click()
```

'or ve and kullanma

```
Dim a As Integer
```

```
Dim b As Integer
```

```
Dim c As Integer
```

```
Dim d As Double
Dim e As Double
Dim f As Double
Dim but As Double
```

```
a = InputBox("Birinci vize notu giriniz")
b = InputBox("ikinci vize notu giriniz")
c = InputBox("Final notunu giriniz")
```

```
d = ((a + b) / 2) * 0.4
e = (c * 0.6) + d
```

```
If e < 50 Then
MsgBox ("Kaldınız")
```

```
    ElseIf e >= 50 And e < 65 Then
MsgBox ("Birinci gruptasınız")
```

```
    ElseIf e >= 65 And e < 80 Then
MsgBox ("ikinci gruptasınız")
```

```
    ElseIf e >= 70 And e < 100 Then
MsgBox ("Ben birtaneyim")
```

```
End If
Print "Son notunuz ="; e
End Sub
```

### **Select Case**

Kullanımı Kalıbı

```
'select case değişken
' case
' case
'end select
```

### **ÖRNEK 1**

Klavyede 0 - 5 arasında bir rakam girilecek ve ne girildiğini msgbox la bize göstereceğiz

### **'birinci bölüm**

```
Dim a As Integer
```

```
a = InputBox("1 - 5 arasında bir sayı giriniz")
```

```
Select Case a
```

```
    Case 1
MsgBox ("girilen sayı 1'dir")
    Case 2
MsgBox ("girilen sayı 2'dir")
    Case 3
MsgBox ("girilen sayı 3'dür")
    Case 4
MsgBox ("girilen sayı 4'dür")
    Case 5
MsgBox ("girilen sayı 5'dir")
```

## **'ikinci bölüm**

'mesala 8 rakamini girin

```
Case Else
MsgBox ("boyle bir sayi girilemez")
```

## **'üçüncü bölüm**

'5-100 arasındaki tüm sayılar için

```
Case 5 To 100
```

'yazılır

End Select

## **ÖRNEK 2**

Dim a As Integer ,b As Integer, c As Integer, d As Double, e As Double, f As Double, but As Double

```
a = InputBox("Birinci vize notu giriniz")
b = InputBox("ikinci vize notu giriniz")
c = InputBox("Final notunu giriniz")
```

```
d = ((a + b) / 2) * 0.4
e = (c * 0.6) + d
```

Select Case e

```
'Case e >= 0, e <= 100: (: case in bittiğini anlatır.)
'case e="ali",e="mehmet": (kullanılabilir.)
```

```
Case 0 To 49
MsgBox ("kaldınız")
Case 50 To 59
MsgBox ("geçer not aldınız")
Case 60 To 84
MsgBox ("sıralamaya girmek üzeresiniz")
Case 85 To 94
MsgBox ("az kaldı")
Case 95 To 100
MsgBox ("iste bu")
'case else kullanılabilir.
```

```
Case Is < 0 > 100
MsgBox ("girdiğin not yanlıştır lütfen kontrol ediniz.")
```

End Select

Print e

## **ÖRNEK 3**

```
Private Sub Command13_Click()
Dim a
Dim b
Dim c
a = InputBox("bir harfe basınız")
Select Case a
Case "A" To "G"
c = "A ile G arasında bir harfe bastınız"
Case "H" To "K"
c = "H ile K arasında bir harfe bastınız"
Case "L" To "Z"
c = "L ile G arasında bir harfe bastınız"
```

```
Case Else
c = "yanlış bir tuşa bastınız"
End Select
```

```
MsgBox c
```

```
End Sub
```

### ***For ..... Next***

Bir grup deyim belli sayıda çalıştırarak bir döngü oluşturur. Döngü bir kez işledikten sonra artırım değeri eklenir. Eğer herhangi bir artırım değeri kullanılmadıysa o zaman bir artırılır. Sayaç değeri bitiş değerinden büyük olduğu zaman döngü sona erdirilir ve program Next deyiminden itibaren devam eder.

Döngü değimleri amaç değişkenlerin değerlerini belli aralıklarla yaptırmak. Belli aralıklardaki sayıların toplamı , çıkartmak,çarpmak

Kalıp

```
FOR değişken=1 TO 70
'
'
'
NEXT
```

### **ÖRNEK 1**

```
Private Sub Command1_Click()
```

'1 den 10 a kadar olan sayıları toplatalım

```
Dim a As Integer
```

```
Dim b As Integer
```

```
For a = 1 To 10
```

```
    b = b + a 'buradaki yeni b= kavramı mat. = değil
```

```
Next
```

```
Print b
```

```
End Sub
```

### **ÖRNEK 2**

```
Private Sub Command1_Click()
```

'1 den 10 a kadar olan sayıları toplatalım print e dikkat

```
Dim a As Integer
```

```
Dim b As Integer
```

```
For a = 1 To 10
```

```
    b = b + a 'buradaki yeni b= kavramı mat. = değil
```

```
Print b
```

```
Next
```

' olayın farkını anlat

```
End Sub
```

### **ÖRNEK 3**

```
Private Sub Command1_Click()  
  
'klavyeden bir sayı girilecek ve  
'makina o sayıya kadar olan sayıların toplamını alacak  
  
Dim a As Double  
Dim b As Double  
Dim c As Double  
  
a = InputBox("birinci sayıyı girin")  
  
For b = 1 To a  
c = c + b  
  
Next  
  
Print ("1 den ") & a & ("ya kadar olan sayıların toplamı"), c  
  
End Sub
```

### **ÖRNEK 4**

```
Private Sub Command1_Click()  
  
' Belli bir sayıya kadar toplama yapmak  
' Bir atlatacak toplatmak için  
' step komutu kullanılarak  
  
Dim a As Double  
Dim b As Double  
Dim c As Double  
  
a = InputBox("birinci sayıyı girin")  
  
For b = 1 To a Step 2  
  
c = c + b  
  
Next  
Print c  
End Sub
```

### **ÖRNEK 5**

```
Private Sub Command11_Click()  
  
' 1 den girilen bir sayıya kadar olan sayıların çarpımı  
' en fazla 170 ! e kadar yapabilirsiniz.  
  
Dim a As Double, b As Double, c As Double  
  
a = InputBox("birinci sayıyı girin")  
c = 1  
For b = 1 To a  
c = c * b  
Next  
Print a; "! ="; c  
End Sub
```

## **ÖRNEK 6**

Private Sub Command12\_Click()

' üç boyutlu bir yazı yazılması

Dim t, i, x1, y1, x, y

x = 200

y = 200

t = "Oguz Kemalettin KUTLU"

FontName = "tahoma"

FontSize = 30

For i = 0 To 15

x1 = x + i \* 5 ' 10 birim sağa kayar

y1 = y + i \* 10 ' 10 birim aşağıya kayar

ForeColor = QBColor(i) 'QBColor (0 - 15 ) ARASI ÖZEL RENKLER

'forecolor=rgb(red,green,blue)

' qbcolor kullanılacaksa eğer to 16 olamaz

CurrentX = x1

CurrentY = y1

Print t

Next ' döngüyü anlatmayı unutma

End Sub

## **ÖRNEK 7**

'Private Sub Form\_Resize() ' formun boyutu değişse bile yazı ona uygun hale gelir.

'Cls ' önce ekranın temizlenmesi lazım

'Show

'Dim t, i

't = "Oguz KUTLU – Visual Basic Öğretmeni"

'FontName = "tahoma"

'For i = 1 To 10

'FontSize = i \* 5

'CurrentX = (ScaleWidth - TextWidth(t)) / 2 ' yazıyı tam merkezden başlatmak için yazıyı 10 kere büyütürük aşağıya doğru yazarak iner

'Print t

'Next

'End Sub

## **QBColor ()**

0	Siyah	8	Gri
1	Mavi	9	Açık Mavi
2	Yeşil	10	Açık Yeşil
3	Cyan	11	Açık Cyan
4	Kırmızı	12	Açık Kırmızı
5	Eflatun	13	Açık Eflatun
6	Sarı	14	Açık Sarı
7	Beyaz	15	Açık Beyaz

## **RGB ()**

Renk	Kırmızı	Yeşil	Mavi
Siyah	0	0	0
Mavi	0	0	255
Yeşil	0	255	0
Cyan	0	255	255
Kırmızı	255	0	0
Eflatun	255	0	255
Sarı	255	255	0
Beyaz	255	255	255

## Disk ve Dosya Komutları

Program içerisinde kullanacağımız bazı kodlar sayesinde çalışma anında geçerli klasör veya dizinin adını öğrenebilir, başka bir dizine geçebilir, istediğiniz dizini veya dosyayı silebilir, yeni bir dizin oluşturabiliriz.

### **CurDir**

Aktif dizini gösterir.

```
Private Sub Command1_Click()  
    MsgBox CurDir  
End Sub
```

### **ChDir**

Aktif dizini değiştirir.

```
Private Sub Command1_Click()  
    ChDir "c:"  
    MsgBox CurDir  
End Sub
```

### **ChDrive**

Geçerli sürücüyü değiştirir.

```
Private Sub Command1_Click()  
    ChDrive "c"  
    ChDir "c:\program files"  
    MsgBox CurDir  
End Sub
```

### **Mkdir**

Herhangi bir sürücüde dizin oluşturmak için kullanılır.

```
Private Sub Command1_Click()  
    Mkdir "c:\deneme klasör"  
End Sub
```

### **Rmdir**

Herhangi bir sürücüde ki içi boş olan bir dizini silmek için kullanılır.

```
Private Sub Command1_Click()  
    Rmdir "c:\deneme klasör"  
    'Dikkat edileceği gibi silinen klasör çöp kutusuna gitmedi. 'eğer dizinin içinde bir dosya kaldıysa yine hata verir.  
    Hatayı önlemek için  
        On Local Error GoTo hata  
        Rmdir "c:\deneme klasör"  
        Exit Sub  
    hata:  
        MsgBox ("Dizin silinemedi " & Error)  
End Sub
```

### **FileCopy**

Bir klasörden başka bir klasöre kopyalamak için kullanılır.

'öncelikle c:\belgelerim 'e bir tane kopya.txt dosyası jazırlanır.

```
Private Sub Command4_Click()  
    FileCopy "c:\belgelerim\kopya.txt", "c:\deneme klasör\kopya.txt" : End Sub
```

### ***FileLen***

Bir dosyanın byte olarak büyüklüğünü verir.

```
Private Sub Command5_Click()  
    MsgBox (FileLen("c:\deneme klasör\kopya.txt"))  
    'ikinci olarak dosyaya bir şeyler yazıp tekrar deneyin  
End Sub
```

### ***FileDateTime***

Bir dosyanın en son değiştirildiği tarihi öğrenmek için kullanılır.

```
Private Sub Command6_Click()  
    MsgBox (FileDateTime("c:\deneme klasör\kopya.txt"))  
End Sub
```

### ***GetAttr***

Bir dosyanın özelliklerini öğrenmek için kullanılır. Bu komut normalde geriye bir rakam döndürür.

#### ***Örnek 1***

```
Private Sub Command7_Click()  
    MsgBox (GetAttr("c:\deneme klasör\kopya.txt"))  
End Sub
```

VbNormal	0	Özelliği olmayan dosya
vbReadOnly	1	Salt okunur dosya
VbHidden	2	Gizli dosya
VbSystem	4	Sistem dosyası
VbDirectory	16	Klasör
VbArchive	32	Arşiv dosyası

#### ***Örnek 2***

```
Private Sub Command8_Click()  
Cls  
Dim dosya  
    dosya = GetAttr("c:\deneme klasör\kopya.txt")  
Print "kopya.txt"  
    If dosya And 0 Then Print "Normal Dosya"  
    If dosya And 1 Then Print "Salt Okunur"  
    If dosya And 2 Then Print "Gizli Dosya"  
    If dosya And 4 Then Print "Sistem Dosyası"  
    If dosya And 16 Then Print "Klasör"  
    If dosya And 32 Then Print "Arşiv Dosyası"  
End Sub
```

### ***SetAttr***

Bir dosyanın özelliklerini değiştirmek için kullanılır.

#### ***Örnek 1***

```
Private Sub Command9_Click()  
    SetAttr "c:\deneme klasör\kopya.txt", 1 + 2 + 4 + 32  
End Sub
```

#### ***Örnek 2***

```
Private Sub Command9_Click()  
    SetAttr "c:\deneme klasör\kopya.txt", vbNormal + vbReadOnly + vbHidden + vbSystem + vbArchive  
End Sub
```

## ***Name***

İstediğimiz bir klasörün veya bir dosyanın adını değiştirmek için kullanılır.

### **Örnek 1**

```
Private Sub Command11_Click()  
    Name "c:\belgelerim\kopya.txt" As "c:\deneme klasör\kopya2.txt"  
End Sub
```

NOT: Program ikinci kez çalıştırıldığında dikkat edileceği gibi hata komutu verir. Bunu önlemek için,

### **Örnek 2**

```
Private Sub Command12_Click()  
On Error GoTo k  
    Name "c:\belgelerim\kopya.txt" As "c:\deneme klasör\kopya2.txt"  
Exit Sub  
k:  
    MsgBox ("Böyle bir dosya bulunamadı")  
End Sub
```

## ***Dir***

İstediğimiz bir klasördeki dosyaların listesini göstermek için kullanılır.

### **Örnek 1**

```
Private Sub Command10_Click()  
Cls  
Dim dosya  
    dosya = Dir("c:\*.*)"   
Print dosya: End Sub
```

### **Örnek 2**

```
Private Sub Command10_Click()  
Cls  
Dim dosya  
    dosya = Dir("c:\*.*)"   
    While dosya <> "" ` Başka dosya kalmayıncaya kadar devam et.  
Print dosya  
    dosya = Dir  
    Wend
```

End Sub

## ***Kill***

İstediğimiz bir klasörün veya bir dosyanın silinmesi için kullanılır.

Bu işlemi yaparken çok dikkatli olmamız lazım çünkü silinen dosya çöp kutusuna gitmez.

### **Örnek 1**

```
Private Sub Command13_Click()  
    Kill "c:\deneme klasör\kopya.txt"  
End Sub
```

### **Örnek 2**

```
Private Sub Command14_Click()  
On Error GoTo k  
    Kill "c:\deneme klasör\kopya.txt"  
Exit Sub  
k:  
    MsgBox ("Böyle bir dosya bulunamadı")  
End Sub
```

### **Örnek 3**

```
Private Sub Command15_Click()  
On Error GoTo k  
Dim c  
    c = MsgBox("dosya silinecek mi ?", vbYesNo)  
    If c = vbYes Then  
        Kill "c:\deneme klasör\kopya.txt"  
        MsgBox ("dosya silindi")  
    Exit Sub  
k:  
    MsgBox ("Böyle bir dosya bulunamadı")  
    End If  
End Sub
```

### **End**

Bir formu sonlandırmak için kullanılır.

```
Private Sub Command17_Click()  
    End  
End Sub
```

## **Çizim Yapmak**

Bir form üzerine çizim yapmak için daha önce belirttiğimiz gibi ScaleWidth(X) ve ScaleHeight(Y) koordinatlarından ve twip ölçü birimlerinden yararlanacağız. Form üzerinde çizeceğimiz bir noktanın koordinatını belirtirken formun üst sol köşesi (0,0) dir ve X değeri sağa doğru Y değeri ise aşağıya doğru artar. Formun üstündeki mavi bar bu değerler dışındadır.

### **Line**

İstenilen koordinatlar arasında çizgi çizmek veya kutu çizmek için kullanılır.

Line(X<sub>başlangıç</sub>,Y<sub>başlangıç</sub>)-(X<sub>bitis</sub>,Y<sub>bitis</sub>),çizgi rengi ,BF

BF = Otomatik kutu yapmak için kullanılır.

### **Örnek 1**

```
Private Sub Command1_Click()  
Cls  
    Line (500, 500)-(2500, 2500)  
End Sub
```

### **Örnek 2**

```
Private Sub Command2_Click()  
Cls  
    Line (500, 500)-(2500, 2500), RGB(200, 25, 12)  
End Sub
```

### **Örnek 3**

```
Private Sub Command3_Click()  
Cls  
    Line (500, 500)-(2500, 2500), QBColor(5)  
End Sub
```

### **Örnek 4**

```
Private Sub Command4_Click()  
    Line (500, 500)-(2500, 2500), RGB(200, 25, 12), B  
End Sub
```

### **Örnek 5**

```
Private Sub Command4_Click()  
    Line (500, 500)-(2500, 2500), RGB(200, 25, 12), BF  
End Sub
```

### **Örnek 6**

*Klavyeden girilen rakamlarla bir kutu çizmek.*

```
Private Sub Command5_Click()  
    Dim a, b, c, d, e, f, g  
    a = InputBox("xbas")  
    b = InputBox("ybas")  
    c = InputBox("xbit")  
    d = InputBox("xbit")  
    e = InputBox("R rengi için 255 e kadar bir sayı gir")  
    f = InputBox("G rengi için 255 e kadar bir sayı gir")  
    g = InputBox("B rengi için 255 e kadar bir sayı gir")  
    Line (a, b)-(c, d), RGB(e, f, g), BF  
End Sub
```

### **Örnek 7**

*BF kodunu kullanmadan bir kutu çizmek.*

```
Private Sub Command6_Click()  
Cls  
    Line (500, 500)-(500, 2000), RGB(250, 12, 35)  
    Line (2000, 500)-(2000, 2000), RGB(250, 12, 135)  
    Line (500, 2000)-(2000, 2000), RGB(25, 12, 235)  
    Line (500, 500)-(2000, 500), RGB(35, 112, 35)  
  
End Sub
```

NOT= DrawWidth özelliğini anlat

### **Örnek 8**

```
Private Sub Form_Resize()  
Show  
Cls  
Dim i, stp  
Form1.Width = Form1.Height  
stp = Form1.ScaleWidth / 20  
For i = 0 To Form1.ScaleHeight / 2 Step stp  
ForeColor = i * 12  
Line (i, i)-(Form1.ScaleWidth - i, Form1.ScaleHeight - i), , BF  
Next  
End Sub
```

### **Örnek 9**

Bir üçgen çizmek için

```
Private Sub Command2_Click()  
Cls  
DrawWidth = 3  
Line (500, 500)-(2000, 2000)  
Line -(500, 2000)  
Line -(500, 500)  
End Sub
```

## **Circle**

İstenilen koordinatlar arasında daire çizmek için kullanılır.

Circle( $X_{\text{başlangıç}}$ ,  $Y_{\text{başlangıç}}$ ), Yarı çap, çizgi rengi

### **Örnek 1**

```
Private Sub Command1_Click()  
    DrawWidth = 3  
    Circle (1000, 1000), 1000  
End Sub
```

### **Örnek 2**

```
Private Sub Command1_Click()  
    DrawWidth = 3  
    Circle (1000, 1000), 1000, RGB(125,32,14)  
End Sub
```

### **Örnek 3**

```
Private Sub Command1_Click()  
    DrawWidth = 3  
    Circle (1000, 1000), 1000, QBColor(2)  
End Sub
```

### **Örnek 4**

```
Private Sub Command2_Click()  
Cls  
    Dim i, j  
    For j = 1 To 15  
        For i = 750 To 3550 Step 200  
            Circle (i + 2000, 2500), Qbcolor(j)  
        Next  
    Next  
End Sub
```

### **Örnek 5**

```
Private Sub Command15_Click()  
Dim a, b, xort, yort  
DrawWidth = 3  
xort=scalewidth/2  
yort=scaleheight/2  
    b = 500  
    For a = 0 To 15  
        Circle (xort,yort), b, QBColor(a) '16 ana renk için  
        b = b + 1000  
    Next  
End Sub
```

### **Örnek 6**

```
Private Sub Form_Load()  
Show  
Cls  
Dim x, y  
DrawWidth = 2  
    x = ScaleWidth / 2  
    y = ScaleHeight / 2  
Circle (x, y), x / 2  
End Sub
```

'Bu örneği Form\_Resize yordamına yazarsak formun boyutu değiştiği anda oda değişir.

### **Örnek 7**

Aslında Circle deyiminin alacağı iki üç parametre daha vardır. 4. parametredeki rakam 0 olursa tam çember 6 olursa da çemberin bir kısmı çizilir. 4. parametre 0-6 arasında değer alabilir.

```
Private Sub Command1_Click()  
    DrawWidth = 4  
    Circle (2500, 2500), 1000, RGB(255, 0, 0), 1 'NOT=rakam değiştirilerek denemeler yapılır.  
End Sub
```

### **Örnek 8**

5. parametredeki rakam 6.27 olursa tam çember 0-6.27 arasında rakam alırsa 5.parametrenin tersi işlem yapar.

```
Private Sub Command1_Click()  
    DrawWidth = 4  
    Circle (2500, 2500), 1000, RGB(255, 0, 0), 0, 6.27  
End Sub
```

### **Örnek 9**

7.parametredeki rakam 0-1 arasında olursa çember enine basıklaşır. 1 den büyük rakam verirsek bu sefer dikine basıklaşır.

