

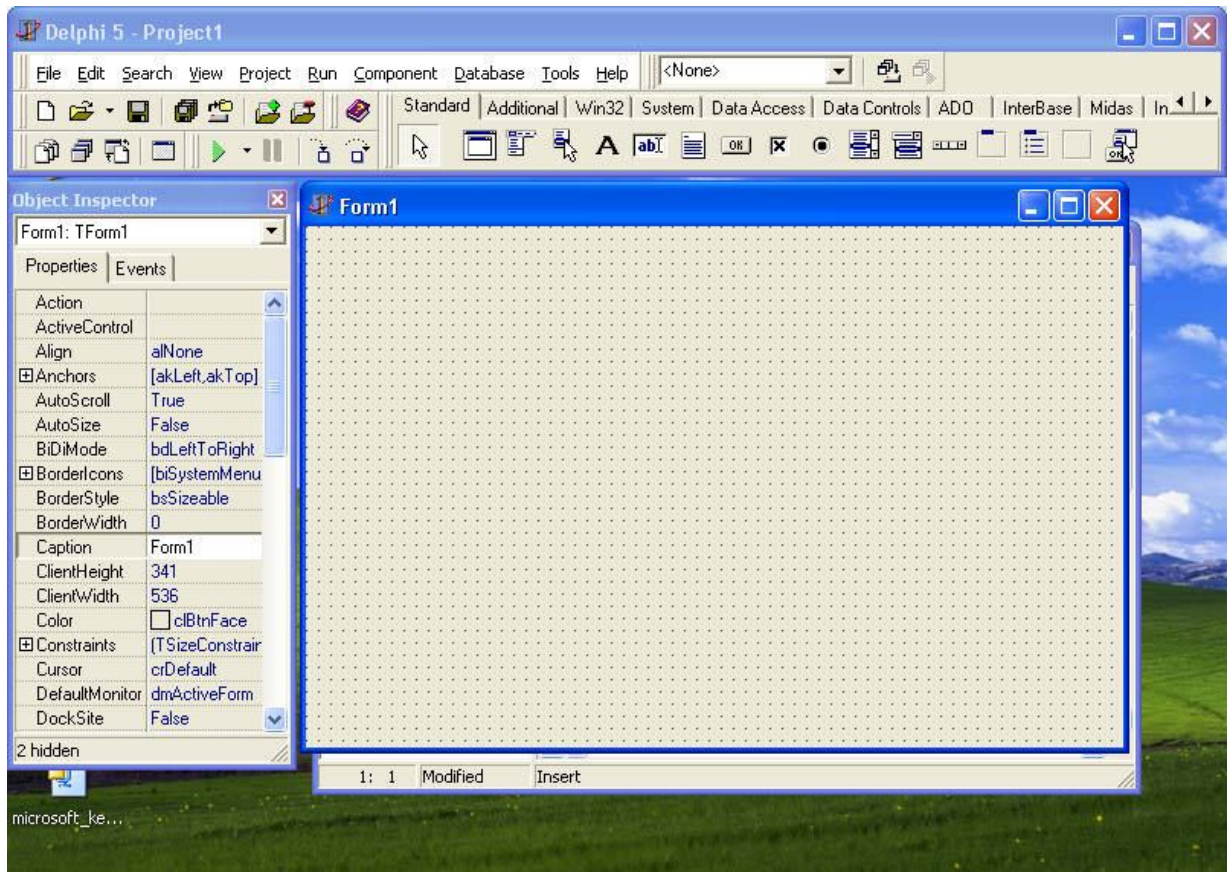
## DELPHI 5.0

### Açıklama:

Herkese merhaba, programcı olmak ve bu işe hevesli olan herkes için kolay anlaşılır delphi derslerine hoşgeldiniz... Sizlerle birlikte delphi dünyasına uzun bir yolculuğa çıkacağız. :-)) Uygulamalı olan derslerimizle çok kolay bir şekilde delphiyi kavrayabileceksiniz... Lafı fazla uzatmadan derslerimize geçelim...

### ADIM ADIM DELPHI - 1

Delphi'ye ilk girdiğinizde aşağıdaki gibi ekranla karşılaşacaksınız. Program 3 ana pencereden meydana geliyor. Form, Object Inspector (Nesne Denetleyicisi) ve Kontrol Paneli. **Form1** başlıklı pencere bizim programımızın anasayfası oluyor. Yani çalışmalarımızı bu form üzerinde gerçekleştireceğiz. Object Inspector penceresi form ve bileşen ayarlarını değiştirebileceğimiz bölümdür. Kendi arasında "**Properties**" ve "**Events**" diye ikiye ayrılır. Bizim için gerekli olan fazlaca kullanacağımız bölüm "**Properties**" penceresidir...



Bizim için gerekli olan bazı özellikleri kısaca açıklayalım...