```
Private Sub Command1_Click()  
    DrawWidth = 4  
    Circle (2500, 2500), 1000, RGB(255, 0, 0), 0, , 0.4 'NOT = denemeler yap  
End Sub
```

### **Örnek 10**

```
Private Sub Form_Resize()  
    Cls  
    Dim x As Integer  
    Dim y As Integer  
    DrawWidth = 3  
    x = ScaleWidth / 2  
    y = ScaleHeight / 2  
    Circle (x, y), x / 2, RGB(0, 0, 255)  
    Line (x - x / 2, y)-(x + x / 2, y)  
    Line (x, y+ x / 2)-(x, y- x / 2)  
End Sub
```

### **PSet**

Formun herhangi bir noktasına nokta koymak için kullanılır.

PSet (X<sub>başlangıç</sub>, Y<sub>başlangıç</sub>), çizgi rengi

### **Örnek 1**

PSet (500, 500), RGB(35,24,111)

### **Örnek 2**

```
Private Sub Command1_Click()  
    Dim i  
    DrawWidth = 4  
    For i = 500 To 2500 Step 10  
    PSet (i, i), QBColor(4)  
    Next  
End Sub
```

### **Örnek 3**

```
Private Sub Command1_Click()  
Cls  
Dim a, b, r, t, x, y  
For t = -100 To 100 Step 0.01  
    a = ScaleWidth / 2  
    b = ScaleHeight / 2  
    r = Sin(6 * t) * Cos(6 * t)  
    x = r * Cos(t)  
    y = r * Sin(t)  
    PSet (a + x * 3800, b - y * 3800)  
Next  
End Sub
```

### **Örnek 4**

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, x As Single, y As Single)  
If Button = 1 Then  
    ForeColor = RGB(255, 0, 0)  
    DrawWidth = 3  
    PSet (x, y)  
End If  
End Sub
```

## **Random Dosyalar**

- '1. 6 tane label ciz
- '2. 6 tane text ciz
- '3. adlarını yaz. Kayıt No, Adınız, Soyadınız, Telefon, Adres, Şehir  
 '3.1 Modül ekle
- '4. Kaydet butonu ekle
- '5. Önceki kayıt
- '6. Sonraki kayıt
- '7. İlk kayıt
- '8. Son kayıt
- '9. Kayıt bul
- '10. Degistir
- '11. Kayıt ekle
- '12. Sil
- '13. Toplam ve bulunan kayıt sayısı sırası yine command sırasıyla
- '14. Çıkis

```
Private Sub Form_Load()
```

```
Open "oguz.dat" For Random As #1 Len = Len(alanlar) ` form yüklendiği anda bir dosya açmak için  
kullanılır. oguz.dat dosya isim
```

```
kayitno = LOF(1) / Len(alanlar)
```

```
kayitno = kayitno + 1
```

```
topkayit.Caption = kayitno - 1
```

```
suankayit.Caption = kayitno
```

```
kayitno.Enabled = False
```

```
End Sub
```

'**For Random** dosyanın random dosya olduğunu gösterir. **As 1** dosya numarası 255 e kadar verilebilir.  
'başlangıçta verdiğimiz dosya numarası sonra kullanılacak. **Len (alanlar)** alanların uzunluğunu len değişkenine  
'atamamız gerekmektedir.

'len alanlar toplam uzunluğu 'lof ise 1 numaraları dosyanın toplam uzunluğu 'toplam uzunluğu alanlar  
'uzunluğuna bölersek kayit no yu gösterir.

'kayitno=kayitno+1 ilk anda kayıt boş olduğu zaman hata vermemek için  
'ilk çalıştırdığımızda oguz.dat bulamayacağı için ilk çalıştırma anında kendisi oluşturur.  
'Random dosyaların bir özelliği de her kayda aynı oranda uzunluk ayırır.

'örneğin 5 dosya var ve hepsine 10 uzunluk ayırdığını düşünelim

'50 / 10 = 5 çıkar ama +1 yapar ve 6 dosyayı bulur.  
'artık aktif dizinde oguz.dat dosyası bulunur.

' Değişkenlerin tüm procedür içinde geçerli olması için modülde bir tip tanımlaması yapmak zorundayız.  
' **Type** baslar ve **End Type** biter. Type bilgileri Text Kutuları için yapılır. Oluşturulacak text'leri makineye tanıttık.

## General\_Declarations Option Explicit

### Type kayıt bilgisi

```
ad As String * 10
soyad As String * 15
telefon As String * 10 'tel. numaralarının (216)4755054 göstermek için maskedit component inde anlatacağız.
adres As String * 30
sehir As String * 10
```

### End Type

### Global alanlar As kayıt bilgisi

' kayıt bilgisi tipini tüm modül ve formlara tanıtmak için kullanılır. **alanlar**. koyduğumuz anda tanımladığımız type'ler otomatik olarak ekrana gelir modulden global yazısı kaldırılınca ekrana gelmez.

'text kutularına yazdığımız yazılar dikkat ederseniz programdan çıkılıp tekrar girildiği zaman yazdığımız yazıların kaydedilmediğini görürüz. Kaydetmek için bir kaydet butonu hazırlar ve içerisine aşağıdaki kod yazılır.

```
Private Sub kaydet_Click()
    Dim c As Integer
```

```
alanlar.ad = form1.ad
alanlar.soyad = form1.soyad
alanlar.telefon = form1.telefon
alanlar.adres = form1.adres
alanlar.sehir = form1.sehir
```

'dikkat edilirse form1 den sonra ad,şehir gelmesinin kolaylığı görülmüş oldu

```
c = MsgBox("Degisiklikler kayddilsin mi ?", vbYesNo)
If c = vbYes Then
    'yes e basıldığı zaman verilerin dosyaya yazılmasını istiyorum bunun için PUT değimi kullanılır.
```

### Put #1, kayıtno, alanlar

'bunun anlamı yeni kaydı alanlar içine kayıtno ya göre kaydetmeyi sağlar 1 rakamı ilk başta verdiğimiz dosya noyla aynı olması lazım bu işlemden sonra text kutularını boşaltıp kayıtno yu arttırıp ekrana getirsin.

```
form1.ad = " "
form1.soyad = " "
form1.telefon = " "
form1.adres = " "
form1.sehir = " "
```

'text kutuları boşaldıktan sonra yeni kayda geçince kayıtno artması gerekir

```
kayıtno = kayıtno + 1
ad.SetFocus 'imlecin ad dan başlaması için kullanılır.
End If
topkayit.Caption = LOF(1) / Len(alanlar)
suankayit.Caption = kayıtno
```

### End Sub

```
Private Sub onceki_Click()
If kayıtno= 1 Then
    MsgBox ("Zaten İlk Kayıttasınız")
```

**Else**

```
Kayıtno=kayıtno-1
```

### Get #1, kayıtno, alanlar

```
form1.ad = alanlar.ad
form1.soyad = alanlar.soyad
form1.telefon = alanlar.telefon
form1.adres = alanlar.adres
form1.sehir = alanlar.sehir
```

```

End if
Form1.ad.SetFocus
End Sub

Private Sub sonraki_Click()
Dim a As Integer
a=LOF(1) / Len (alanlar)
If kayitno >= a Then
    MsgBox ("Zaten Son Kayıttasınız")
Else
Kayitno=kayitno+1
Get #1, kayitno, alanlar
    form1.ad = alanlar.ad
    form1.soyad = alanlar.soyad
    form1.telefon = alanlar.telefon
    form1.adres = alanlar.adres
    form1.sehir = alanlar.sehir
End if
Form1.ad.SetFocus
End Sub
Private Sub bul_Click()
Dim bul As Integer
bul = InputBox("Bulmak istediğiniz Kayıt No yu giriniz")
kayitno = LOF(1) / Len(alanlar)
If bul <= kayitno Then
kayitno = bul
    Get #1, kayitno, alanlar 'dosya ya çağırdığım bilgileri ekrana yazar
    form1.ad = alanlar.ad
    form1.soyad = alanlar.soyad
    form1.telefon = alanlar.telefon
    form1.adres = alanlar.adres
    form1.sehir = alanlar.sehir
Else
MsgBox ("Böyle bir kayıt yok")
End If
End Sub

Private Sub kayitekle_Click()
kayitno = LOF(1) / Len(alanlar)
kayitno = kayitno + 1
    form1.ad = " "
    form1.soyad = " "
    form1.telefon = " "
    form1.adres = " "
    form1.sehir = " "
    form1.ad.SetFocus
End Sub
Private Sub iptal_Click()
Dim c
c = MsgBox("İptal etmek istiyormusunuz?", vbYesNo, "İptal Penceresi")
If c = vbYes Then
    form1.ad = " "
    form1.soyad = " "
    form1.telefon = " "
    form1.adres = " "
    form1.sehir = " "
End If
ad.SetFocus
End Sub

Private Sub degistir_Click()
Dim c As Integer

If form1.ad <> alanlar.ad Or form1.soyad <> alanlar.soyad Or form1.telefon <> alanlar.telefon Or
form1.adres <> alanlar.adres Or form1.sehir <> alanlar.sehir Then

```

```

c = MsgBox("Değişiklikler kaydedilsin mi ?", vbYesNo, "Değişiklik Penceresi")
If c = vbYes Then

    alanlar.ad = form1.ad
    alanlar.soyad = form1.soyad
    alanlar.telefon = form1.telefon
    alanlar.adres = form1.adres
    alanlar.sehir = form1.sehir
    Put #1, kayitno, alanlar

End If
End If

ad.SetFocus
End Sub

```

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

```

Dim silsayi, i, c

    silsayi = LOF(1) / Len(alanlar)
    For i = kayitno To silsayi - 1
        Get #1, i + 1, alanlar
        Put #1, i, alanlar
    Next i

    Open "oguz.tmp" For Random As #2 Len = Len(alanlar)
    For i = 1 To silsayi - 1
        Get #1, i, alanlar
        Put #2, i, alanlar
    Next i
    Close #1
    Close #2
FileCopy "oguz.tmp", "oguz.dat"
Kill "oguz.tmp"
c = MsgBox("Dosya silindi.")
Unload form1
Load form1
form1.Show

End Sub

```

## **Fonksiyonlar**

Fonksiyonlar belli işlemleri yaparlar (örneğin bir sayının karakökünü almak yada verilen bir tarih bilgisini yıl cinsine çevirmek gibi) ve bir değeri döndürürler. Bu nedenle genellikle bir değere eşitlenerek kullanılır.

### **Abs Fonksiyonu**

Bir sayının mutlak değerini bulmak için kullanılır.

#### **Örnek**

```

Private Sub Command1_Click()
Dim a As Double
Dim b As Double
    a = InputBox("Bir sayı girin")
    b = Abs(a)
Print "Girdiğiniz sayının mutlak değeri = "; b
End Sub

```

## **Asc Fonksiyonu**

Herhangi bir string bilginin ilk karakterine karşılık gelen ASCII kodunun değerini verir.

### **Örnek**

```
Private Sub Command1_Click()  
    Dim say  
    say = Asc("A")  
    Print say  
    say = Asc("a")  
    Print say  
    say = Asc("oguz")  
    Print say  
End Sub
```

## **Chr Fonksiyonu**

Asc fonksiyonunun tersidir yani ASCII kodunun karakter olarak karşılığını verir.

### **Örnek**

```
Private Sub Command1_Click()  
    Dim say  
    say = InputBox("Lütfen bir rakam giriniz")  
    Print Chr(say)  
End Sub
```

## **Date Fonksiyonu**

Sistem tarihini gösterir.

### **Örnek**

```
Private Sub Command1_Click()  
    MsgBox "Bu Günün Tarihi :" & Date, vbOKOnly, "Tarih Bilgisi"  
End Sub
```

## **DateDiff Fonksiyonu**

İki tarih arasındaki sürede geçen zamanı belirtir.

### **Örnek**

```
Private Sub Command1_Click()  
    Dim tarih As Date  
    tarih = InputBox("Doğum tarihinizi giriniz")  
    MsgBox ("siz" & DateDiff("d", Now, tarih) & "gün önce doğdunuz")  
End Sub
```

## **Int Fonksiyonu**

Girilen sayıyı yuvarlayarak yazılmasını sağlar.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim a, b, c  
    a = InputBox("bir sayı giriniz")  
    b = InputBox("bir rakam giriniz")  
    c = Int(a / b)  
    Print c  
    c = a / b  
    Print c
```

End Sub

### **LCase Fonksiyonu**

Yazdığınız bir yazının bütün harflerini küçük harfe çevirir.

#### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim a, b  
    a = "OGUZ KUTLU"  
    b = LCase(a)  
    Print b  
End Sub
```

#### **Örnek 2**

```
Private Sub Command1_Click()  
    Dim a, b  
    a = "OGUZ"  
    b = InputBox("şifreyi giriniz")  
    If b = a Then  
        Print "şifreyi bildiniz"  
    Else  
        Print "şifreyi bilemediniz"  
    End If  
End Sub
```

#### **Örnek 2.1**

```
Private Sub Command1_Click()  
    Dim a, b  
    a = "OGUZ"  
    b = InputBox("şifreyi giriniz")  
    If LCase(b) = LCase(a) Then  
        Print "şifreyi bildiniz"  
    Else  
        Print "şifreyi bilemediniz"  
    End If  
End Sub
```

### **Left Fonksiyonu**

Soldan istediğimiz kadar karakteri bulmamızı sağlar.

#### **Örnek 1**

```
Private Sub Command1_Click()  
    Cls  
    Dim ad, b  
    ad = InputBox("adınızı giriniz")  
    b = Left(ad, 4)  
    Print b  
End Sub
```

#### **Örnek 2**

```
Private Sub Command2_Click()  
    Cls  
    Dim ad, b, c  
    ad = InputBox("adınızı giriniz")  
    b = InputBox("kaç karakter")  
    c = Left(ad, b)  
    Print c  
End Sub
```

### **Örnek 3**

```
Private Sub Command1_Click()  
    Dim a, b  
    Cls  
    b = InputBox("adınızı giriniz")  
    For a = 1 To Len(b)  
        Print Left(b, a)  
    Next  
End Sub
```

### **Len Fonksiyonu**

Bir bilginin karakter olarak uzunluğunu veya değişken tipinin kapladığı byte alanını verir.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim ad, uz  
    ad = "Oguz KUTLU"  
    uz = Len(ad)  
    Print uz  
End Sub      'SONUÇ=10 olur.
```

### **Örnek 2**

```
Private Sub Command1_Click()  
    Dim ad  
    Dim a As Integer  
    Dim b As Long  
    a = 123  
    b = 1234567890  
    ad = "Oguz KUTLU"  
    Print "Tamsayı"; Len(a)  
    Print "Uzun tamsayı"; Len(b)  
    Print "Karakter"; Len(ad)  
End Sub
```

### **Örnek 3**

```
Private Sub Command1_Click()  
    Dim ad  
    Dim a As Integer  
    Dim b As Long  
    a = InputBox("birinci sayıyı giriniz")  
    b = InputBox("ikinci sayıyı giriniz")  
    ad = InputBox("adınızı giriniz")  
    Print "tamsayı"; Len(a)  
    Print "uzun tamsayı"; Len(b)  
    Print "sozluk" & Len(ad); "karakterdir"  
End Sub
```

### **Ltrim, Rtrim ve Trim Fonksiyonu**

Bir metnin içindeki boşlukları çıkartarak bitişik metin oluşturur. LTrim bilginin sol tarafındaki boşlukları atar. Rtrim bilginin sağ tarafındaki boşlukları atar. Trim bilginin sol ve sağ tarafındaki boşlukları atar.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim ad, a, b, c  
    ad = " Oguz Kutlu "  
    a = LTrim(ad)  
    b = RTrim(ad)
```

```
c = Trim(ad)
Print a
Print b
Print c
End Sub
```

### **Mid Fonksiyonu**

Bir metnin içindeki istenilen kısımdan kelime almak için kullanılır.

**Mid(değişken,başlangıç,uzunluk)**

#### **Örnek 1**

```
Private Sub Command1_Click()
    Dim ad
    ad = "Oguz Kemalettin Kutlu"
    FontName = "tahoma"
    FontSize = 15
    Print Mid(ad, 6)
    Print Mid(ad, 1, 23)
    Print Mid(ad, 17)
    Print Mid(ad, 1, 9)
End Sub
```

#### **Örnek 2**

```
Private Sub Command1_Click()
    Dim ad As String
    Dim a As Integer
    Dim b As Integer
    ad = "Oguz Kemalettin Kutlu"
    FontName = "tahoma"
    FontSize = 15
    a = InputBox("başlangıç değeri")
    b = InputBox("uzunluğu giriniz")
    Print Mid(ad, a, b)
End Sub
```

### **Now Fonksiyonu**

Sistem tarih ve zamanını verir.

#### **Örnek 1**

```
Private Sub Command1_Click()
    MsgBox "Bu Günün Bilgisi :" & Now, vbOKOnly, "Tarih Bilgisi"
End Sub
```

### **Round Fonksiyonu**

Belli sayıda ondalık basamağa sahip yuvarlanmış sayı oluşturur.

**Round (değişken,ondalık kısım)**

```
Sayı = 123,456789
Yenisayı=Round(sayı,3) ` Sonuç = 123,456
```

### **Space Fonksiyonu**

İstenilen sayıda boşluk oluşturur.

```
Text2.Text = "Oğuz" & Space(3) & "KUTLU"
```

## **Shell Fonksiyonu**

Hazırladığımız program içerisinde başka bir programı çalıştırmak için kullanılır.

### **Shell(yol[windowtipi])**

VbHide	0	Pencere gizli olarak aktifleşir.
vbNormalFocus	1	Pencere normal boyutlarında açılır.
vbMinimizedFocus	2	Pencere simge olarak açılır.
vbMaximizedFocus	3	Pencere tüm ekranı kaplar.
vbNormalNoFocus	4	Pencere geri yüklenir. Focus başka komponentte
vbMinimizedNoFocus	6	Pencere simge olarak yüklenir. Focus başka komponentte.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim a  
    a = Shell("c:\program files\microsoft office\excel",3)  
End Sub
```

## **Sqr Fonksiyonu**

Bir sayının karakökünü alır.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim a As Double  
    Dim b As Double  
    Dim c As Double  
    Dim d As Double  
    c = InputBox("bir sayı giriniz")  
    If c < 0 Then  
        MsgBox ("yanlış sayı girdiniz")  
    Else  
        d = Sqr(c)  
        Print d  
    End If  
End Sub
```

## **String Fonksiyonu**

İstenilen karakterin ilk harfini yanyana istediğimiz adtte yazmamızı sağlar.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim a  
    Dim b  
    b = InputBox("bir karakter giriniz.")  
    FontSize = 10  
    Print String(10, Chr(65))  
    Print String(20, "_")  
    Print String(5, b)  
End Sub
```

## **Time Fonksiyonu**

Sistem zamanını gösterir.

### **Örnek 1**

```
Private Sub Command1_Click()
```

```
Dim zaman
MsgBox ("sistem tarihi :") & Time
End Sub
```

### **UCase Fonksiyonu**

Bir ifadeyi büyük harfe çevirir.

#### **Örnek 1**

```
Private Sub Command1_Click()
Dim a, b
a = "oguz kutlu"
b = UCase(a)
Print b
End Sub
```

#### **Örnek 2**

```
Private Sub Command1_Click()
Dim a, b
a = "oguz"
b = InputBox("şifreyi giriniz")
If b = a Then
Print "şifreyi bildiniz"
Else
Print "şifreyi bilemediniz"
End If
End Sub
```

#### **Örnek 2.1**

```
Private Sub Command1_Click()
Dim a, b
a = "oguz"
b = InputBox("şifreyi giriniz")
If UCase(b) = UCase(a) Then
Print "şifreyi bildiniz"
Else
Print "şifreyi bilemediniz"
End If
End Sub
```

### **Val Fonksiyonu**

Bir ifadenin içindeki ilk rakam veya rakamları alır.

#### **Örnek 1**

```
Private Sub Command1_Click()
Dim a
Dim b As Single
a = InputBox("bir sayı giriniz")
b = Val(a)
Print b
End Sub
```

### **Dizi Değişkenler**

Aynı tür bilgileri aynı anda bir değişkene atamak için kullanılan yöntemlerdir. Diziler birçok değişkene aynı adla ulaşmayı sağlayan bir grup veri yapısıdır. Dizilerin eleman sayısı 0 dan başlar. Eğer değişken numarasını 1 den başlamasını istiyorsak ilk satıra **Option Base 1** yazılır. Dizi değişkenlerin üzerinde çalışılan form veya modülün **declarations** kısmında tanımlanması gerekir.

### **Örnek 1**

```
Private Sub Form_Load()  
    Dim ad(2) As String  
    Dim no(2) As Integer  
    ad(0) = "oguz"  
    ad(1) = "ahmet"  
    ad(2) = "mehmet"  
    no(0) = 25  
    no(1) = 75  
    no(2) = 85  
End Sub
```

### **Örnek 2**

Bir sınıftaki öğrencilerin notlarını toplayıp ortalamalarını almak.

General\_Declarations

```
Option Explicit  
Option Base 1  
Dim ogrenci(30) As Integer
```

```
Private Sub Form_Load()  
    Dim sayi As Integer  
    Dim a, toplam, ortalama  
    sayi = InputBox("Öğrenci sayısını giriniz")  
    For a = 1 To sayi  
        ogrenci(a) = InputBox(Str(a) + ".Öğrencinin notunu girin")  
        toplam = toplam + ogrenci(a)  
    Next a  
    FontSize = 15  
    Show  
    ortalama = toplam / sayi  
    Print "Not Ortalaması .:"; ortalama  
End Sub
```

### **Örnek 2**

```
Option Explicit  
Option Base 1  
Dim a(3) As Integer
```

```
Private Sub Command1_Click()  
    Dim hesap  
    a(1) = Text1.Text  
    a(2) = Text1.Text  
    a(3) = Text1.Text  
    hesap = a(1) + a(2) + a(3)  
    Print hesap  
End Sub
```

### **Dinamik Diziler**

Bu tip dizilerde değişkene atanacak dizi sayısında bir sınırlama yoktur.

### **Örnek 1**

```
Option Explicit  
Option Base 1  
  
Private Sub Form_Load()  
    Dim ogrenci() As Integer  
    Dim sayi As Integer
```

```

Dim a, toplam, ortalama
sayi = InputBox("Öğrenci sayısını giriniz")
ReDim ogrenci(sayi) As Integer
For a = 1 To sayi
ogrenci(a) = InputBox(Str(a) + ".Öğrencinin notunu girin")
toplam = toplam + ogrenci(a)
Next a
FontSize = 15
Show
ortalama = toplam / sayi
Print "Not Ortalaması .:"; ortalama
End Sub

```

## Function Oluşturma

Bir procedure olarak düzenlenen fonksiyonlar belli bir işlemi kod olarak içeren ve bir değeri geri döndüren program parçalarıdır. Örneğin bir çok yerde faktöriyel hesaplamamız gerekmektedir, her seferinde faktöriyel formülünü yazmaktansa bu işlemi bir fonksiyonda tanımlayıp sonra bu fonksiyonu çağırırız. Fonksiyonlar **Function** deyimi ile başlayıp **End Function** deyimi ile biter. Bir function oluşturmak için code penceresi açıkken Tools menüsünden Add Procedure seçilir.

### Örnek 1

Klavyeden girilen bir sayının faktöriyelini bulan fonksiyon. Önce fak bir function tanımlanır.

```

Public Function fak(x)
Dim a, b
b = 1
For a = 1 To x
b = b * a
Next
fak = b
End Function

Private Sub Command1_Click()
Dim a
a = InputBox("Lütfen bir sayı giriniz")
Print a & "! =" & fak(a)
End Sub

```

### Örnek 2

Klavyeden girilen bir sayının faktöriyelini bulan ve tüm sayıları ekrana yazan fonksiyon. Önce fak bir function tanımlanır.

```

Public Function fak(x)
Dim a, b
b = 1
For a = 1 To x
b = b * a
Next
fak = b
End Function

Private Sub Command1_Click()
Dim a, b
a = InputBox("Lütfen bir sayı giriniz")
For b = 1 To a
Print b & "! =" & fak(b)
Next
End Sub

```

### **Örnek 3**

Klavyeden girilen bir sayının fahrenheit cinsinden değerini veren fonksiyon.

```
Public Function fah(x)
    fah = x * 9 / 5 + 32
End Function

Private Sub Command1_Click()
    Cls
    Dim a
    a = InputBox("Lütfen bir sayı giriniz")
    MsgBox a & " °C Fahrenheit cinsinden karşılığı =" & fah(a)
End Sub
```

### **Örnek 4**

Klavyeden girilen bir sayının karesini ve karakökünü bulan fonksiyon.

```
Public Function kare(x)
    kare = x ^ 2
End Function

Public Function karekok(x)
    karekok = Sqr(x)
End Function

Private Sub Command1_Click()
    Cls
    Dim a
    a = InputBox("Lütfen bir sayı giriniz")
    Print a & " sayısının karesi =" & kare(a)
    Print a & " sayısının karakökü =" & karekok(a)
End Sub
```

### **Örnek 4**

bir üçgenin hipotenüsünü bulan function.

```
Public Function hipo(x, y)
    hipo = Sqr(x ^ 2 + y ^ 2)
End Function

Private Sub Command1_Click()
    Cls
    Dim a, b
    a = InputBox("Lütfen birinci kenarı giriniz")
    b = InputBox("Lütfen ikinci kenarı giriniz")
    Print hipo(a, b)
End Sub
```

## **Componentler (Kontrol Elemanları)**

### **Textbox (Metin Kutusu)**

Bilgi girişi için kullanılır. Programlarda kullanılan adı, soyadı, telefonu gibi alanlardır. Metin kutularına girilen bilgiler daha sonra başka amaçlar için kullanılabilirler. Genel bir alışkanlık olarak textbox ların önüne bir label nesnesi konularak kullanıcı bilgilendirilir.

### **Özellikleri :**

#### **Name**

Yordam içerisinde text kutusunun adını değiştirmek için kullanılır.

### **Alignment**

Textbox içerisindeki yazının sağa, sola veya ortaya yazılmasını sağlar.

0 = Sola  
1 = Sağa  
2 = Ortaya

```
Private Sub Command1_Click()  
    Text1.Alignment = 0  
End Sub
```

### **Appearance**

Text kutusunun üç boyutlu olarak görünümünü sağlar.

0 = Düz  
1 = Üç boyutlu

```
Private Sub Command1_Click()  
    Text1.Appearance = 0  
End Sub
```

### **BackColor**

Text kutusunun zemin rengini değiştirir. Properties penceresinden değiştirebildiğimiz gibi aşağıdaki kodlarla da değiştirebiliriz.

```
Private Sub Command1_Click()  
    Text1.BackColor = QBColor(4)  
End Sub
```

```
Private Sub Command1_Click()  
    Text1.BackColor = RGB(120, 25, 30)  
End Sub
```

### **BorderStyle**

Text kutusunun çerçeve özelliğini belirler.

0 = Çerçeve yok  
1 = Çerçeve var

### **CausesValidation**

Text kutusundan çıkarken bir şartın gerçekleşip gerçekleşmeyeceğini belirtmeye yarar. Default olarak true dir.

### **Örnek 1**

```
Private Sub Form_Load()  
    Text1 = ""  
    Text2 = ""  
End Sub
```

```
Private Sub Text1_Change()  
    Print Text1  
End Sub
```

```
Private Sub Text1_Validate(Cancel As Boolean)  
    If 100 < Val(Text1) < 0 Then  
        MsgBox ("0 - 100 arasında bir sayı giriniz")  
        Cancel = True ` Burada Cancel = False olursa kodu işletir ama diğer yordama geçer.  
    End If  
End Sub
```

## **Örnek 2**

```
Private Sub Text2_Validate(Cancel As Boolean)
    Dim a
    a = Val(Text2.Text)
    If Text2.Text <> a Then
        MsgBox ("Lütfen sadece rakam giriniz")
        Cancel = True
    End If
End Sub
```

## **Enabled**

Text kutusunun aktif veya pasif olmasını sağlar.

```
Private Sub Command1_Click()
    Text1.Enabled = True ` False olursa kullanıcı giriş yapamaz.
End Sub
```

## **Font**

Text kutusunun yazı özelliklerini değiştirir.

```
Private Sub Command1_Click()
    Text1.FontBold = True
    Text1.FontItalic = True
    Text1.FontName = "tahoma"
    Text1.FontSize = 15
    Text1.FontStrikethru = True
    Text1.FontUnderline = True
    Text1.ForeColor = QBColor(4)
End Sub
```

Program esnasında bütün textlerin özelliklerini bir anda değiştirmemizde mümkündür. Bunun için öncelikle bir text çizilir ve sonra bu text copy – paste yapılır. Çıkan soruya YES denir. Bu işlemin sonucunda aynı ad ve özelliklere sahip text kutuları oluşmuş olur. Onları birbirinden ayıran indis numaralarıdır.

```
Private Sub Command1_Click()
    Dim a
    For a = 0 To 5
        Text1(a).FontBold = True
        Text1(a).FontItalic = True
        Text1(a).FontName = "tahoma"
        Text1(a).FontSize = 15
    Next
End Sub
```

## **Height**

Text kutusunun yüksekliğini ayarlar.

```
Private Sub Command1_Click()
    Text1.Height = 950
End Sub
```

## **Index**

Aynı isimli birden fazla nesne varsa VB bunlara bir index numarası vererek ayırır.

## **Left**

Text kutusunun soldan uzaklığını ayarlar.

## **Locked**

Text kutusunu veri girilemeyecek şekilde kitler.

### **MaxLength**

Text kutusuna girilebilecek maksimum karakter sayısını belirler.

```
Private Sub Command1_Click()  
    Text1.MaxLength = 5  
End Sub
```

### **MousePointer, MouseIcon**

Text kutusunun üzerine mouse la gelindiği zaman mousenin alacağı şekli gösterir. 0 – 15 arasında bir değer verilebilir. 99 verirsek bizim MouseIcon da seçtiğimiz icon görüntülenir.

```
Private Sub Command1_Click()  
Static a  
    Text1.Text = Str(a) + "mouse şekilleri"  
    Text1.MousePointer = a  
    a = a + 1  
    If a = 15 Then End  
End Sub
```

### **MultiLine**

Text kutusunun için birden fazla satır açmak için kullanılır.

### **PasswordChar**

Text kutusunu şifre girmek için kullanılacak; kullanıcının girdiği karakterleri belli etmemek için kullanılır. Multiline özelliği = True olunca çalışmıyor.

```
Private Sub Form_Load()  
    Text1.Text = ""  
    Text1.PasswordChar = "o"  
End Sub
```

### **SelStart, SelLength, SelText**

Text kutusunun içinde seçme yapmak için kullanılır.

#### **Örnek 1**

```
Private Sub Command1_Click()  
    Text1.SelStart = 0  
    Text1.SelLength = 6  
    Text2.Text = Text1.SelText  
End Sub
```

Text1.SelLength = Len(Text1) 'yazarsak text1 in tamamını kopyalamış oluruz.

#### **Örnek 2**

```
Private Sub Command1_Click()  
If Text1.SelLength > 0 Then 'seçili bir kısım varsa  
    Text1.SelText = UCase(Text1.SelText) 'Seçili kısmı büyük harfe çevir.  
Else  
    Text1.Text = UCase(Text1.Text)  
End If  
End Sub
```

```
Private Sub Command2_Click()  
If Text1.SelLength > 0 Then 'seçili bir kısım varsa  
    Text1.SelText = LCase(Text1.SelText) 'Seçili kısmı küçük harfe çevir.  
Else  
    Text1.Text = LCase(Text1.Text)  
End If
```

End Sub

### **ScrollBars**

Text kutusunun MultiLine özelliği True olması durumunda etkin olur ve text kutusuna kaydırma çubuklarını ekler.

### **TabIndex**

VB form üzerinde kullanıcının ulaşabildiği her nesneye bir tabindeksi verir. Bu indeks numarası yardımıyla tab tuşuna basıldığı zaman kontroller arasında geçiş sağlanır.

### **TabStop**

Bilgi girişinin çok olduğu formlarda bazı kutulara tabstop koyarak tab tuşuyla bu kontrole ulaşmayı engelleyebiliriz.

### **Text**

Text kutusuna girilen metne ulaşmak için veya bir metin atamak için kullanılır.

```
Private Sub Command1_Click()  
    Text1.Text = "prestige"  
End Sub
```

```
Private Sub Form_Load()  
    Dim a, b, c, d  
    a = InputBox("adınızı giriniz")  
    b = InputBox("soy adınızı giriniz")  
    c = InputBox("telefonunuzu giriniz")  
    d = InputBox("adresinizi giriniz")  
    Text1.Text = a  
    Text2.Text = b  
    Text3.Text = c  
    Text4.Text = d  
End Sub
```

### **ToolTipText**

Program içerisinde text kutusunun üzerine gelindiği zaman açılan bir bilgi penceresi oluşturur.

```
Private Sub Form_Load()  
    Text1.ToolTipText = "Adı Bilgisi"  
    Text2.ToolTipText = "Soyadı Bilgisi"  
    Text3.ToolTipText = "Telefon Bilgisi"  
    Text4.ToolTipText = "Adres Bilgisi"  
End Sub
```

### **Top**

Text kutusunun yukarıdan olan uzaklığını ayarlar.

### **Visible**

Text kutusunun ekranda görünmesi sağlar.

### **Width**

Text kutusunun genişliğini ayarlamak için kullanılır.

### **Text\_Change**

Text kutusunun içerdiği metin değiştiği anda bir olayın olmasını istiyorsak bu yordama yazarız.

### **Örnek 1**

```
Private Sub Text1_Change()  
    Print Text1  
End Sub
```

## **Örnek 2**

Rakamlar arasına virgöl koyma.

```
Private Sub Text1_Change()  
Dim a  
a = Text1.SelStart  
Text1 = Format(Text1, "###,###")  
Text1.SelStart = a + 1  
End Sub
```

## **Text Click, DbClick**

Text kutusuna fare ile birkez tıkladığı zaman bir olayın olmasını istiyorsak bu yordama yazarız.

## **Örnek 1**

```
Private Sub Text1_Click()  
Text1.BackColor = QBColor(Rnd * 15)  
End Sub
```

## **Örnek 2**

```
Private Sub Text1_DbClick()  
Text1.BackColor = QBColor(Rnd * 15)  
End Sub
```

## **Text GotFocus**

Text kutusuna geldiği anda meydana gelmesini istediğimiz bir olayın kodunu bu yordama yazarız.

## **Örnek 1**

```
Private Sub Text1_GotFocus()  
Text1.SelStart = 0  
Text1.SelLength = Len(Text1)  
End Sub
```

## **Text LostFocus**

Text kutusundan çıktığı anda meydana gelmesini istediğimiz bir olayın kodunu bu yordama yazarız.

## **Örnek 1**

```
Private Sub Text1_LostFocus()  
Text1.Text = UCase(Text1)  
End Sub
```

## **Örnek 2**

```
Private Sub Text1_LostFocus()  
If 100 < Val(Text1) < 0 Then  
MsgBox ("0 - 100 arasında bir sayı giriniz")  
Cancel = True ' Burada Cancel = False olursa kodu işletir ama diğer yordama geçer.  
Text1.SetFocus  
Text1.SelStart = 0  
Text1.SelLength = Len(Text1)  
End If  
End Sub
```

## **Text KeyPress, KeyUp, KeyDown**

KeyDown : Klavyeden bir tuşa basıldığı zaman istediğimiz bir olayın kodunu bu yordama yazarız.  
KeyUp : Klavyeden bir tuş bırakıldığı zaman istediğimiz bir olayın kodunu bu yordama yazarız.  
KeyPress : KeyDown ile KeyUp arasında oluşan olayları kontrol eder.

### **Örnek 1**

Ctrl tuşlarıyla beraber ok tuşları kullanılınc text kutusu yön değiştirecek, alt tuşuyla beraber basılınc boyutları değişecek.

```
Private Sub Text1_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
If Shift And vbCtrlMask Then
```

```
    Select Case KeyCode
```

```
        Case vbKeyLeft: Text1.Left = Text1.Left - 25
```

```
        Case vbKeyRight: Text1.Left = Text1.Left + 25
```

```
        Case vbKeyUp: Text1.Top = Text1.Top - 25
```

```
        Case vbKeyDown: Text1.Top = Text1.Top + 25
```

```
    End Select
```

```
End If
```

```
If Shift And vbAltMask Then
```

```
    Select Case KeyCode
```

```
        Case vbKeyLeft: Text1.Width = Text1.Width - 25
```

```
        Case vbKeyRight: Text1.Width = Text1.Width + 25
```

```
        Case vbKeyUp: Text1.Height = Text1.Height - 25
```

```
        Case vbKeyDown: Text1.Height = Text1.Height + 25
```

```
    End Select
```

```
End If
```

### **Örnek 2**

```
Private Sub Text1_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
If Shift And vbCtrlMask Then
```

```
    Select Case KeyCode
```

```
        Case vbKeyLeft:
```

```
            Text1.Left = Text1.Left - 80
```

```
        Case vbKeyRight:
```

```
            Text1.Left = Text1.Left + 80
```

```
        Case vbKeyUp:
```

```
            Text1.Top = Text1.Top - 80
```

```
        Case vbKeyDown:
```

```
            Text1.Top = Text1.Top + 80
```

```
    End Select
```

```
End If
```

```
End Sub
```

### **Örnek 3**

Klavyeden yazı yazarken büyük harfe çevirir.

```
Private Sub Text1_KeyPress(KeyAscii As Integer)
```

```
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
```

```
End Sub
```

### **MouseUp, MouseDown**

MouseDown : Text kutusu üzerindeyken mousenin bir tuşuna basıldığı zaman istediğimiz bir olayın kodunu bu yordama yazarız.

MouseUp : Mousenin bir tuşundan parmağımızı kaldırdığımız zaman istediğimiz bir olayın kodunu bu yordama yazarız.

### **Örnek 1**

```
Private Sub Text1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
Dim a
```

```
    Text1.BackColor = QBColor(4)
```

```
End Sub
```

```
Private Sub Text1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
Dim a
```

```
    Text1.BackColor = QBColor(0)
```

```
End Sub
```

### **MouseMove**

MouseMove : Text kutusu üzerindeyken mouse ile dolaşırken istediğimiz bir olayın kodunu bu yordama yazarız.

#### **Örnek 1**

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Text1 = "Visual Basic"
End Sub

Private Sub Text1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Text1.Text = UCase(Text1)
End Sub
```

#### **Örnek 2**

Text kutusu üzerinde gezinirken mousenin koordinatlarını verir.

```
Private Sub Text1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Text1.Text = "fare şu anda " & " dikey :" & Y & " yatay :" & X & "koordinatlarında"
End Sub
```

#### **Örnek 3**

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Static a, b
    a = Int(Rnd * 40) + 1
    b = Int(Rnd * 16000) + 1
    DrawWidth = a
    PSet (X, Y), b
End Sub
```

#### **Örnek 4**

Dim a, b

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    a = X
    b = Y
    PSet (X, Y)
End Sub
```

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = vbRightButton Then Circle (a, b), Abs(a - X) ' sağ tuşa basılıyken çember çiz
    If Button = vbLeftButton Then Line (a, b)-(X, Y), , B 'sol tuşa basılıyken kare çiz
End Sub
```

### **Label (Etiket)**

Form üzerinde bir bilginin gösterilmesi için veya textbox ın açıklaması olarak kullanılır.

#### **Özellikleri :**

##### **AutoSize**

Label içindeki bilginin uzunluğuna göre otomatik olarak boyutlarının ayarlanmasını sağlar.

##### **BackStyle**

Label transparent veya opak olmasını sağlar.

### **Caption**

Label üzerindeki yazıyı buradan ayarlarız.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Label1.Caption = "Adı Soyadı"  
End Sub
```

### **Command Button (Komut Düğmesi)**

Bir olayın kullanıcı tarafından başlatılması için kullanılır.

### **Özellikleri :**

#### **Cancel**

Klavyeden ESC tuşuna basıldığı zaman command buttonun aktif olması durumudur.

```
Private Sub Command1_Click()  
    Command1.Caption = "Esc tuşuna basıldı"  
End Sub
```

```
Private Sub Form_Load()  
    Command1.Cancel = True  
End Sub
```

#### **Caption**

Command Buttonun adını belirtir. Yazdığınız kelimelerin arasına & işareti konursa kısayol tuşu görevi görür.

```
Private Sub Form_Load()  
    Command1.Caption = "o&guz"  
    Command2.Caption = "og&uz"  
End Sub
```

#### **Picture, DisabledPicture, DownPicture**

Style özelliği 1 verilerek Command Buttonun resimli bir komut düğmesi olarak belirlendikten sonra.

Picture : Komut düğmesi aktif durumdayken

DisabledPicture : Komut düğmesi pasif durumdayken

DownPicture : Komut düğmesi basılı durumdayken

#### **Enabled**

Command Buttonun aktif olup olmamasının sağlar.

```
Private Sub Text1_Change()  
    If Text1.Text <> "" And Text2.Text <> "" Then  
        Command1.Enabled = True  
    End If  
End Sub
```

```
Private Sub Text2_Change()  
    If Text1.Text <> "" And Text2.Text <> "" Then  
        Command1.Enabled = True  
    End If  
End Sub
```

## **MouseMove**

```
Private Sub Command1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
Dim a As Integer, b As Integer
a = Int(Rnd * 5000)
b = Int(Rnd * 5000)
If a > 7000 Then a = 3500
If b > 7000 Then b = 2700
Command1.Left = a
Command1.Top = b
End Sub
```

## **OYUN**

Sayı bulma oyunu, 4 Label ,1 TextBox, 2 Commond Buton

```
Private Sub Command1_Click()
Command2.Enabled = True
Command1.Enabled = False
Label1 = Int(Rnd * 100)
Label4 = 1000
Label3 = 0
Label2 = "0 ile 100 arasında bir sayı girin"
Text1.SetFocus
End Sub
```

```
Private Sub Command2_Click()
Label3 = Label3 + 1
Label4 = Label4 - Label3 * 10

If Label1 < Text1 Then
Label2 = "Küçült"
Else
If Label1 > Text1 Then
Label2 = "Büyült"
Else
Label2 = "Tebrikler " + Label3 + " defada bildiniz " + Label4 + " puan aldınız "
Command1.Enabled = True
Command2.Enabled = False
End If
End If
Text1 = ""
Text1.SetFocus
End Sub
```

```
Private Sub Form_Load()
Command1.Default = True
Text1 = ""
Command2.Enabled = False
Label1.Visible = False
End Sub
```

## **PASSWORD**

```
Private Sub Command1_Click()
Text1 = "" : Text2 = ""
End Sub
```

```
Private Sub Command2_Click()
If Text1 = "oguz" And Text2 = "kutlu" Then
MsgBox ("Merhaba")
Exit Sub
End If
If Text1 <> "oguz" Then
Text1.SetFocus
```

```

Text1.SelStart = 0
Text1.SelLength = Len(Text1)
Exit Sub
End If
If Text2 <> "kutlu" Then
Text2.SetFocus
Text2.SelStart = 0
Text2.SelLength = Len(Text2)
Exit Sub
End If : End Sub
Private Sub Text2_Change()
If Text1 <> "" Then Command2.Enabled = True
If Text2 = "" Then Command2.Enabled = False
End Sub
Private Sub Text1_Change()
If Text2 <> "" Then Command2.Enabled = True
If Text1 = "" Then Command2.Enabled = False
End Sub

```

### **CheckBox (İşaret Kutusu)**

Belli seçeneklerin seçilmesini sağlar.

#### **Özellikleri :**

##### **Alignment**

Check işaretinin solda veya sağda olmasını sağlar.

##### **Value**

Check işaretinin özelliklerini belirler.

0 : İşaretsiz  
1 : İşaretli  
2 : Belirsiz

```

If Check1.Value = 0 Then Text1.Text = "İşaretsiz"
If Check1.Value = 1 Then Text1.Text = "İşaretli"
If Check1.Value = 2 Then Text1.Text = "Belirsiz"

```

#### **Örnek 1**

1 tane text ve 3 tane checkbox çiz ve style =1 yap sonra \common\graphics\bitmaps\tlbr\_w95 dizininden gerekli resimleri üzerine koy (bold, underline, italic)

```

Private Sub Check1_Click()
Text1.FontBold = Check1.Value
End Sub

```

```

Private Sub Check2_Click()
Text1.FontItalic = Check2.Value
End Sub

```

```

Private Sub Check3_Click()
Text1.FontUnderline = Check3.Value
End Sub

```

## **Option Button (Seenek Dügmesi)**

### **Örnek 1**

3 tane text ve 2 tane optionbox çizin,

```
Private Sub Form_Load()  
    Show  
    Text1.SetFocus  
    Option1 = False  
    Option2 = False  
End Sub  
  
Private Sub Option1_Click()  
    Text2.Text = Val(Text1.Text) * 0.17  
    Text3.Text = Val(Text1.Text) + Val(Text2.Text)  
    Text3 = Format(Text3, "###,###")  
End Sub  
  
Private Sub Option2_Click()  
    Text2.Text = Val(Text1.Text) * 0.25  
    Text3.Text = Val(Text1.Text) + Val(Text2.Text)  
    Text3 = Format(Text3, "###,###")  
End Sub
```

## **Horizontal & Vertical ScrollBar (Kaydırma Çubuğu)**

### **Value**

Kaydırma çubuğunun aldığı değeri gösterir. Bu değeri min. - max. Arasında olabilir.

### **Min. Max.**

Kaydırma çubuğunun alabileceği değerleri gösterir.

### **Large Change**

Kaydırma çubuğunun üzerine tıkladığı zaman scroll.value'nin uğrayacağı değışimi gösterir.

### **Small Change**

Kaydırma çubuğunun üzerine oklara tıkladığı zaman scroll.value'nin uğrayacağı değışimi gösterir.

### **Scroll**

Kaydırma çubuğunun üzerindeki scrollun değıştiği zaman bir olayın olmasını istiyorsak bu yordama yazarız.

### **Örnek 1**

1 textbox, 3 hscroll , 3 label çiz.

```
Private Sub Form_Load()  
    HScroll1.Max = 255  
    HScroll2.Max = 255  
    HScroll3.Max = 255  
End Sub  
  
Private Sub HScroll1_Change()  
    Text1.BackColor = RGB(HScroll1.Value, HScroll2.Value, HScroll3.Value)  
    Label1.Caption = HScroll1.Value  
End Sub  
  
Private Sub HScroll2_Change()  
    Text1.BackColor = RGB(HScroll1.Value, HScroll2.Value, HScroll3.Value)  
    Label2.Caption = HScroll2.Value  
End Sub
```

```
Private Sub HScroll3_Change()  
    Text1.BackColor = RGB(HScroll1.Value, HScroll2.Value, HScroll3.Value)  
    Label3.Caption = HScroll3.Value  
End Sub
```

## **Shape (Şekil Kontrol Elemanı)**

Formlara dikdörtgen, kare, daire gibi geometrik şekiller çizmemize yarar.

### **BorderStyle**

Çizilecek olan şeklin kenar çizgi özelliklerini belirtir.

0	Çizgi yok
1	Düz çizgi
2	Uzun kesikli çizgi
3	Nokta çizgi
4	Kesikli + noktalı çizgi
5	Kesikli + noktalı çizgi+ noktalı
6	İnside solid (şekil ile çerçeve kenarları çakışık şekilde)

### **BorderWidth**

Çizilecek olan şeklin kenar çizgi kalınlığını değiştirir.

### **FillColor**

Çizilecek olan şeklin iç rengini değiştirir.

### **FillStyle**

Çizilecek olan şeklin iç desenini belirtir.

0	Tam dolu
1	Nesnenin zemin rengi
2	Yatay çizgi
3	Dikey çizgi
4	Sola eğik
5	Sağa eğik
6	Kareli
7	Çapraz

### **Shape**

Çizilecek olan şeklin şeklini belirtir.

0	Dikdörtgen
1	Kare
2	Elips
3	Çember
4	Oval Dörtgen
5	Oval Kare

### **Örnek 1**

Scrool `ları kullanarak Shape özelliklerini gösteren program yap.

4 Vscroll, 1 Shape

## **Timer Control (Zamanlayıcı)**

### **Enabled**

True olursa zaman çalışmaya başlar. False olursa durur.

### **Interval**

Saatın çalışma periyodunu belirtir. 0 – 65535 arasında değer alır. 1000 olursa 1 saniye olur.

```
Private Sub Form_Load()  
    Label1.Caption = Now  
End Sub
```

```
Private Sub Timer1_Timer()  
    Label1.Caption = Now  
End Sub
```

### **Örnek 1**

#### **Saat**

```
Private Sub Form_Load()  
    Label1.Caption = Time  
End Sub
```

```
Private Sub Timer1_Timer()  
    Label1.Caption = Time  
End Sub
```

### **Örnek 2**

#### **Kronometre**

```
Option Explicit  
Dim saat, dakika, saniye As Integer
```

```
Private Sub basla_Click()  
    saat = 0  
    dakika = 0  
    saniye = 0  
End Sub
```

```
Private Sub durdur_Click()  
    Label2.Caption = saat & " : " & dakika & " : " & saniye  
End Sub
```

```
Private Sub Form_Load()  
    Timer1.Interval = 1000  
    Label1.Visible = False  
End Sub
```

```
Private Sub Timer1_Timer()  
    Label1.Caption = Time  
    saniye = saniye + 1  
    If saniye = 59 Then  
        dakika = dakika + 1  
        saniye = 0  
    End If  
    If dakika = 59 and saniye = 59 Then  
        saat = saat + 1  
        dakika = 0  
    End If  
End Sub
```

### **Örnek 3**

```
Private Sub Form_Load()  
Label1.Caption = Time  
Label2.Caption = Year(Now) & Month(Now) & Day(Now) & Hour(Now) & Minute(Now) & Second(Now)  
Timer2.Enabled = False  
End Sub
```

```
Private Sub Timer1_Timer()  
Label1.Caption = Time  
If Hour(Now) = 20 And Minute(Now) = 59 Then Timer2.Enabled = True  
End Sub
```

```
Private Sub Timer2_Timer()  
Label2.Caption = Second(Now)  
End Sub
```

### **Örnek 4**

#### **Ekran Koruyucu**

```
Private Sub Form_Load()  
Timer1.Interval = 1000  
End Sub  
Private Sub Timer1_Timer()  
Dim x, y, a, b  
a = Form1.ScaleWidth  
b = Form1.ScaleHeight  
x = Int(Rnd * a)  
y = Int(Rnd * b)  
Line -(x, y)  
DrawWidth = Int(Rnd * 20) + 1  
ForeColor = Int(Rnd * 16000000)  
End Sub
```

### **Örnek 5**

#### **Top oyunu**

```
Form_KeyPreview = True
```

```
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)  
If KeyCode = vbKeyRight Then Command1.Left = Command1.Left + 100  
If Command1.Left + Command1.Width >= Form1.ScaleWidth Then Command1.Left = Form1.ScaleWidth -  
Command1.Width  
If KeyCode = vbKeyLeft Then Command1.Left = Command1.Left - 100  
If Command1.Left <= 0 Then Command1.Left = 0  
End Sub
```

```
Private Sub Form_KeyPress(KeyAscii As Integer)  
If Timer1.Enabled = True Then Timer1.Interval = 1  
End Sub
```

```
Private Sub Form_Load()  
Command1.Left = (Form1.ScaleWidth / 2) - (Command1.Width / 2)  
Option1.Left = (Form1.ScaleWidth / 2) - (Option1.Width / 2)  
Option1.Top = Command1.Top - Option1.Height  
End Sub
```

```
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)  
If Command1.Left = Form1.ScaleWidth - Command1.Width Then  
Command1.Left = Form1.ScaleWidth - Command1.Width  
Else  
Command1.Left = X  
End If  
End Sub
```

```
Private Sub Timer1_Timer()
    Static sx, sy, X, Y
    Randomize Timer
```

If IsEmpty(sx) Then ' **değişkenin boş olup olmadığını kontrol eder ve boolean değer döndürür. Static olarak tanımladığımız (sx) programı ilk çalıştırdığımızda bir değeri olmadığı için isEmpty çalışacak ve sx=50 olacak, program geri döndüğünde sx bu sefer bir değere sahip olacağı için isEmpty çalışmayacak ve sx kendi değeri ile kalacak.**

```
        sx = 50
        sy = -50
        X = Option1.Left
        Y = Option1.Top
    End If
    If X <= 0 Then sx = -sx: Beep 'sol kenara gelince geri dön
    If X + Option1.Width > Form1.ScaleWidth Then sx = -sx: Beep 'sağ kenara gelince geri dön
    If Y < 0 Then sy = -sy: Beep 'üste gelince geri dön
    If (Y >= Command1.Top) And (Y < (Command1.Top + Command1.Height)) And X + Option1.Width >
Command1.Left And X < (Command1.Left + Command1.Width) Then sy = -sy 'command butona çarpınca geri dön
        If Y >= Form1.ScaleHeight Then 'top altta gelince oyunu bitir.
            MsgBox ("oyun bitti")
            Timer1.Interval = False
        End
    End If
    X = X + sx
    Y = Y + sy
    Option1.Left = X
    Option1.Top = Y
End Sub
```

### **Örnek 6**

Parlayan yazı : 2 label, 1 textbox, 1 timer, Timer1.Interval = 10

```
Private Sub Form_Load()
    Label2.Left = Label1.Left + Label2.Width
End Sub

Private Sub Text1_Change()
    If Len(Text1) = 0 Then Exit Sub
    Label1 = Left(Text1, Len(Text1) - 1)
    Label2 = Right(Text1, 1)
    Label1.FontSize = 15
    Label2.Left = Label1.Left + Label2.Width
End Sub

Private Sub Timer1_Timer()
    Static a, b
    If IsEmpty(a) Then
        a = Label1.FontSize
        b = 5
    End If
    If a > 50 Or a < 5 Then b = -b
        a = a + b
        Label2.FontSize = a 'İkinci örnek olarak Label1 yazılır.
End Sub
```

### **Frame Control (Çerçeve)**

İçerdiği kontrolleri gruplamak için kullanılır. Genellikle Option Button'larda kullanılır. Enabled veya Visible özelliği kullanılır.

## **UpDown(Yukarı aşağı arttırma)**

Textbox lara bilgi girişi yaparken kontrollü bir şekilde yapılmasını istiyorsak kullanırız. Bunun için Components menüsünden Microsoft Windows Common Controls-2 seçilerek ekrana getirilir.

### **Alignment**

BuddyControl ile herhangi bir kontrole bağlantı kurulmuşsa UpDown o nesnenin sağında mı solunda mı olacağını ayarlar.

### **AutoBuddy**

UpDown kontrolünün herhangi bir kontrole otomatik olarak atanmasını sağlar genellikle kullanılmaz.

### **BuddyControl**

UpDown kontrolünün hangi kontrole atanmasını sağlar.

### **BuddyProperty**

UpDown kontrolünün bağlandığı kontrolün hangi özelliğine bağlanacağını belirler.

### **Increment**

UpDown kontrolünün her tıklamada ne kadar değişime uğrayacağını belirtiriz.

### **Min - Max**

UpDown kontrolünün alt ve üst sınırlarını belirtiriz.

### **Orientation**

UpDown kontrolünün yatay veya dikey olarak yerleştirilmesini sağlar.

### **SyncBuddy**

UpDown kontrolünün herhangi bir kontrole atasanız bile bu özelliği False yaparak işlemi durdurabilirsiniz.

### **Value**

UpDown kontrolünün aldığı değer.

### **Örnek 1**

UpDown kullanarak bir text kutusunun fontsize ve forecolor özelliklerini, checkbox kullanarak textin Alignment, Bold, İtalic, Underline özelliklerini Vscroll kullanarak Backcolor özelliğini değiştirelim. Ekranda bir saat ve ToolTipte tarih olsun. Form büyütüldüğü zaman Text ona göre davransın.

## **Line (Çizgi)**

## **ComboBox (Açılan Liste)**

### **IntegralHeight**

Bu özellik True ise açılan kutuda en alttaki yazının tam olarak görünmesi için kendisini otomatik olarak boyutlar.

### **List**

Combo içine direk veri girişi için kullanılır.

### **ListIndex**

Combo içindeki yazan textlerden seçim yapmayı sağlar. İlk veri **0** son veri ise **ListCount – 1** dir.

## **Sorted**

Combo içindeki yazıların sıralı olmasını sağlar.

## **Style**

Combo tipini ve çalışma biçimini düzenler.

- 0 Aşağı doğru açılan ve içeriği değiştirilebilen combo.
- 1 Aşağı doğru açılmayan ama içeriği değiştirilebilen combo.
- 2 Aşağı doğru açılan ama veri girişi yapılamayan combo.

## **Text**

Combonun aktif olan texti.

```
Private Sub Combo1_Click()  
    Text1.Text = Combo1.Text  
End Sub
```

Aynı zamanda seçilen texte görede program işletebiliriz.

```
Private Sub Combo1_Click()  
    If Combo1.Text = "kırmızı" Then Text1.BackColor = RGB(255, 0, 0)  
    If Combo1.Text = "yeşil" Then Text1.BackColor = RGB(0, 255, 0)  
    If Combo1.Text = "mavi" Then Text1.BackColor = RGB(0, 0, 255)  
End Sub
```

## **Örnek 1**

Yazı sitilini ve büyüklüğünü değiştiren program.

```
Private Sub Combo1_Click()  
    Text2.FontName = Combo1.Text  
End Sub  
  
Private Sub Form_Load()  
    Dim i  
    UpDown1.BuddyControl = Combo1  
    UpDown1.BuddyProperty = "listindex"  
  
    UpDown2.BuddyControl = Text1  
    UpDown2.BuddyProperty = "text"  
  
    For i = 0 To Screen.FontCount - 1  
        Combo1.AddItem Screen.Fonts(i)  
    Next  
  
    Combo1.ListIndex = 0  
  
    UpDown1.Min = 0  
    UpDown1.Max = Combo1.ListCount - 1  
    UpDown1.Increment = 1  
  
    UpDown2.Min = 0  
    UpDown2.Max = 100  
    UpDown2.Increment = 5  
End Sub  
  
Private Sub Text1_Change()  
    Text2.FontSize = Val(Text1)  
End Sub
```

### **AddItem**

Combonun listesine yeni bir text eklemek için kullanılır.

```
Private Sub Combo1_Click()  
    If Combo1.Text = "kırmızı" Then Text1.BackColor = RGB(255, 0, 0)  
    If Combo1.Text = "yeşil" Then Text1.BackColor = RGB(0, 255, 0)  
    If Combo1.Text = "mavi" Then Text1.BackColor = RGB(0, 0, 255)  
End Sub
```

```
Private Sub Form_Load()  
    Combo1.AddItem "kırmızı"  
    Combo1.AddItem "mavi"  
    Combo1.AddItem "yeşil"  
    Combo1.ListIndex = 0  
End Sub
```

### **RemoveItem**

Combonun listesinden bir text çıkartmak için kullanılır.

```
Private Sub Command1_Click()  
    Combo1.RemoveItem 0  
End Sub
```

### **ListBox (Listeleme Kutusu)**

Elemanların gösterilmesini sağlayan komponentdir.

### **Columns**

Listbox ın içine kolon koymamıza yarar.

### **MultiSelect**

Listbox içerisindeki verilerin seçilmesini sağlar.

- 0 Seçim yapılamaz
- 1 Fare ile üzerine gidilen text seçilebilir.
- 2 Klavyeden Shift veya Ctrl tuşuyla seçim yapmamızı sağlar.

### **Style**

Listbox içeriği bir checkbox gibi olur.

### **AddItem**

ListBoxın içeriğine bir veri girişi yapmak için kullanılır.

```
Private Sub Form_Load()  
    List1.AddItem "kırmızı"  
    List1.AddItem "mavi"  
    List1.AddItem "yeşil",1 'Buradaki 1 rakamı yeşil yazısını 2.sırada görünmesini sağlar.  
End Sub
```

### **RemoveItem**

ListBoxın içeriğinden bir veriyi silmek için kullanılır.

```
Private Sub Command1_Click()  
    List1.RemoveItem 2  
End Sub
```

### **Text**

ListBoxın içeriğinden bir veriyi göstermek için kullanılır.

```
Private Sub List1_Click()  
    Text1.Text = List1.Text  
End Sub
```

### **Clear**

ListBoxın içeriğini tamamen silmek için kullanılır.

```
Private Sub Command1_Click()  
    List1.Clear  
End Sub
```

### **ListIndex**

ListBoxın içerisinde seçilmiş olan veriyi veya seçilecek olan verinin index numarası. Aynı zamanda yan yana 2-3 ListBox olduğunda birbirlerine bağlamak içinde kullanılır.

```
Private Sub Command1_Click()  
    List1.RemoveItem List1.ListIndex  
End Sub
```

### **ListCount**

ListBoxın içerisindeki veri sayısını verir. **List1.ListCount - 1** son elemanı verir.

```
Private Sub Command1_Click()  
    MsgBox ("listede " & List1.ListCount & " eleman var")  
End Sub
```

### **SelCount**

MultiSelect özelliği 1 veya 2 olan Listbox larda seçili eleman sayısını verir.

```
Private Sub Command1_Click()  
    MsgBox ("listede " & List1.SelCount & " eleman var")  
End Sub
```

### **Selected**

MultiSelect özelliği 1 veya 2 olan Listbox larda index numarasına göre seçim yapmayı sağlar.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Dim i  
    List2.Clear  
    For i = 0 To List1.ListCount - 1  
        If List1.Selected(i) Then  
            List2.AddItem List1.List(i)  
        End If  
    Next  
End Sub  
  
Private Sub Form_Load()  
    List1.AddItem "1"  
    List1.AddItem "2"  
    List1.AddItem "3"  
    List1.AddItem "4"  
    List1.AddItem "5"  
    List1.AddItem "6"  
    List1.AddItem "7"  
    List1.AddItem "8"
```

```
List1.AddItem "5"
List1.AddItem "6"
List1.AddItem "7"
List1.AddItem "8"
End Sub
```

### **Örnek 2**

Tüm listeyi seçmek için bir program.

```
Private Sub Command2_Click()
Dim a
  For a = 1 To List1.ListCount
    List1.Selected(a - 1) = True
  Next
End Sub
```

### **Örnek 3**

Listeye yazılan rakamları sıraya dizen program.

```
Private Sub Form_Load()
Randomize
  Dim i
  For i = 1 To 10
    List1.AddItem Int(Rnd * 100)
  Next
End Sub
```

```
Private Sub za_Click()
Dim i, j, c
  ReDim x(List1.ListCount - 1)
  For i = 0 To List1.ListCount - 1
    x(i) = List1.List(i)
  Next
  For i = 0 To List1.ListCount - 1
    For j = 0 To List1.ListCount - 1
      If Val(x(j)) < Val(x(i)) Then
        c = x(i)
        x(i) = x(j)
        x(j) = c
      End If
    Next
  Next
  For i = 0 To List1.ListCount - 1
    List1.List(i) = x(i)
  Next
End Sub
```

### **Örnek 4**

Ctrl-A Kısayolunun çalıştığı program.

```
Private Sub List1_KeyDown(KeyCode As Integer, Shift As Integer)
Dim a
  If Shift And vbCtrlMask Then
    Select Case KeyCode
      Case vbKeyA
        For a = 0 To List1.ListCount - 1
          List1.Selected(a) = True
        Next
    End Select
  End If
End Sub
```

## **Örnek 5**

Adres defteri

### **FORM1**

```
Private Sub Command1_Click()  
    Me.Hide  
    Load Form2  
    Form2.Show  
End Sub
```

```
Private Sub Command2_Click()  
    Me.Hide  
    Load Form3  
    Form3.Show  
End Sub
```

```
Private Sub Command3_Click()  
    Me.Hide  
    Load Form4  
    Form4.Show  
End Sub
```

```
Private Sub Command4_Click()  
    Me.Hide  
    Load Form5  
    Form5.Show  
End Sub
```

```
Private Sub Command5_Click()  
End  
End Sub
```

### **FORM2**

```
Private Sub cikis_Click()  
    Unload Form2  
    Form1.Show  
End Sub
```

```
Private Sub Form_Load()  
    Combo1.AddItem "1"  
    Combo1.AddItem "2"  
    Combo1.AddItem "3"  
    Combo1.AddItem "4"  
    Combo2.AddItem "Lisans"  
    Combo2.AddItem "Yüksek Lisans"  
    Combo2.AddItem "Doktora Lisans"  
    Combo3.AddItem "İstanbul"  
    Combo3.AddItem "Ankara"  
    Combo3.AddItem "İzmir"  
    Combo3.AddItem "Eskişehir"  
    Combo3.AddItem "Erzincan"  
    Combo1.ListIndex = 0  
    Combo2.ListIndex = 0  
    Combo3.ListIndex = 0  
    List1.Visible = False  
    List2.Visible = False  
    List3.Visible = False  
    List4.Visible = False  
    List5.Visible = False  
    List6.Visible = False  
    List7.Visible = False  
    List8.Visible = False  
    Dim ad, sad, sin, bol, adres, ilc, il, pkod
```

```

If Dir("ogren.dat") <> "" Then 'daha önce kayddilmiş dosya varsa onu aç yoksa
  Open "ogren.dat" For Input As #1
  While Not EOF(1) 'dosyanın sonuna kadar
    Input #1, ad, sad, sin, bol, adres, ilc, il, pkod
    List1.AddItem ad
    List2.AddItem sad
    List3.AddItem sin
    List4.AddItem bol
    List5.AddItem adres
    List6.AddItem ilc
    List7.AddItem il
    List8.AddItem pkod
  Wend
  Close #1
End If
End Sub
Private Sub Form_Unload(Cancel As Integer)
  Dim ad, sad, sin, bol, adres, ilc, il, pkod, i
  Open "ogren.dat" For Output As #1
  For i = 0 To List1.ListCount - 1
    ad = List1.List(i)
    sad = List2.List(i)
    sin = List3.List(i)
    bol = List4.List(i)
    adres = List5.List(i)
    ilc = List6.List(i)
    il = List7.List(i)
    pkod = List8.List(i)
    Write #1, ad, sad, sin, bol, adres, ilc, il, pkod
  Next
  Close #1
End Sub

Private Sub kaydet_Click()
  List1.AddItem Text1
  List2.AddItem Text2
  List3.AddItem Combo1.Text
  List4.AddItem Combo2.Text
  List5.AddItem Text3
  List6.AddItem Text4
  List7.AddItem Combo3.Text
  List8.AddItem Text5
End Sub

```

### **FORM3**

```

Private Sub Command1_Click()
  Unload Form3
  Form1.Show
End Sub

Private Sub Form_Load()
  Dim ad, sad, sin, bol, adres, ilc, il, pkod
  If Dir("ogren.dat") <> "" Then 'daha önce kayddilmiş dosya varsa onu aç yoksa
    Open "ogren.dat" For Input As #1
    While Not EOF(1) 'dosyanın sonuna kadar
      Input #1, ad, sad, sin, bol, adres, ilc, il, pkod
      List1.AddItem ad
      List2.AddItem sad
      List3.AddItem sin
      List4.AddItem bol
      List5.AddItem adres
      List6.AddItem ilc
      List7.AddItem il
      List8.AddItem pkod
    Wend
  End If
End Sub

```

```

    Close #1
End If
List2.Enabled = False
List3.Enabled = False
List4.Enabled = False
List5.Enabled = False
List6.Enabled = False
List7.Enabled = False
List8.Enabled = False
End Sub

Private Sub List1_Click()
List2.ListIndex = List1.ListIndex
List3.ListIndex = List1.ListIndex
List4.ListIndex = List1.ListIndex
List5.ListIndex = List1.ListIndex
List6.ListIndex = List1.ListIndex
List7.ListIndex = List1.ListIndex
List8.ListIndex = List1.ListIndex
List2.TopIndex = List1.TopIndex
List3.TopIndex = List1.TopIndex
List4.TopIndex = List1.TopIndex
List5.TopIndex = List1.TopIndex
List6.TopIndex = List1.TopIndex
List7.TopIndex = List1.TopIndex
List8.TopIndex = List1.TopIndex
End Sub

```

#### **FORM4**

```

Private Sub cikis_Click()
Unload Form5
Form1.Show
End Sub

Private Sub degis_Click()
Dim a
If List1.ListIndex < 0 Then
    MsgBox ("Lütfen bir eleman seçiniz")
    Exit Sub
End If
a = List1.ListIndex
List1.List(a) = Text1
List2.List(a) = Text2
List3.List(a) = Combo1.Text
List4.List(a) = Combo2.Text
List5.List(a) = Text3
List6.List(a) = Text4
List7.List(a) = Combo3.Text
List8.List(a) = Text5

Dim ad, sad, sin, bol, adres, ilc, il, pkod, i
Open "ogren.dat" For Output As #1
For i = 0 To List1.ListCount - 1
    ad = List1.List(i)
    sad = List2.List(i)
    sin = List3.List(i)
    bol = List4.List(i)
    adres = List5.List(i)
    ilc = List6.List(i)
    il = List7.List(i)
    pkod = List8.List(i)
    Write #1, ad, sad, sin, bol, adres, ilc, il, pkod
Next
Close #1

```

End Sub

```
Private Sub Form_Load()  
    Combo1.AddItem "1"  
    Combo1.AddItem "2"  
    Combo1.AddItem "3"  
    Combo1.AddItem "4"  
    Combo2.AddItem "Lisans"  
    Combo2.AddItem "Yüksek Lisans"  
    Combo2.AddItem "Doktora Lisans"  
    Combo3.AddItem "İstanbul"  
    Combo3.AddItem "Ankara"  
    Combo3.AddItem "İzmir"  
    Combo3.AddItem "Eskişehir"  
    Combo3.AddItem "Erzincan"  
    Combo1.ListIndex = 0  
    Combo2.ListIndex = 0  
    Combo3.ListIndex = 0  
    Dim ad, sad, sin, bol, adres, ilc, il, pkod  
    If Dir("ogren.dat") <> "" Then 'daha önce kaydedilmiş dosya varsa onu aç yoksa  
        Open "ogren.dat" For Input As #1  
        While Not EOF(1) 'dosyanın sonuna kadar  
            Input #1, ad, sad, sin, bol, adres, ilc, il, pkod  
            List1.AddItem ad  
            List2.AddItem sad  
            List3.AddItem sin  
            List4.AddItem bol  
            List5.AddItem adres  
            List6.AddItem ilc  
            List7.AddItem il  
            List8.AddItem pkod  
        Wend  
        Close #1  
    End If  
    List2.Enabled = False  
    List3.Enabled = False  
    List4.Enabled = False  
    List5.Enabled = False  
    List6.Enabled = False  
    List7.Enabled = False  
    List8.Enabled = False
```

End Sub

```
Private Sub Form_Unload(Cancel As Integer)  
    Dim ad, sad, sin, bol, adres, ilc, il, pkod, i  
    Open "ogren.dat" For Output As #1  
    For i = 0 To List1.ListCount - 1  
        ad = List1.List(i)  
        sad = List2.List(i)  
        sin = List3.List(i)  
        bol = List4.List(i)  
        adres = List5.List(i)  
        ilc = List6.List(i)  
        il = List7.List(i)  
        pkod = List8.List(i)  
        Write #1, ad, sad, sin, bol, adres, ilc, il, pkod  
    Next  
    Close #1
```

End Sub

```
Private Sub List1_Click()  
    List2.ListIndex = List1.ListIndex  
    List3.ListIndex = List1.ListIndex  
    List4.ListIndex = List1.ListIndex
```

```

List5.ListIndex = List1.ListIndex
List6.ListIndex = List1.ListIndex
List7.ListIndex = List1.ListIndex
List8.ListIndex = List1.ListIndex
List2.TopIndex = List1.TopIndex
List3.TopIndex = List1.TopIndex
List4.TopIndex = List1.TopIndex
List5.TopIndex = List1.TopIndex
List6.TopIndex = List1.TopIndex
List7.TopIndex = List1.TopIndex
List8.TopIndex = List1.TopIndex
Text1.Text = List1.Text
Text2.Text = List2.Text
Combo1.Text = List3.Text
Combo2.Text = List4.Text
Text3.Text = List5.Text
Text4.Text = List6.Text
Combo3.Text = List7.Text
Text5.Text = List8.Text
End Sub

```

### **Örnek 6**

Öğrenci takip programı

#### **FORM1**

3 Command Buton çiz,

```

Private Sub cikis_Click()
    End
End Sub

```

```

Private Sub Kayit_Click()
    Load Form2
    Form2.Show
End Sub

```

```

Private Sub Listele_Click()
    Load Form3
    Form3.Show
End Sub

```

#### **FORM2 Kayıt**

17 Label, 5 textbox, 11 Listbox, 8 option button, 3 command

```

Private Sub cikis_Click()
    Dim a
    a = MsgBox("Kaydetmek istiyormusunuz", 4)
    If a = 6 Then
        Unload Form2
    Else
        End
    End If
End Sub
Private Sub Form_Load()
    Combo1.AddItem 1
    Combo1.AddItem 2
    Combo1.AddItem 3
    Combo1.AddItem 4
    Combo1.AddItem 5
    Combo1.AddItem 6
    Combo1.ListIndex = 0
    Dim a, b, c, d, e, f, g, h, m, n
    If Dir("kurs.dat") <> "" Then

```

```

Open "kurs.dat" For Input As #1
  While Not EOF(1)
    Input #1, a, b, c, d, e, f, g, h, m, n
    List1.AddItem a
    List2.AddItem b
    List3.AddItem c
    List4.AddItem d
    List5.AddItem e
    List6.AddItem f
    List7.AddItem g
    List8.AddItem h
    List9.AddItem m
    List10.AddItem n
  Wend
Close #1
End If
Option1 = False
List11.Clear
List11.FontName = "tahoma"
List11.FontSize = 8
List11.FontBold = True
List11.ForeColor = RGB(255, 0, 0)
List11.BackColor = RGB(255, 255, 0)
Label12 = "1.200.000.000 TL."
Label13 = "700.000.000 TL."
Label14 = "130.000.000 TL."
Label15 = "90.000.000 TL."
Label16 = "150.000.000 TL."
Label17 = "120.000.000 TL."
End Sub
Private Sub Form_Unload(Cancel As Integer)
Dim a, b, c, d, e, f, g, h, m, n, i
  Open "kurs.dat" For Output As #1
  For i = 0 To List1.ListCount - 1
    a = List1.List(i)
    b = List2.List(i)
    c = List3.List(i)
    d = List4.List(i)
    e = List5.List(i)
    f = List6.List(i)
    g = List7.List(i)
    h = List8.List(i)
    m = List9.List(i)
    n = List10.List(i)
    Write #1, a, b, c, d, e, f, g, h, m, n
  Next
Close #1
End Sub
Private Sub kaydet_Click()
List1.AddItem Text1
List2.AddItem Text2
List3.AddItem Text3
List4.AddItem Text4
List5.AddItem Text5
If Option1 Then List6.AddItem "Windows 2000"
If Option2 Then List6.AddItem "Programcılık"
If Option3 Then List6.AddItem "Office 2000"
If Option4 Then List6.AddItem "Grafik"
If Option5 Then List6.AddItem "Muhasebe"
If Option6 Then List6.AddItem "İngilizce"
If Option7 Then List7.AddItem "Peşin"
If Option8 Then List7.AddItem "Taksitle"
List8.AddItem Combo1.Text
List9.AddItem Label8
List10.AddItem Label11
End Sub

```

```

Private Sub Option1_Click()
    List11.Clear
    List11.AddItem "Windows 2000 Server"
    List11.AddItem "Windows 2000 Professional"
    Label8 = "168 Saat"
End Sub
Private Sub Option2_Click()
    List11.Clear
    List11.AddItem "Visual Basic 6.0"
    List11.AddItem "Delphi"
    List11.AddItem "Visual C"
    List11.AddItem "Java Script"
    List11.AddItem "ASP"
    Label8 = "300 Saat"
End Sub
Private Sub Option3_Click()
    List11.Clear
    List11.AddItem "Windows 2000"
    List11.AddItem "Word 2000"
    List11.AddItem "Excel 2000"
    List11.AddItem "Outlook 2000"
    List11.AddItem "Access 2000"
    List11.AddItem "Powerpoint 2000"
    List11.AddItem "İnternet"
    Label8 = "96 Saat"
End Sub
Private Sub Option4_Click()
    List11.Clear
    List11.AddItem "Corel Draw"
    List11.AddItem "Photoshop"
    List11.AddItem "Freehand"
    Label8 = "60 Saat"
End Sub
Private Sub Option5_Click()
    List11.Clear
    List11.AddItem "Genel Muhasebe"
    List11.AddItem "Eta"
    List11.AddItem "Logo"
    List11.AddItem "Mikro"
    Label8 = "160 Saat"
End Sub
Private Sub Option6_Click()
    List11.Clear
    List11.AddItem "Yerli ve Yabancı Eğitimciler Eşliğinde"
    Label8 = "125 Saat"
End Sub
Private Sub Option7_Click()
    Combo1.Enabled = False
    Combo1.Text = "-----"
End Sub
Private Sub Option8_Click()
    Combo1.Enabled = True
    Combo1.ListIndex = 0
End Sub

```

### **FORM3 Listele ve Sil**

10 Listbox, 10 Label, 3 Command Button

```

Private Sub Command1_Click()
    Dim kay
    kay = MsgBox("Kaydetmek istiyormusunuz", 4)
    If kay = 6 Then
        Dim a, b, c, d, e, f, g, h, m, n, i
        Open "kurs.dat" For Output As #1
        For i = 0 To List1.ListCount - 1

```

```

    a = List1.List(i)
    b = List2.List(i)
    c = List3.List(i)
    d = List4.List(i)
    e = List5.List(i)
    f = List6.List(i)
    g = List7.List(i)
    h = List8.List(i)
    m = List9.List(i)
    n = List10.List(i)
    Write #1, a, b, c, d, e, f, g, h, m, n
Next
Close #1
Unload Form3
Else
Unload Form3
End If
End Sub

```

```

Private Sub Form_Load()
Dim a, b, c, d, e, f, g, h, m, n
If Dir("kurs.dat") <> "" Then
Open "kurs.dat" For Input As #1
While Not EOF(1)
Input #1, a, b, c, d, e, f, g, h, m, n
List1.AddItem a
List2.AddItem b
List3.AddItem c
List4.AddItem d
List5.AddItem e
List6.AddItem f
List7.AddItem g
List8.AddItem h
List9.AddItem m
List10.AddItem n
Wend
Close #1
End If

```

```

End Sub
Private Sub List1_Click()
List2.ListIndex = List1.ListIndex
List3.ListIndex = List1.ListIndex
List4.ListIndex = List1.ListIndex
List5.ListIndex = List1.ListIndex
List6.ListIndex = List1.ListIndex
List7.ListIndex = List1.ListIndex
List8.ListIndex = List1.ListIndex
List9.ListIndex = List1.ListIndex
List10.ListIndex = List1.ListIndex
List2.TopIndex = List1.TopIndex
List3.TopIndex = List1.TopIndex
List4.TopIndex = List1.TopIndex
List5.TopIndex = List1.TopIndex
List6.TopIndex = List1.TopIndex
List7.TopIndex = List1.TopIndex
List8.TopIndex = List1.TopIndex
List9.TopIndex = List1.TopIndex
List10.TopIndex = List1.TopIndex
End Sub

```

```

Private Sub sil_Click()
Dim sl
If List1.ListIndex < 0 Then
MsgBox ("Bir eleman seçiniz")
Exit Sub

```

```

Else
sl = List1.ListIndex
List1.RemoveItem sl
List2.RemoveItem sl
List3.RemoveItem sl
List4.RemoveItem sl
List5.RemoveItem sl
List6.RemoveItem sl
List7.RemoveItem sl
List8.RemoveItem sl
List9.RemoveItem sl
List10.RemoveItem sl
End If
End Sub

```

### **DriveListBox (Sürücü Listeleme Kutusu)**

Sistemdeki sürücü listesini göstermeye yarar.

```

Private Sub Drive1_Change()
Dir1.Path = Drive1.Drive
ChDrive Drive1.Drive
End Sub

```

### **Directory List Box (Dizin Listeleme Kutusu)**

```

Private Sub Dir1_Change()
File1.Path = Dir1.Path
ChDir Dir1.Path
End Sub

```

### **FileList Box (Dosya Listeleme Kutusu)**

```

Private Sub Combo1_Click()
File1.Pattern = Combo1.Text
End Sub

```

```

Private Sub Combo1_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
File1.Pattern = Combo1.Text
End If
End Sub

```

```

Private Sub Dir1_Change()
File1.Path = Dir1.Path
ChDir Dir1.Path
Text1.Text = Dir1.Path + "\ " + File1.FileName
End Sub

```

```

Private Sub Drive1_Change()
Dir1.Path = Drive1.Drive
ChDrive Drive1.Drive
Text1.Text = Dir1.Path + "\ " + File1.FileName
End Sub

```

```

Private Sub File1_Click()
Text1.Text = Dir1.Path + "\ " + File1.FileName
End Sub

```

```

Private Sub File1_KeyDown(KeyCode As Integer, Shift As Integer)
Dim ac
If KeyCode = 13 Then
ac = Shell(File1.FileName, vbNormalFocus)
End If
End Sub

```

```
Private Sub Form_Load()  
    Combo1.AddItem "*"*.*)" "  
    Combo1.ListIndex = 0  
    Text1.Text = Dir1.Path + "\" + File1.FileName  
End Sub
```

### **PictureBox (Resim Kutusu)**

Programınıza resim eklemek için kullanılır.

#### **Align**

- |   |                |
|---|----------------|
| 0 | yok            |
| 1 | Üste hizalı    |
| 2 | Aşağıya hizalı |
| 3 | Sola hizalı    |
| 4 | Sağa hizalı    |

### **Image (Resim Gösterme Kontrolü)**

#### **Stretch**

Otomatik boyutlandırma.

### **Common Dialog (Diyalog Pencereleeri)**

Microsoft Common Dialog 6.0 Componenti seçilir. Aç, Kaydet, Renk, Help, Yazdır ve Font diyalog kutularının çıkartılmasını sağlar.

CommonDialog1.Action = 1	Aç
CommonDialog1.Action = 2	Yeni Adla Kaydet
CommonDialog1.Action = 3	Renk
CommonDialog1.Action = 4	Font
CommonDialog1.Action = 5	Yazdır
CommonDialog1.Action = 6	Help

**CommonDialog1.Action = 1 Aç**

#### **CancelError**

Dosya aç kutusunda iptal düğmesine basılınca uyarı vermez.

CancelError = False

#### **Flags**

Dosya açma özelliklerini etkiler.

Flags = 1 Aç kutusu üzerinde tek eleman seçilebilir ve salt okunur kutusu seçili.  
Flags = 2 Aç kutusu üzerinde tek eleman seçilebilir ve salt okunur kutusu seçili değil.  
Flags = 4 Aç kutusu üzerinde tek eleman seçilebilir ve salt okunur kutusu olmaz  
Flags = 256 Aç kutusu üzerinde tek eleman seçilebilir.  
Flags = 512 Aç kutusu üzerinde birden fazla eleman seçilebilir. Win 3.1 gibi olur.  
Flags = 4096 Klasörde olmayan bir dosya açılırsa hata kodu verir.

#### **DialogTitle**

Dosya aç kutusunun ismini değiştirir.

DialogTitle = "Dosya Aç"

### **DefaultExt**

Dosya uzantısı belirtilmezse bile otomatik olarak uzantıyı ekler.

DefaultExt = "dat"

### **FileName**

Dosya aç kutusunda açılmasını istediğiniz dosyalar.

FileName = "\*.txt"

### **Filter**

Dosya aç kutusunda açılmasını istediğiniz dosyalar.

Filter= "Text dosyaları (\*.txt)|\*.txt|Bitmap(\*.bmp)|\*.bmp|Tüm resimler(\*.jpg;\*.wmf;\*.ico)| \*.jpg;\*.wmf;\*.ico|Tüm dosyalar(\*.\*)|\*.\*|"

### **FilterIndex**

Dosya aç kutusunda açılmasını istediğiniz dosyalardan default olarak gelmesini istediğinizi yazarsınız. İlk sıradaki 1 numaradır.

CommonDialog1.FilterIndex = 2

### **InitDir**

Dosya aç kutusunda default olarak açılmasını istediğiniz directory.

CommonDialog1.InitDir = "c:\program files"

### **CommonDialog1.Action = 2 Kaydet**

Flags = 2 Kaydet kutusunda aynı isimli dosyaları kaydederken uyarı verir.

### **CommonDialog1.Action = 3 Color**

Flags = 2 Color kutusunda ayrıntılarda açılır.

Flags = 4 Ayrıntılar gizlenir.

### **CommonDialog1.Action = 4 Font**

Flags = 1 Screen Fontlar gelir.

Flags = 2 Printer Fontlar gelir.

Flags = 3 Tüm Fontlar gelir.

Flags = 257 Font konusunda özellikli gelir.

Flags = 258 Font konusunda özellikli gelir.

Flags = 8192 Min-Max arasında bir değer girilmezse uyarı verir.

Flags =131074 Font kutusunda yalnız TTF fontlar gözüktür.

Flags =131330 Font kutusunda yalnız TTF fontlar gözüktür ve özelliklidir.

CommonDialog1.Min = Minimum değer

CommonDialog1.Max = Maximum değer

### **CommonDialog1.Action = 5 Print**

PrinterDefault = True ise varsayılan printerin ayarları aktif olur.

Copies = Kopya sayısı verilir.

Flags = 1 Yazdır kutusunda Seçili seçili olur.

Flags = 2 Yazdır kutusunda sayfalar seçili olur.

Flags = 4 Yazdır kutusunda Seçili kısım pasif olur.

Flags = 8 Yazdır kutusunda Sayfalar pasif olur ve harmanla açılır.

Flags = 16 Yazdır kutusunda Tümü seçili olur ve harmanlama şekli açılır.

Flags = 32 Yazdır kutusunda Dosyaya yazdır seçili olur.

Flags = 64 Yazdır kutusunda Yazıcı ayarları görünür.  
Flags = 2048 Yazdır kutusunda Help menüsünde görünür.  
Flags = 524288 Yazdır kutusunda Dosyaya yazdır kutusu pasif olur.  
Flags = 1048576 Yazdır kutusunda Dosyaya yazdır ekranda olmaz.

Orientation = 1 Dikey 2 Yatay olur.

### **Örnek 1**

Aç,Kaydet, Renk , Print, Font

```
Private Sub ac_Click()  
On Local Error Goto hata  
Dim a  
CommonDialog1.Filter = "metin|*.txt|"  
CommonDialog1.ShowOpen  
Open CommonDialog1.FileName For Input As #1  
Text1.Text = ""
```

```
While Not EOF(1)  
Line Input #1, a  
Text1 = Text1.Text + a + Chr(13) + Chr(10)  
Wend  
Close #1  
Exit Sub  
Hata:  
Exit Sub
```

End Sub

```
Private Sub font_Click()  
CommonDialog1.FontName = Text1.FontName  
CommonDialog1.FontSize = Text1.FontSize  
CommonDialog1.FontBold = Text1.FontBold  
CommonDialog1.FontItalic = Text1.FontItalic  
CommonDialog1.FontUnderline = Text1.FontUnderline  
CommonDialog1.FontStrikethru = Text1.FontStrikethru  
CommonDialog1.Color = Text1.ForeColor  
'Bu kodları font CommonDialog1 açıldığında  
'Text kutusunun özelliklerinin otomatik olarak gelmesini sağlar.
```

```
CommonDialog1.Flags = 257  
CommonDialog1.ShowFont
```

```
Text1.FontName = CommonDialog1.FontName  
Text1.FontSize = CommonDialog1.FontSize  
Text1.FontBold = CommonDialog1.FontBold  
Text1.FontItalic = CommonDialog1.FontItalic  
Text1.FontUnderline = CommonDialog1.FontUnderline  
Text1.FontStrikethru = CommonDialog1.FontStrikethru  
Text1.ForeColor = CommonDialog1.Color
```

End Sub

```
Private Sub color_Click()  
CommonDialog1.Flags = cdlCCPreventFullOpen  
CommonDialog1.ShowColor  
Text1.BackColor = CommonDialog1.Color  
End Sub
```

```
Private Sub print_Click()  
CommonDialog1.Flags = &H40  
CommonDialog1.ShowPrinter  
Printer.Print Text1  
Printer.EndDoc  
End Sub
```

```

Private Sub kay_Click()
Dim a
CommonDialog1.DialogTitle = "Kaydet"
CommonDialog1.DefaultExt = ".txt"
CommonDialog1.Flags = 2
CommonDialog1.ShowSave
Open CommonDialog1.FileName For Output As #1
a = Text1
Print #1, a
Close #1
End Sub

```

## **MSMasked (Formatlı Giriş)**

Microsoft Masked Edit Control 6.0

Bu kontrol sayesinde formlarınızda özel formatlı telefon, tarih, saat gibi verileri hızlı bir şekilde girmesi sağlanır.

### **Mask**

Kullanıcının ne şekilde veri girileceği belirlenir.

### **Örnek**

Telefon	(###) ### ## ##	(216) 455 55 55
Telefon	#.###.### ## ##	0.216.455 55 55
Tarih	##-??-####	12-Ara-2000
Tarih	##-##-##	12-12-00
Saat	##:## ??	09:11 PM
Saat	##:##	09:11
Rakam	###	
Harf	???	

### **Appearance**

MaskEdit textin 3D ve normal gözükmesini sağlar.

### **AutoTab**

Kullanıcının veri girişini tamamlamasının ardından setfocus un diğer komponente geçmesini sağlar.

### **Text**

Kullanıcının ne şekilde veri girdiyse o şekilde alınmasını sağlar.

### **ClipText**

Kullanıcının ne şekilde veri girilmesi önemli değildir. Veriler mask özelliği olmadan görünür.

### **ClipMode**

0 verilirse kopyalama işlemlerinde text özelliğini  
1 verilirse kopyalama işlemlerinde cliptext özelliğini alır.

### **Format**

Texte yazılan bilgilerin ekranda görünme şekillerini değiştirir.

Para	#,##0.00TL
Tam sayı	0
Basamaklı sayı	#,##0
Yüzde sayı	%0
Üstlü sayı	0.00E+00
Tarih,saat	c

Tarih gg/aa/yyyy	dddd
Tarih gün gg ay yyyy	dddddd
Tarih gg-ay-yy	dd-mmm-yy
Saat 24 saat	hh:mm
Saat 12 saat AM/PM	hh:mm AM/PM
Saat sa:dk:sn	ttt

### **MaxLength**

Maskedit texte en fazla 64 karakter bilgi girilebilir.

### **PromptChar**

MaskEdit texte yazılan mask aralarına konan \_ işareti değiştirilebilir.

### **PromptInclude**

PromptChar karakterinin gizlenmesi sağlanabilir.

### **Örnek 1**

```
Private Sub tarih_LostFocus()  
    If IsDate(tarih) = False Then  
        MsgBox ("Lütfen uygun tarih giriniz")  
        tarih.Mask = ""  
        tarih.Text = ""  
        tarih.Mask = "##/##/####"  
        tarih.SetFocus  
    End If  
End Sub
```

### **SSTab**

Microsoft Tabbed Dialog Control 5.0

Bu kontrol sayesinde seperatör kullanmamızı sağlar. Tab üzerine bir komponent yerleştirmek için Frame kullanılır.

### **Tabs**

Tabs üzerinde bulunacak tabs sayısı belirlenir. En fazla 99 olur.

### **Tab**

Program çalışınca hangi Tabın aktif olarak geleceği belirlenir.

### **TabHeight**

Tabın başlık kısmının yüksekliğini ayarlar.

### **TabMaxWidth**

Tabın başlık kısmının genişliğini ayarlar.

### **TabOrientation**

4 özellik alır.

- |   |                    |
|---|--------------------|
| 0 | Tablar üstte       |
| 1 | Tablar altta       |
| 2 | Tablar solda       |
| 3 | Tablar sağda olur. |

### **TabsPerRow**

Tab üzerinde bir satırda yerleştirilecek eleman sayısını belirlenir.

### **TabEnabled(index)**

Program çalışınca herhangi bir Tab'a ulaşılmaması için kullanılır.

```
Private Sub Form_Load()  
    SSTab1.TabEnabled(3) = False  
End Sub
```

### **Picture**

Tabın baş tarafına resim koymak için kullanılır.

### **PreviousTab**

Bir Tabdan başka bir taba geçildiği zaman hangi tabdan geçildiğini bulmak için kullanılır.

### **Style**

0 olunca tabın otomatik veya manuel olur.  
1 olunca tabın genişliği sabit olur.

### **ShowFocus Rect**

True olunca program çalışınca Tabın üzerine gelince caption etrafı focuslu olur.

### **WordWrap**

Tabın üzerindeki captiondaki yazının kaymasını sağlar.

### **Richtextbox**

Microsoft Rich Textbox Control 6.0

Bu kontrol sayesinde zenginleştirilmiş text özelliklere sahiptir.

### **SelChange**

RichTextBox üzerinde kursorun yer değişmesi sonucu oluşan olaylar.

### **Örnek 1**

```
Private Sub Chcbold_Click()  
    RichTextBox1.SelBold = Chcbold.Value  
End Sub  
  
Private Sub hcitalic_Click()  
    RichTextBox1.SelItalic = hcitalic.Value  
End Sub  
  
Private Sub hcunder_Click()  
    RichTextBox1.SelUnderline = hcunder.Value  
End Sub  
  
Private Sub Command1_Click()  
    CommonDialog1.Flags = 258  
    CommonDialog1.ShowFont  
    RichTextBox1.SelFontName = CommonDialog1.FontName  
    RichTextBox1.SelFontSize = CommonDialog1.FontSize  
    RichTextBox1.SelUnderline = CommonDialog1.FontUnderline  
    RichTextBox1.SelStrikeThru = CommonDialog1.FontStrikethru  
    RichTextBox1.SelColor = CommonDialog1.Color  
    RichTextBox1.SelBold = CommonDialog1.FontBold  
    RichTextBox1.SelItalic = CommonDialog1.FontItalic  
End Sub
```

```
Private Sub Form_Load()  
    RichTextBox1.TextRTF = "En Büyük Galatasaray Başka Büyük Yok"  
End Sub
```

```
Private Sub oporta_Click()  
    If oporta Then RichTextBox1.SelAlignment = 2  
End Sub
```

```
Private Sub opsag_Click()  
    If opsag Then RichTextBox1.SelAlignment = 1  
End Sub
```

```
Private Sub opsol_Click()  
    If opsol Then RichTextBox1.SelAlignment = 0  
End Sub
```

## **Örnek 2**

Daktilo gibi yazan program.

```
Option Explicit  
Dim i As Double
```

```
Private Sub Command1_Click()  
    Dim a  
    CommonDialog1.Filter = "metin|*.txt|"  
    CommonDialog1.Flags = 259  
    CommonDialog1.ShowOpen  
    Open CommonDialog1.FileName For Input As #1  
    Text1.Text = ""
```

```
While Not EOF(1)  
    Line Input #1, a  
    rt1.Text = rt1.Text + a + Chr(13) + Chr(10)  
Wend  
Close #1  
Timer1.Enabled = True
```

```
End Sub
```

```
Private Sub Form_Load()  
    Timer1.Interval = 10  
    Timer1.Enabled = False  
End Sub
```

```
Private Sub Text2_Change()  
    Dim a  
    a = Text2.SelStart  
    Text2 = Format(Text2, "###,###")  
    Text2.SelStart = a + 1  
End Sub
```

```
Private Sub Timer1_Timer()  
    Text1.Text = Left(rt1.Text, i): Beep  
    i = i + 1  
    If i = Len(rt1.Text) Then Timer1.Enabled = False  
End Sub
```

## **Popup Menü**

**PopupMenu** menü adı, flags, x, y, Defaultmenu

Farenin sağ klick yapılıncaya oluşan menüye denir.

**X.Y**

Menünün ekranda çıkacağı koordinatlar.

### ***Flags***

- 0 Farenin sol tuşuyla eleman seçilecek ve menü x koordinatının sağında açılacak.
- 2 Farenin sağ tuşuyla eleman seçilecek.
- 4 Menü x koordinatını ortalayacak
- 8 Menü x koordinatının solunda çıkacak

### ***DefaultMenü***

Menü açıldığı zaman bir parametre bold çıkartılabilir.

Menü editöre girilir ve orada bir menü oluşturulur. Menülerin Nameleri yazılırken sonra şaşırmamak için önlerine aynı harfleri koyun (oomenü gibi) ve Visible= False olsun

### ***Örnek 1***

Menü isimleri

Kopyala  
Yapıştır  
Kes  
Sil

---- - ` Düz çizgi koymak için kullanılır.

Font

Zemin Rengi      olsun.

**Private Sub Form\_Load()**

Label1.Caption = X

Label2.Caption = Y

**End Sub**

**Private Sub mnfont\_Click()**

On Error GoTo hata

With CommonDialog1

.Flags = 259

.ShowFont

End With

If CommonDialog1.FontName <> "" Then ActiveControl.FontName = CommonDialog1.FontName

ActiveControl.FontSize = CommonDialog1.FontSize

ActiveControl.ForeColor = CommonDialog1.Color

ActiveControl.FontUnderline = CommonDialog1.FontUnderline

ActiveControl.FontBold = CommonDialog1.FontBold

ActiveControl.FontItalic = CommonDialog1.FontItalic

hata: Exit Sub

**End Sub**

**Private Sub mnkes\_Click()**

Clipboard.SetText ActiveControl.SelText

ActiveControl.SelText = ""

**End Sub**

**Private Sub mnkopyala\_Click()**

Clipboard.SetText ActiveControl.SelText

**End Sub**

**Private Sub mnsil\_Click()**

ActiveControl.SelText = ""

**End Sub**

**Private Sub mnyapistir\_Click()**

ActiveControl.SelText = Clipboard.GetText

**End Sub**

```

Private Sub mnzemin_Click()
On Error GoTo hata
    With CommonDialog1
        .CancelError = True
        .ShowColor
    End With
    ActiveControl.BackColor = CommonDialog1.Color
Exit Sub
hata:
    Exit Sub
End Sub

Private Sub text1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 4 Then PopupMenu mnmenu, 4, ActiveControl.Left, ActiveControl.top, mnkopyala
End Sub

Private Sub Text1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Label1.Caption = X
    Label2.Caption = Y
End Sub

Private Sub Text2_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 4 Then PopupMenu mnmenu, 4, ActiveControl.Left, ActiveControl.top, mnkopyala
End Sub

Private Sub oozrengi_Click()
    CommonDialog1.ShowColor
    Text1.BackColor = CommonDialog1.Color
End Sub

```

## **MSFlexGrid**

Microsoft FlexGrid Control (MsflxGrd.OCX)

### **Cols**

Grid üzerinde oluşturulacak Kolon sayısı.

### **ColAlignment(index)**

Grid üzerindeki aktif kolonun alignment özelliğini belirler.

- 0 Sola, üste yanaşık
- 1 Sola, orta yanaşık
- 2 Sola, alta yanaşık
- 3 Orta, üste yanaşık
- 4 Orta, orta yanaşık
- 5 Orta, alta yanaşık
- 6 Sağ, üst yanaşık
- 7 Sağ, orta yanaşık
- 8 Sağ, alta yanaşık
- 9 Rakamlar sağ-orta ve metinler sol-orta yanaşık

Dim i

```

MSFlexGrid1.Rows = 15
MSFlexGrid1.Cols = 15

```

```

MSFlexGrid1.Col = 0
For i = 0 To MSFlexGrid1.Rows - 1
    MSFlexGrid1.Row = i
    MSFlexGrid1.Text = i

```

```
MSFlexGrid1.RowHeight(i) = 500
Next
MSFlexGrid1.ColAlignment(0) = 4

MSFlexGrid1.Row = 0
For i = 1 To MSFlexGrid1.Cols - 1
    MSFlexGrid1.FixedAlignment(i) = 4
    MSFlexGrid1.Col = i
    MSFlexGrid1.Text = Chr(64 + i)
    MSFlexGrid1.ColWeight(i) = 1500
Next
```

### **ColWidth(index)**

Grid üzerindeki kolonların genişliğini belirler.

### **FillStyle**

0 Değişiklikler tek hücreyi etkiler.  
1 Değişiklikler tüm gridi etkiler.

### **FixedCols, Rows**

Grid üzerinde normalde ilk satır ve sütuna veri girişi yapılmaz. İstenirse bu sayı arttırılabilir.

### **FixedAlignment(index)**

Grid üzerindeki fixed kolonun alignment özelliğini belirler.

### **GridLines**

Grid üzerinde çizgilerin özelliklerini değiştirir.

### **GridLineWidth**

Grid üzerinde çizgilerin kalınlığını ayarlar.

### **GridLinesFixed**

Grid üzerinde çizgilerin özelliklerini değiştirir.

### **HighLight**

Grid üzerindeki aktif hücre üzerindeki taramayı kapatır.

### **Rows**

Grid üzerindeki satır sayısını ayarlar.

### **RowHeight(index)**

Grid üzerindeki satırların yüksekliğini belirler.

### **Col, Row**

Grid üzerinde o andaki aktif hücreyi belirtir.

### **Text**

Grid üzerindeki hücreye yazılacak yazıyı belirler.

### **Clip**

Grid üzerindeki hücrelere birden veri atamak için kullanılır. Chr(9) ve Chr(13) kullanılabilir.

### **ColSel**

Seçilen alanların ilk kolon değerini verir. MouseMove yazılır.

### **RowSel**

Seçilen alanların son satır değerini verir. MouseMove yazılır.

### **TextMatrix**

Grid üzerinde koordinat vererek yazı yazmak için kullanılır.

```
MSFlexGrid1.TextMatrix(1, 1) = "Oguz"
```

### **Sort**

Grid üzerinde sıralama yapmak için kullanılır. Seçili kısım varsa o kısımda uygulanır. Seçilmiş alan yoksa aktif sütunda yapılır.

0	Sıralama yok
1	Artan sıra
2	Azalan sıra
3	Artan sayısal sıralama
4	Azalan sayısal sıralama
5	Büyük-küçük harf ayrımı yapmayan artan sıralama
6	Büyük-küçük harf ayrımı yapmayan azalan sıralama
7	Büyük-küçük harf ayrımı yapan artan sıralama
8	Büyük-küçük harf ayrımı yapan azalan sıralama
9	özel

### **MergeCells**

Grid üzerinde hücreleri birleştirmek için kullanılır.

```
Private Sub MSFlexGrid1_KeyPress(KeyAscii As Integer)
    Text1 = MSFlexGrid1.Text
    Text1.SetFocus
    Text1.SelStart = Len(Text1)
    SendKeys Chr(KeyAscii)
End Sub
Private Sub Text1_Change()
    MSFlexGrid1.Text = Text1
End Sub
Dim a
MSFlexGrid1.MergeCells = 1
For a = 0 To MSFlexGrid1.Cols - 1
    MSFlexGrid1.MergeCol(a) = True
Next
```

### **ColPosition, RowPosition**

Grid üzerinde index numarası verilen satır veya sütunların yerini değiştirir .

```
MSFlexGrid1.ColPosition(2) = 3
```

### **ColFont**

Grid üzerindeki aktif hücrenin font özelliklerini değiştirir.

### **CellFont, CellFontSize**

Grid üzerindeki seçilmiş hücrenin font özelliklerini değiştirir.

### **Additem**

Grid üzerine veri eklemek için kullanılır.

```
'Satır Ekle
MSFlexGrid1.AddItem ""

MSFlexGrid1.AddItem "",1 'Birinci satırın sonuna ekler.
```

### **Removeitem**

Grid üzerine veri silmek için kullanılır.

### **Clear**

Grid üzerine veri silmek için kullanılır.

```
MSFlexGrid1.Clear
```

```
Private Sub ac_Click()
Dim a
If Dir("oguz.xls") <> "" Then
Open "oguz.xls" For Input As #1
MSFlexGrid1.Col = 0
MSFlexGrid1.Row = 1
MSFlexGrid1.ColSel = MSFlexGrid1.Cols - 1
MSFlexGrid1.RowSel = MSFlexGrid1.Rows - 1
Input #1, a
MSFlexGrid1.Clip = a
Close #1
End If
End Sub

Private Sub Command1_Click()
MSFlexGrid1.ColPosition(2) = 3
End Sub

Private Sub Command2_Click()
'Satır Ekle
MSFlexGrid1.AddItem "" : End Sub
Private Sub Command3_Click()
MSFlexGrid1.Clear
End Sub

Private Sub font_Click()
MSFlexGrid1.FillStyle = 1
CommonDialog1.Flags = 259
CommonDialog1.ShowFont
MSFlexGrid1.CellFontName = CommonDialog1.FontName
MSFlexGrid1.CellFontSize = CommonDialog1.FontSize
MSFlexGrid1.CellFontBold = CommonDialog1.FontBold
MSFlexGrid1.CellForeColor = CommonDialog1.Color
MSFlexGrid1.CellFontStrikeThrough = CommonDialog1.FontStrikeThrough
MSFlexGrid1.CellFontUnderline = CommonDialog1.FontUnderline

End Sub

Private Sub Form_Load()
Dim i
MSFlexGrid1.Cols = 4
MSFlexGrid1.Rows = 10
MSFlexGrid1.FixedCols = 0
MSFlexGrid1.TextMatrix(0, 0) = "Adı"
MSFlexGrid1.TextMatrix(0, 1) = "Soyadı"
MSFlexGrid1.TextMatrix(0, 2) = "Telefon"
MSFlexGrid1.TextMatrix(0, 3) = "Adresi"

End Sub
```

```

Private Sub kaydet_Click()
Dim a
  Open "oguz.xls" For Output As #1
  MSFlexGrid1.Col = 0
  MSFlexGrid1.Row = 1
  MSFlexGrid1.ColSel = MSFlexGrid1.Cols - 1
  MSFlexGrid1.RowSel = MSFlexGrid1.Rows - 1
  a = MSFlexGrid1.Clip
  Write #1, a
  Close #1
End Sub

Private Sub MSFlexGrid1_KeyPress(KeyAscii As Integer)
  Text1 = MSFlexGrid1.Text
  Text1.SetFocus
  Text1.SelStart = Len(Text1)
  SendKeys Chr(KeyAscii)
End Sub

Private Sub MSFlexGrid1_Click(KeyAscii As Integer)
  Text1 = MSFlexGrid1.Text
  Text1.SelStart = Len(Text1)
End Sub

Private Sub MSFlexGrid1_EnterCell(KeyAscii As Integer)
  Text1 = MSFlexGrid1.Text
  Text1.SelStart = Len(Text1)
End Sub

Private Sub sirala_Click()
  Dim a
  MSFlexGrid1.MergeCells = 1
  For a = 0 To MSFlexGrid1.Cols - 1
    MSFlexGrid1.MergeCol(a) = True
  Next
  For a = 0 To MSFlexGrid1.Rows - 1
    MSFlexGrid1.MergeRow(a) = True
  Next
End Sub

Private Sub Text1_Change()
  MSFlexGrid1.Text = Text1
End Sub

Private Sub zrenge_Click()
  MSFlexGrid1.FillStyle = 1
  CommonDialog1.ShowColor
  MSFlexGrid1.CellBackColor = CommonDialog1.Color
End Sub

Private Sub sil_Click()
  Dim a,b,c,i
  a=MSFlexGrid1.Row
  b=MSFlexGrid1.Row
  if a > b Then
  c=a
  a=b
  b=c
  End if
  For i=a To b
    MSFlexGrid1.RemoveItem a
  Next
End Sub

```

## **ProgressBar (İlerleme Çubuğu)**

Yapılan işlemlerin ilerleme durumunu gösterir.

### **Min, Max**

ProgressBar'ın alacağı değerler yazılır.

### **Orientation**

ProgressBar'ın dikey veya yatay olması sağlanır.

### **Scrolling**

ProgressBar'ın üzerindeki artışlar kutu kutu veya düz gözükür.

### **Value**

ProgressBar'ın artma oranı.

### **Örnek 1**

```
Private Sub Command1_Click()  
    Timer1.Enabled = True  
    Timer1.Interval = 1000  
End Sub  
  
Private Sub Timer1_Timer()  
    ProgressBar1.Value = ProgressBar1.Value + 5  
    If ProgressBar1.Value > 99 Then  
        MsgBox ("Yükleme tamamlandı")  
        Timer1.Enabled = False  
    End If: End Sub
```

### **Örnek 2**

```
On Local Error GoTo hata  
Static a, b  
For a = 1 To 10000  
    ProgressBar1.Value = ProgressBar1.Value + b  
    If ProgressBar1.Value >= 9999 Then  
        MsgBox ("Yükleme tamamlandı")  
        b = a + b  
    End If  
hata:  
Exit Sub  
Next
```

### **Örnek 3**

```
Private Sub Command1_Click()  
    ProgressBar1.Value = 0  
    ProgressBar2.Value = 0  
    Timer1.Enabled = True  
    Timer1.Interval = 100  
End Sub  
  
Private Sub Timer1_Timer()  
    If ProgressBar1.Value >= ProgressBar1.Max Then  
        ProgressBar1.Value = 0  
    Else  
        ProgressBar1.Value = ProgressBar1.Value + 10  
    End If
```

```
If ProgressBar1.Value = ProgressBar1.Max Then
    ProgressBar2.Value = ProgressBar2.Value + 10
    If ProgressBar2.Value = ProgressBar1.Max Then
        MsgBox ("Yükleme Tamamlandı")
        Timer1.Enabled = False
        Exit Sub
    End If
End If
End If
End Sub
```

## **Slider**

Microsoft Windows Common Controls 6.0 (MSCOMCTL.OCX)

Ekrana yerleştirilen herhangi bir komponentin özelliklerini değiştirmede kullanılır.

### **LargeChange**

Mouse ile Slider üzerine tıkladığımız zaman ne kadarlık bir değişim olacağı belirlenir .

### **Max**

Slider in alacağı en fazla değer.

### **Min**

Slider in alacağı en düşük değer.

### **Orientation**

0	Yatay
1	Dikey

### **SmallChange**

Slider in üzerinde ok tuşlarıyla ne kadarlık bir değişim olacağını belirler.

### **TextPosition**

0	Yazılar Slider in üstünde.
1	Yazılar Slider in altında.

### **TickFrequency**

Slider in min-max arasında kaç adet çizgi olacağını belirler.

### **TickStyle**

Slider in üzerindeki çizgilerin stillerini belirler.

0	Çizgiler altta
1	Çizgiler üstte
2	Çizgiler iki tarafta
3	Çizgi yok.

### **Value**

Slider üzerinde oluşan değişim miktarı.

## **DateTimePicker**

Ekrana bir tarih gösterici koymak için kullanılır.

## **CheckBox**

DateTimePicker üzerine bir checkbox koyarak kullanıcının tarihi onaylamasını sağlayabilirsiniz. Eğer check işaretlenmezse tarih boş olarak döner.

### **Format**

0	Uzun tarih
1	Kısa tarih
0	Saat
1	

### **MinDate, MaxDate**

Minimum ve Max tarih girilir.

### **UpDown**

Eğer bu seçilirse Combobox gibi gözükmez sadece updown larla değişim olabilir.

### **Örnek 1**

```
Private Sub DTPicker1_CallbackKeyDown(ByVal KeyCode As Integer, ByVal Shift As Integer, ByVal CallbackField As String,
CallbackDate As Date)
    Text1.Text = DTPicker1.Value
End Sub
```

### **MonthView**

Ekranı bir takvim koymak için kullanılır. Mouse ile ay üzerine tıklanırsa ay seçmek için otomatik bir liste çıkar. Yıl üzerine tıklanırsa bir updown çıkar.

### **MaxSelCount**

Multiselect özelliği true olan bir monthiew de en fazla kaç adet tarih seçilebileceğini belirtmek için kullanılır.

### **MonthRows, MonthColumns**

Ekranı kaç aylık Monthiew olacağını belirtmek için kullanılır.

### **ShowWeekNumber**

Haftanın numaraları gösterilir.

### **StartOfWeek**

Haftanın başlangıç günü ayarlanır.

### **StartOfWeek**

Haftanın başlangıç günü ayarlanır.

```
Private Sub MonthView1_DateClick(ByVal DateClicked As Date)
    Text1.Text = DateClicked
End Sub
```

```
Private Sub MonthView1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Text1.Text = MonthView1.SelStart & "-" & MonthView1.SelEnd
End Sub
```

### **Coolbar**

Microsoft Windows Common Controls-3 6.0 (COMCT332.OCX)

Ekranı toolbar yerleştirmek için kullanılır.

0	Ekranı bir tane ImageList1 eklenir ve içerisine koymak istediğiniz resimler yerleştirilir.
1	Coolbar üzerine bir tane Toolbar yerleştirilir.

- 2 Toolbar üzerinde sağ klik yapılır ve ButtonHeight = 330 yapılır.
- 3 ImageList eklenir.
- 4 Button sekmesinden butonlar eklenir.
- 5 Coolbar properties. Bands sekmesinden.
- 6 Child bölümüne Toolbar eklenir.

Bands sekmesinden.

### **InsertBand**

Coolbar üzerine bant eklemek için kullanılır.

### **FixedOrder**

Coolbarın boyutlarıyla oynanmasını istemiyorsanız burası işaretlenir.

### **RemoveBand**

Coolbar üzerinden bant silmek için kullanılır.

### **MinWidth**

Coolbarın en fazla küçülebileceği mesafe.

### **MinHeight**

Coolbarın yüksekliği.

### **Key**

Kod yazarken kullanılacak ismi.

### **Image**

Coolbarın başında bulunan resim.

### **NewRow**

Coolbarın üzerine yeni bir row ekler.

### **NewRow**

Coolbarın üzerine yeni bir row ekler.

### **Toolbar**

Ekrana toolbar yerleştirmek için kullanılır. Öncelikle ekrana bir tane toolbar yerleştirilir ve sağ klik yapılarak properties ine girilir. Toolbara sonradan resim yerleştirmek içinde bir tane imagelist konulur.

### **ImageList**

Toolbar üzerine yerleştirilecek resimlerin listesi eklenir.

Button sekmesine geçilir.

### **InsertButton**

Toolbar üzerine buton eklemek için kullanılır.

### **RemoveButton**

Toolbar üzerinden buton silmek için kullanılır.

### **Index**

Toolbar üzerindeki butonları temsil eder.

### **Key**

Toolbar üzerinde hangi butona basıldığını anlamak için buraya name yazılır.

### **Style**

Toolbar üzerindeki butonun özelliğini belirtir.

- |   |  |
|---|--|
| 0 | Normal   |
| 1 | Check Butonu   |
| 2 | Option Buton   |
| 3 | Seperatör Araya boşluk koyar. (Frame gibi davranır)                |
| 4 | Seperatör gibidir. Boşluk oranını biz verebiliriz.                 |
| 5 | Aşağı açılan menü<br>İnsert buton yaparak alt menüler oluşturulur. |

### **Wrappable**

Toolbar üzerindeki butonlar forma sığmadığı zaman düğmeler bir alt satırdan devam eder.

### **Statusbar**

Ekranın altına açıklama barı yerleştirmek için kullanılır.

#### **General sekmesinden.**

### **Style**

Status barın görünümünü belirler.

### **Style**

Status barın görünümünü belirler.

- |   |                    |
|---|--------------------|
| 0 | Çoklu panel        |
| 1 | Bir panel bulunur. |

### **SimpleText**

Style özelliği 1' ken barın üzerindeki yazıyı görmemizi sağlar.

#### **Panels Sekmesinden**

### **Index**

Status barın üzerine panel eklememize yarar max. 16 adet.

### **Text**

Status barın index numarasına göre panellere text yazmamızı sağlar.

### **Key**

Status barın panellerini index numarasına göre değilde ismine göre kullanmak istediğimizde kullanılır.

### **Style**

Status barın panellerine otomatik olarak komut koymak için kullanılır.

- |   |                 |
|---|-----------------|
| 0 | Text veya resim |
| 1 | Caps Lock tuşu  |
| 2 | Num Lock tuşu   |

- 3 İnsert tuşu
- 4 Scroll Lock tuşu
- 5 Saati gösterir.
- 6 Tarihi gösterir.

### ***Bevel***

Status barın panellerinin görünümünü belirler.

- 0 Düz
- 1 İçeri basık
- 2 Dışarıya doğru

### ***AutoSize***

Status barın panellerinin boyut görünümünü belirler.

- 0 Boyutlandırma yok
- 1 Form resize olduğunda paneller oluşan yeni mesafeyi paylaşır.
- 2 Paneller metinlerini gösterecek şekilde genişler.

### ***Picture***

Status barın panellerine resim koymak için kullanılır.

### ***Örnek 1***

```
Private Sub Combo1_Click()  
    RichTextBox1.SelFontName = Combo1.Text  
End Sub  
  
Private Sub Combo2_Click()  
    RichTextBox1.SelFontSize = Combo2.Text  
End Sub  
  
Private Sub Form_Load()  
    Dim i  
    For i = 0 To Screen.FontCount - 1  
        Combo1.AddItem Screen.Fonts(i)  
    Next  
    'Combo1.Sorted = True  
    Combo1.ListIndex = 5  
  
    For i = 6 To 72 Step 2  
        Combo2.AddItem (i)  
    Next  
    Combo2.ListIndex = 2  
    RichTextBox1.SelBullet = True  
End Sub  
  
Private Sub Toolbar1_ButtonClick(ByVal Button As MSComctlLib.Button)  
    Select Case Button.Key  
        Case "bld"  
            RichTextBox1.SelBold = Button.Value  
        Case "italik"  
            RichTextBox1.SelItalic = Button.Value  
        Case "altcizgi"  
            RichTextBox1.SelUnderline = Button.Value  
        Case "sol"  
            RichTextBox1.SelAlignment = 0  
        Case "orta"  
            RichTextBox1.SelAlignment = 2  
        Case "sag"  
            RichTextBox1.SelAlignment = 1  
    End Select  
End Sub
```

```

Private Sub Toolbar2_ButtonClick(ByVal Button As MSComctlLib.Button)
Select Case Button.Key
Case "ac"
CommonDialog1.FileName = "*.txt"
CommonDialog1.ShowOpen
Open CommonDialog1.FileName For Input As #1
RichTextBox1.TextRTF = ""
While Not EOF(1)
Line Input #1, a
RichTextBox1.Text = RichTextBox1.Text + a + Chr(13) + Chr(10)
Wend
Close #1

Case "kaydet"
CommonDialog1.Flags = 2
CommonDialog1.DefaultExt = ".txt"
CommonDialog1.ShowSave
Open CommonDialog1.FileName For Output As #1
a = RichTextBox1.Text
Print #1, a
Close #1

Case "kes"
Clipboard.SetText RichTextBox1.SelRTF
RichTextBox1.SelText = ""

Case "koya"
Clipboard.SetText RichTextBox1.SelRTF

Case "yap"
RichTextBox1.SelRTF = Clipboard.GetText

Case "yaz"
CommonDialog1.ShowPrinter
Printer.Print RichTextBox1.Text
Printer.EndDoc

End Select
End Sub

```

```

Private Sub rtfText_SelChange()
fMainForm.tbToolBar.Buttons("Bold").Value = IIf(rtfText.SelBold, tbrPressed, tbrUnpressed)
fMainForm.tbToolBar.Buttons("Italic").Value = IIf(rtfText.SelItalic, tbrPressed, tbrUnpressed)
fMainForm.tbToolBar.Buttons("Underline").Value = IIf(rtfText.SelUnderline, tbrPressed, tbrUnpressed)
fMainForm.tbToolBar.Buttons("Align Left").Value = IIf(rtfText.SelAlignment = rtfLeft, tbrPressed, tbrUnpressed)
fMainForm.tbToolBar.Buttons("Center").Value = IIf(rtfText.SelAlignment = rtfCenter, tbrPressed, tbrUnpressed)
fMainForm.tbToolBar.Buttons("Align Right").Value = IIf(rtfText.SelAlignment = rtfRight, tbrPressed, tbrUnpressed)
End Sub

```

## **Veri Tabanı Uygulamaları**

### **Visual Data Manager**

Add-Ins den Visual Data Manager seçilir.

New – Microsoft Access – Version 7.0 MDB

Bir isim verilir. **(üye takibi)**

Properties üzerindeyken sağ klik yapılır ve New Table seçilir.

**Table Name : (Bilgi\_Girisi)**

**Add Field** seçilerek tabloya girecek veriler girilir, **Remove Field** seçilerek tablodan veri silinir. ve **Built The Table** basılır.

<b>Name</b>	:	`verilere girilecek isim
<b>Type</b>	:	`verilerin typeleri
<b>Size</b>	:	`verinin uzunluğu kaç verilirse o sayıdan fazla bilgi girilemez
<b>FixedField</b>	:	`seçili olursa random dosyada olduğu gibi her veriye aynı uzunluk ayırır.
<b>VariableField</b>	:	`seçili olursa veriler kendi uzunluklarını alır.
<b>AutoIncField</b>	:	`seçili ise sayılar otomatik olarak artar.
<b>AllowZeroLength</b>	:	`seçili ise herhangi bir veri girilmezse bile işlem devam eder.
<b>Required</b>	:	`seçili ise herhangi bir veri girilmezse işlem devam etmez. Kullanıcıyı zorlar.
<b>OrdionalPosition</b>	:	`veri alanı sırası girilir.
<b>ValidationText</b>	:	`veri alanına girilecek bilgi hakkında ValidationRule `ye göre açıklama yazılır.
<b>ValidationRule</b>	:	`veri alanına girilebilecek yöntem girilir. Text kutusundaki Validate olayı.
<b>DefaultValue</b>	:	`İnputbox'daki varsayım değeri gibi çalışır.

Field

Name	Tupe	Size
uyeno	İnteger	2
Ad	Text	25
soyad	Text	30
Uye_turu	Text	15
unvan	Text	10
telefon	Text	15
Faks	Text	10
Email	Text	30
adres	Memo(32Kb bilgi )	0
sehir	Text	20
pkodu	long	4

**Add Index** tablolar arasında ilişki kurmak, tablomuzda arama yaptırmak için kullanılır ve sadece bir tanesi birincil anahtar (**Primary**) olabilir. Ayrıca bir veri üzerinde aynı veriden olmasını istemiyorsanız (örneğin aynı üye numarasından olmasını istemiyorsak) **Unique** işaretlenir

### Otomatik Form Hazırlama

Utility sekmesinden Data Form Designer seçilir. Record Source sekmesinden table seçilir ve From Name bölümüne de bir isim verilir. Built the Form seçilerek tablo oluşturulur.

### Data Environment ile Form Hazırlama

Add menüsünden More ActiveX Designers, Data Environment seçeneği ile ekrana bir tane Data Environment eklenir. (Eğer gözükmüyorsa Project menüsünden references den seçilir.) Connection1 üzerindeyken sağ klik yapılarak Properties ine girilir. Microsoft 3.51 veya 4.0 seçilir. Next'e basılır. 1 numaralı bölümden mdb dosyamız bulunur ve güvenli olması için Test Connection yapılır. Tamam diyerek oradan çıkılır. View menüsünden Data View Window açılır. Data View Window üzerinde istediğimiz table ları Data Environment mouse yardımıyla taşıyoruz ve Data View Window kapatılır. Data Environment üzerinden istediğimiz elemanlar bu form üzerine yerleştirilir.

```
Private Sub Onceki_Click()
    If DataEnvironment1.rsBilgiGirisi.BOF Then
        MsgBox "İlk kayıt"
    Else
        DataEnvironment1.rsBilgiGirisi.MovePrevious
    End If
End Sub
```

```
Private Sub Sonraki_Click()
    If DataEnvironment1.rsBilgiGirisi.EOF Then
        MsgBox "İlk kayıt"
    Else
        DataEnvironment1.rsBilgiGirisi.MoveNext
    End If
End Sub
```

### **Veri Tabanları İşlemleri**

Data1.Recordset.MoveNext	Sonraki kayda götürür.
Data1.Recordset.MovePrevious	Önceki kayda götürür.
Data1.Recordset.AddNew	Yeni kayıt oluşturur.
Data1.Recordset.MoveFirst	İlk kayıta götürür.
Data1.Recordset.MoveLast	Son kayıta götürür.
Data1.Recordset.BOF	İlk kayıt üzerinde olduğunuzu belirtir.
Data1.Recordset.EOF	Son kayıt üzerinde olduğunuzu belirtir.
Data1.Recordset.AbsolutePosition	O anda hangi kayıt üzerinde olduğunuzu belirtir.
Data1.Recordset.Bookmark	Mevcut kaydın data içerisindeki yerini belirler.
Data1.Recordset.LastModified	Üzerinde değişiklik yapılan son kayda gider.
Data1.Recordset.Close	Veri dosyasını kapatmak için kullanılır.
Data1.Recordset.Delete	Veri dosyasından bilgi silmek için kullanılır.
Data1.Recordset.EditMode (Bir kaydın düzenlenişini belirler)	DbEditAdd : Yeni bir kayıt üzerinde işlem yapılıyorsa DbEditInProgress : Kullanıcı bir kayıt üzerinde değişiklik yapıyorsa
Data1.UpdateControls	Yenilemek istediğiniz kayıt.
Data1.UpdateRecord	Bir önceki kayıta yapılan değişiklikleri kaydeder.
Data1.Recordset.CancelUpdate	Yenilemek istediğiniz kaydı iptal eder.
Data1.ReadOnly = True	Data dosyası ReadOnly olur.
Data1.Recordset.Seek	Arama yapılacak bölümlerdeki şart belirtilir.
Data1.Recordset.RecordCount	Toplam kayıt sayısını verir.
Data1.Recordset.Update	Kayıda bilgi ekler.
Data1.Recordset.NoMatch	Arama sonucunda aranan kayıt bulunamazsa True olur.
Data1.Recordset.Index	Arama yapılırken başlıklarda kullanılır.
Data1.Recordset.Fields(index)	Herhangi bir yerden kayıtlı verilere ulaşmak için kullanılır.
Data1.Refresh	Data dosyasının yeniden açılmasını sağlar
Data1.Exclusive = True	Açılmış olan bir dataya başka kullanıcıların erişimi engellenir

Öncelikle form üzerine bir tane **Data** komponenti yerleştirilir ve **DataBaseName** bölümüne hazırladığımız **üye takibi** 'ni yazarız. **RecordSource** bölümüne ise **Bilgi\_Girisi** seçeriz. Artık komponentleri ekrana yerleştirebiliriz.

11 Label ve 11 TextBox, 9 Command Buton yerleştirilir.

Textlere sırasıyla **DataSource** bölümüne **Data1** aktarılır.  
Textlere sırasıyla **DataField** bölümleri doldurulur.

```
Private Sub cikis_Click()
    Data1.Recordset.Close
    Unload Me
End Sub
```

```
Private Sub Data1_Reposition()
    Data1.Caption = Str(Data1.Recordset.AbsolutePosition + 1) + "/" + Str(Data1.Recordset.RecordCount)
End Sub
```

```
Private Sub ikayit_Click()
    Data1.Recordset.MoveFirst
End Sub
```

```
Private Sub iptal_Click()
    Data1.UpdateControls
    If Data1.Recordset.EditMode = dbEditAdd Then
        Data1.Recordset.CancelUpdate
    End If
End Sub
```

```
Private Sub kaydet_Click()
    On Local Error Resume Next 'Hata olursa hatanın olduğu satırdan sonraki satıra geçer.
    Data1.UpdateRecord
    Data1.Recordset.Bookmark = Data1.Recordset.LastModified
```

```
' Data1.Recordset.Update
' Data1.Recordset.MoveLast
End Sub
```

```

Private Sub kayitsil_Click()
On Local Error Resume Next
If Data1.Recordset.RecordCount > 0 Then
    If MsgBox("Kaydı silmek istediğinizden emin misiniz?", vbYesNo + vbDefaultButton2) = vbYes Then
        If Data1.Recordset.AbsolutePosition = 0 Then
            Data1.Recordset.Delete
            Data1.Recordset.MoveNext
        Else
            Data1.Recordset.Delete
            Data1.Recordset.MovePrevious
        End If
    Else
        Exit Sub
    End If
Else
    MsgBox ("Dosyada kayıt yok")
End If
End Sub

```

```

Private Sub onkayit_Click()
On Local Error Resume Next
    Data1.Recordset.MovePrevious
If Data1.Recordset.BOF Then
    MsgBox "İlk kayıt üzerindesiniz"
    Data1.Recordset.MoveFirst
End If
End Sub

```

```

Private Sub sonkayit_Click()
On Local Error Resume Next
    Data1.Recordset.MoveLast
End Sub

```

```

Private Sub sonrakikayit_Click()
On Local Error Resume Next
    Data1.Recordset.MoveNext
If Data1.Recordset.EOF Then
    MsgBox "Son kayıttasınız"
    Data1.Recordset.MoveLast
End If
End Sub

```

```

Private Sub yenikayit_Click()
On Local Error Resume Next
    Data1.Recordset.AddNew
    Text1.SetFocus
End Sub

```

## **DBGrid**

Ekrana bir tane DBGrid yerleştirilir ve DataSource: Data1 yapılır.

### **AllowAddNew**

Tablo üzerinde yeni kayıt düğmesini aktifleştirir.

### **AllowArrows**

Tablo üzerinde yön tuşlarını aktifleştirir.

### **AllowDelete**

Tablo üzerinde silme işlemi yapılabilir.

### **AllowUpdate**

Tablo üzerinde yenileme işlemi yapılabilir.

İstenirse DBGrid de birkaç bölüm gösterilebilir. Bunun için sağ klik yapılarak Properties'e girilir ve oradan da Columns sekmesine geçilir ve istediğiniz kolonlar ayarlanır. Default olarak 2 kolon gelmektedir istenirse arttırılabilir. Bunun için sağ klik yapılır ve Edit yapılır, sonra yine sağ klik ve Append veya Insert yapılarak kolon eklenir.

### **ADO ile Veri Tabanı**

Microsoft ADO Data Control 6.0 (MSADODC.OCX)

Data bileşenine benzeyen yeni bir komponentdir.

Adodc üzerindeyken sağ klik yapılır ve propertiese girilir. Use Connection String seçiliyken Build seçeneğinden mdb dosyamız seçilir. RecordSource sekmesinden Command Type bölümünden 2-adCmdTable seçilir. Altta açılan menüden Table adı seçilir ve Tamam'a basılır.

Form üzerine gerekli komponentler yerleştirilir.

Komponentlerin DataSource bölümüne Adodc1 yazılır.

### **DataGrid**

Microsoft DataGrid Control 6.0 (MSDATGRD.OCX)

Ekranı bir tane DataGrid yerleştirilir ve DataSource: Adodc1 yapılır. DataGrid üzerinde istediğimiz gibi ayarlamalar yapabiliriz. Bunun için datagrid üzerindeyken sağ klik yapılır ve Retrieve Fields seçilir. Tekrar sağ klik yapılır ve Edit seçilir.

### **AllowAddNew**

Tablo üzerinde yeni kayıt düğmesini aktifleştirir.

### **AllowArrows**

Tablo üzerinde yön tuşlarını aktifleştirir.

### **AllowDelete**

Tablo üzerinde silme işlemi yapılabilir.

### **AllowUpdate**

Tablo üzerinde yenileme işlemi yapılabilir.

### **ColumnHeaders**

Tablo üzerindeki isimleri gizler.

### **DefColWidth**

Tablo üzerindeki sütunların genişlikleri ayarlanır.

### **HeadLines**

Tablo üzerindeki başlıklara ayrılacak satır sayısı.

### **RowHeight**

Tablo üzerindeki satırların yükseklikleri ayarlanır.

### **RowDviderStyle**

Tablo üzerindeki satırlar çizgilerini biçimlendirir.

### **TabAcrossSplits**

Tablo üzerinde bir split uygulanmışsa tabla bu split içine gidilebilir.

### **WrapCellPointer**

Tablo üzerinde yön tuşlarıyla hareket ederken satır bitince otomatik olarak bir üst veya alt satıra imleci götürür.

### **TabAction**

Tablo üzerinde tab tuşunun rolünü ayarlar.

- |   |   |
|---|---|
| 0 | Tab tuşu grid üzerinde aktif olmaz.   |
| 1 | Tab tuşu grid üzerinde sadece kolonlar arasında geçer. Kolonlar bitince bir üst komponente geçer. |
| 2 | Tab tuşu sadece grid üzerinde geçerlidir.   |

### **Locked**

Tablo üzerinde işaretlenen kolona girilemez.

### **AllowSizing**

Tablo üzerinde işaretli sütunun genişliği fare ile ayarlanabilir.

### **Button**

Tablo üzerinde işaretli kolon üzerinde aşağıya doğru açılan bir ok çıkar.

### **DividerStyle**

Tablo üzerinde işaretli kolon üzerindeki çizgileri değiştirir.

### **Split Locked**

Tablo üzerinde split olan bölüme bilgi girişi yapılamaz.

### **Split AllowFocus**

Tablo üzerinde split olan bölüme geçiş yapılabilir.

### **Split AllowSizing**

Tablo üzerinde split olan bölümün genişliği ayarlanabilir.

### **Split AllowRowSizing**

Tablo üzerinde split olan bölümün satır genişlikleri ayarlanabilir.

### **Split RecordSelectors**

Tablo üzerinde split olan bölümün başındaki satırı seçme bölümü kaldırılır.

### **Format**

Tablo üzerinde kolonlara uygulanacak formatlar ayarlanır.

### **Data Report**

Project menüsünden Add Data Report eklenir

### **RptLabel**

Rapor başlıklarında kullanılır.

### **RptTextBox**

Bilgi girişlerini göstermek için kullanılır.

### **RptImage**

Rapora resim eklemek için kullanılır.

### **RptLine**

Rapor üzerine çizgi koymak için kullanılır.

### **RptShape**

Rapor üzerine şekilleri çizmek için kullanılır.

### **RptFunction**

Rapor içinde üst veya alt bilgi bölümlerinde kullanılır.

Form üzerine bir tane DataEnvironment yerleştirilir, mdb dosyası açılır. Data View açılır ve istediğiniz Table DataEnvironment üzerine mouse yardımıyla yerleştirilir. DataEnvironment üzerinden istediğiniz komponentler Rapor üzerine yerleştirilir. Komponentler üzerinde değişiklik yapmak için sağdaki properties seçeneğinden veya sağ klik yapılarak uygulanabilir.

```
Private Sub Command2_Click()  
    DataReport1.Show  
End Sub
```

### **Raporlara özel komutlar eklemek**

Rapor içinde sağ klik yapılarak açılan menüden Insert Control seçeneğinden

%p	Aktif sayfanın numarası
%P	Toplam sayfa sayısı
%d	Kısa formatlı günün tarihi
%D	Uzun formatlı günün tarihi
%t	Sistem saati kısa formatlı
%T	Sistem saati uzun formatlı
%i	Raporun başlığı

### **Raporlara hesaplamalar eklemek**

Rapor içinde rptFunction eklenir.

0	rptFuncSum	Toplamı bulur,
1	rptFuncAve	Ortalamayı bulur,
2	rptFuncMin	En küçük değeri bulur,
3	rptFuncMax	En büyük değeri bulur,
4	rptFuncRCnt	Satır sayısını bulur,
5	rptFuncVCnt	Boş olmayan alan sayısını bulur,
6	rptFuncSDEV	Standart sapmayı bulur,
7	rptFuncSERR	Standart hatayı bulur.

### **Like**

Like komutu ile yazdığımız bir Text in neresinde olursa olsun bir yazıyı bulmak için kullanılır.

### **ÖRNEK 1**

```
Private Sub Command1_Click()  
If Text1 Like "*"ISTANBUL*" Then 'like komutu text içinde arama yaptırır
```

```
    '* * yazinin neresinde olursa olsun bulur.  
    '* yazinin sonunda olursa bulur.  
    ' * yazinin başında olursa bulur.
```

' ÖNEMLİ = arama büyük küçük harfe duyarlıdır. Çünkü arama ASCII kodlarına göre yapar.

```
MsgBox ("Bu arkadaşın İstanbul'da ikamet ettiği tespit edildi")
Else
MsgBox ("Bu arkadaş İstanbul' lu değil")
End If
End Sub
```

## **ÖRNEK 2**

```
Private Sub Command2_Click()
' ??İstanbul dersek İstanbul'un önündeki 2 harfi anlatır.
' ## İstanbul dersek İstanbul'un önündeki 2 rakamı anlatır.
' "[a-g]İstanbul" dersek İstanbul'un önünde a-g harfi varmı diye ararır.

'örnek aistanbul, eistanbul gibi

If Text1 Like "[a-g]İSTANBUL" Then

MsgBox ("Bu arkadaşın İstanbul'da ikamet ettiği tespit edildi")

Else

MsgBox ("Bu arkadaş İstanbullu değil")

End If : End Sub
```

## **SQL Komutları**

Yeni bir mdb oluşturulur.

Name: uye  
TableName : giris

ÜyeNo  
Adı  
Soyadı  
dogtarihi  
Sehir  
SehirKodu  
Telefon  
Maas `long

En çok kullanılan sql komutları.

Form üzerine bir tane Adodc1 ve DataGrid1 yerleştirilir. Adodc1'e mdb dosyası bağlanır. RecordSource : 8 yapılır ve Command Text bölümüne : **select \* from mdb** yazılır. Oluşturduğumuz DataGrid1.DataSource : Adodc1 yapılır.

### **Select**

Tablo içerisinden bilgi seçmek için kullanılır.

1.  
Adodc1.RecordSource = "select \* from giris"  
Adodc1.Refresh
2.  
Adodc1.RecordSource = "select uyeno, ad, soyad from giris"  
Adodc1.Refresh

### **Where**

Tablo içerisinden sorgulama kriterini, koşul belirtmeyi sağlar.

ComboBox oluşturulur ve içerisinde oluşturduğumuz datalar eklenir. İki tane OptionButton eklenir ve adları asc ve desc yapılır.

```
Private Sub Combo1_Change()  
Dim a  
If Option1.Value = True Then  
    a = "asc"  
Else  
    a = "desc"  
End If  
Adodc1.Recordset.Sort = Combo1.Text & " " & a  
End Sub
```

1.  
Adodc1.RecordSource = "select \* from giris where (ad like '%o%') " **Sadece adı O harfi ile başlayan kişiler gösterilir.**  
Adodc1.Refresh
2.  
Adodc1.RecordSource = "select \* from giris where ad = 'oguz'" **Sadece adı oguz olanlar listelenir.**  
Adodc1.Refresh
- 2.1.  
Dim a  
a = InputBox("Sorgulama yapmak istediğiniz harfi giriniz")  
Adodc1.RecordSource = "select \* from giris where (ad like '&a &'^)" **Girdiğiniz harfle başlayanlar listelenir.**  
Adodc1.Refresh
3.  
Adodc1.RecordSource = "select \* from giris where ad = 'oguz' and sehir='İstanbul'" **sadece istanbulda oturan oğuz adlı kişiler listelenir.**  
Adodc1.Refresh
4.  
Adodc1.RecordSource = "select \* from giris where ad = 'oguz' or sehir='İstanbul'" **sadece istanbulda oturanlar veya ismi oğuz olanlar listelenir.**  
Adodc1.Refresh
5.  
Adodc1.RecordSource = "select \* from giris where dogtarihi >= #01/01/73# " **sadece doğum tarihi 01/01/73 tarihinden büyük olanlar listelenir.**  
Adodc1.Refresh
6.  
Adodc1.RecordSource = "select \* from giris where maas > 250000000" **sadece maaşı 25000000 TL üzerinde olanlar listelenir.**  
Adodc1.Refresh
- 6.1.  
Dim a  
a = InputBox("Miktarı giriniz")  
Adodc1.RecordSource = "select \* from giris where maas <" & a **Maaşı girdiğiniz miktardan küçük olanlar listelenir.**  
Adodc1.Refresh
7.  
Adodc1.RecordSource = "select distinct ad, soyad from giris " **sadece ad, soyad gösterilir ve aynı kayıtlar gösterilmez**  
Adodc1.Refresh
8.  
Adodc1.RecordSource = "select ad, soyad, sehir, (sehkodu\*2) as [yeni] from giris order by ad desc"  
Adodc1.Refresh **Buradaki (sehkodu\*2) sehir kodunu 2 ile çarpar ve as [yeni] adlı bir sütün açarak oraya yazar.**
9.  
Adodc1.RecordSource = "select top 3 \* from giris where sehkodu='212'" **Burada sehirkodu 212 olan ve üst sırada bulunan 3 kayıt gösterilir.**  
Adodc1.Refresh
10.  
Adodc1.RecordSource = "select count(ad) from giris "  
Adodc1.Refresh **ad bölümünde bulunan eleman sayısını verir.**
11.  
Adodc1.RecordSource = "select max(sehkodu) from giris " **ilgili elemanın en büyük sayısını bulur.**  
Adodc1.Refresh

12.  
Adodc1.RecordSource = "select min(sehkodu) from giris " **ilgili elemanın en küçük sayısını bulur.**  
Adodc1.Refresh
13.  
Adodc1.RecordSource = "select avg(sehkodu) from giris " **ilgili elemanın ortalamasını bulur.**  
Adodc1.Refresh
14.  
Adodc1.RecordSource = "select sum(sehkodu) from giris " **ilgili elemanın toplamını bulur.**  
Adodc1.Refresh
15.  
Adodc1.RecordSource = "select ad, maas from giris group by [ad]" **ada göre grup yapar ve maaşları toplar.**  
Adodc1.Refresh
16.  
Adodc1.RecordSource = "select ad, sum(maas) from giris group by [ad] having avg(maas) < 250000000"  
Adodc1.Refresh **ada göre grup yapar ve oluşturulacak grubu maaşı < 250000000 olanları gösterir.**
17.  
Adodc1.Recordset.Sort = "ad asc" **ad'a göre sort yapar.**
18.  
Dim a  
a = InputBox("Sıralama yapmak istediğiniz alan adını giriniz")  
Adodc1.Recordset.Sort = a & " asc" **istediğiniz bir alana göre sıralama yaptırabilirsiniz.**
19.  
Dim a, b  
a = InputBox("Sıralama yapmak istediğiniz alan adını giriniz")  
b = InputBox("Sıralama şeklini giriniz", , "asc")  
Adodc1.Recordset.Sort = a & " " & b & " "
20.  
Dim a  
a = InputBox("Sıralama yapmak istediğiniz alan adını giriniz")  
Adodc1.Recordset.Filter = a & "='oguz'" **istediğiniz bir alanı filtrelemek için kullanılır.**
21.  
Dim a  
If KeyAscii = 13 Then  
a = InputBox("Sıralama yapmak istediğiniz alan adını giriniz")  
Adodc1.Recordset.Filter = a & "='" & Text2.Text & "'" & ""  
End If **istediğiniz bir alanı textbox kullanarak filtrelemek için kullanılır.**
22.  
Private Sub Text3\_Change()  
If Len(Text3.Text) <> 0 Then  
Adodc1.Recordset.Filter = "ad like '" & Text3.Text & "%'"  
Else  
Adodc1.Recordset.Filter = adFilterNone  
Adodc1.Refresh **Bir text kutusuna harf yazdıkça filitreleme yapar.**  
End If  
End Sub

### **Order By**

Tablo içerisinden bilgileri sıralı şekilde göstermek için kullanılır.

Asc A-Z sıralı  
Desc Z-A sıralı

### **Distinct**

Tablo içerisinden aynı kayıtları göstermez.

### **Distinctrow**

Tablo içerisinden aynı kayıtları gösterir.

### **Group By**

Tablo içerisinden seçili veriler arası gruplandırma yapar.

### **Between**

Tablo içerisinde şartın iki aralıkta olup olmadığını kontrol eder.

### **Like**

Tablo içerisinde belirlenen kriterlere uygun bilgileri seçmek için kullanılır.

### **Having**

Tablo içerisinde gruplar arası şart belirler.

### **Formlar**

#### **AutoRedraw**

Form üzerinde bulunan çizimlerin ve yazıların form üzerine başka bir form geldiğinde veya form kapatıldığı zaman bil bozulmamasını sağlar.

#### **BorderStyle**

- |   |  |
|---|--|
| 0 | Formun boyutları değiştirilemez.                 |
| 1 | Boyutları değiştirilemeyen ama taşınabilir form. |
| 2 | Default değeri.                                  |
| 3 | 1 e benzer                                       |
| 4 | Başlığı küçük olan form.                         |
| 5 | 4 e benzer ama boyutlandırılabilir.              |

#### **ControlBox**

Form üzerinde bulunan sol taraftaki ikon üzerine basınca çıkan control kutusunun açılması sağlar.

#### **Icon**

Form üzerinde bulunan ikonun değişmesi için kullanılır.

#### **KeyPreview**

Form üzerinde klavyenin kullanılması için True yapılır.

#### **MaxButton**

Form üzerinde bulunan max butonu için kullanılır.

#### **MinButton**

Form üzerinde bulunan min butonu için kullanılır.

#### **MDIChild**

Form üzerinde eğer bir MDI form olursa true yaparak bu formu onun üyesi yapar.

#### **Moveable**

Form un kullanıcı tarafında taşınmaması için kullanılır.

#### **ShowInTaskbar**

Formu minimize yapınca görev çubuğunda göstermez.

#### **StartPosition**

Formun başlangıçta ekranın neresinde olacağını belirler.

- |   |  |
|---|--|
| 0 | Manual   |
| 1 | Form içinde bulunduğu formun ortasında (MDI formlarda) |
| 2 | Ekranın ortasında                                      |

3 Devamlı kaydırarak

### **WindowState**

0	Normal
1	Minimize
2	Maximize

### **MDIFormlar**

Add Form dan MDI form seçilir.

### **AutoRedraw**

Form üzerinde bulunan çizimlerin ve yazıların form üzerine başka bir form geldiğinde veya form kapatıldığı zaman bil

### **Arrange**

MDIForm içerisinde bulunan formların görünüm özelliklerini değiştirir.

0	Basamakla
1	Yatay
2	Dikey

### **WindowList**

MDIForm içinde açık bulunan formları otomatik olarak gösterilmesini sağlar.