**Caption:** Programımızın başlığını (adını) ve diğer bileşenlerin isimlerini buradan değiştirebilirsiniz. Varsayım olarak "Form1" yazar fakat bunu silip kendi programımızın ismini yazabilirsiniz...

**Color:** Programımızın zemin renginin değiştirilmesini sağlar.

**BorderIcons:** Programımızın sağ üst köşesinde bulunan "Simge Durumuna Küçült", "Ekranı Kapla", "Kapat" simgelerini buradan kaldırabilir veya aktif yapabilirsiniz. Yanındaki '+'ya basarsanız karşınıza seçenekler çıkacaktır.

---**Minimize:** Simge Durumuna Küçült

---**Maximize:** Ekranı Kapla

True olanlar şu an aktif, False olanlar ise kapalı anlamına gelmektedir...

**BorderStyle+:** Programın boyutlarının değiştirilmesinin serbest yada kapalı olması ayarlarıdır... Normalde "bsSizeable"dir. Bu, programın boyutunu herkes istediği gibi ayarlayabilir anlamına gelmektedir. (Örneğin mouse'u programımızda sağ alt köşeye götürürsek, bir ok çıkar ve istediğimiz gibi genişletip daraltabiliriz). Eğer boyutu

sabitlemek ve başkalarının değiştirmesini engellemek için "bsSingle" seçeneğini kullanınız. Geri kalanları kullanmaya gerek olmadığı için açıklamıyorum...

**Font+:** Program (form) üzerindeki yazıların şeklini, rengini, boyutunu ayarlayabilirsiniz...

**Auto Size:** Programın boyutlarını kendi otomatik olarak ayarlamaktadır. Fakat kullanmanızı pek tavsiye etmem. Ama kullanmak için "True" yapabilirsiniz...

**Enabled:** Formda bulunan bir bileşeni (bileşenler program üzerine eklenen butonlar, metinler gibi şeylerin genel adı olarak nitelendirilmiştir) aktif yada kullanılmaz yapar. Bizim işimize yaramayacak... :-)

**Constraints+:** Programın boyutlarını sayı cinsinden ayarlamamızı sağlar. Pek de kullanışlı değildir. Elle ayarlamak daha kolay...

Geri kalan özelliklere yeri geldiği zaman tekrar değineceğiz. Şimdi gelelim "Events"e. buradaki özellikler direkt olarak bileşenlerle ilgili olduğu için fazla bi açıklamaya gerek yok ama ben size kısaca en önemlilerini açıklayayım.

Örneğin form (programımız) üzerinde bir buton var.

**OnClick:** Butona bir kez tıklamayla başlatılacak uygulama buradan ayarlanır...

**OnDbClick:** Buton yada herhangi bir düğmeye iki kez tıklamayla başlatılacak olan uygulama buradan ayarlanır..

Mesala butona tıklayınca başka bir pencere (form) açılsın. Butona "OnClick" komutuyla gerekli kodu verirse pencere bir kez tıklamayla açılır. Ama "OnDbClick" komutuyla gerekli kodu verirse pencere iki kez tıklamayla açılır...

## Bileşenler (Hani hep bahsederdik :-)



Bileşenler form'a eklenen butonları ve bizim için gerekli olan bir çok aracı barındırmaktadır. Bileşenler gördüğünüz gibi 18 sayfadan oluşuyor (Standard, Additional, Win32, System ... vs vs.) Programcılar genellikle en çok kullandığı bölümler ilk 3 bölümdür. Tabii diğerleri de yeri geldiğinde kullanılmaktadır... Form'a (programımıza) yeni bileşen eklemek için kontrol panelinden istediğiniz bir bileşene tıklayın ve sonra da form üzerinde herhangi bir boş yere tıklayın. Bileşen form üzerine yerleştirilecektir... Size tavsiyem sırasıyla tüm bileşenleri form üzerine yerleştirin. Neyin ne olduğunu daha iyi anlarsınız... Bileşenin adını öğrenmek için mouse'u bileşenin üstüne götürerek bekleyiniz. Böylece bileşenlerin adlarını da öğrenmiş olacaksınız... Bazı bileşenleri size açıklayacağım...

**Button:** Bu bileşen sayesinde form (programımız) üzerine düğmeler yerleştirebilir ve bu düğmelere programın çalışması sırasında basıldığı zaman gerçekleştirilmesini istediğimiz bileşenleri bağlayabiliriz... Programcılar tarafından çok kullanılır ve her programda mutlaka bir butona rastlarsınız... Form'a bir buton eklediğinize buton üzerindeki isim "Button1" olarak geçer. Biz bunu "**Object Inspector**" penceresindeki **Caption** bölümünden değiştirebiliriz.

**Label:** Form üzerine sadece yazı yazmak için kullanılan bir bileşendir. Label'i eklediğiniz takdirde "Object Inspector" penceresindeki "Caption" bölümüne istediğiniz yazıyı girebilirsiniz...

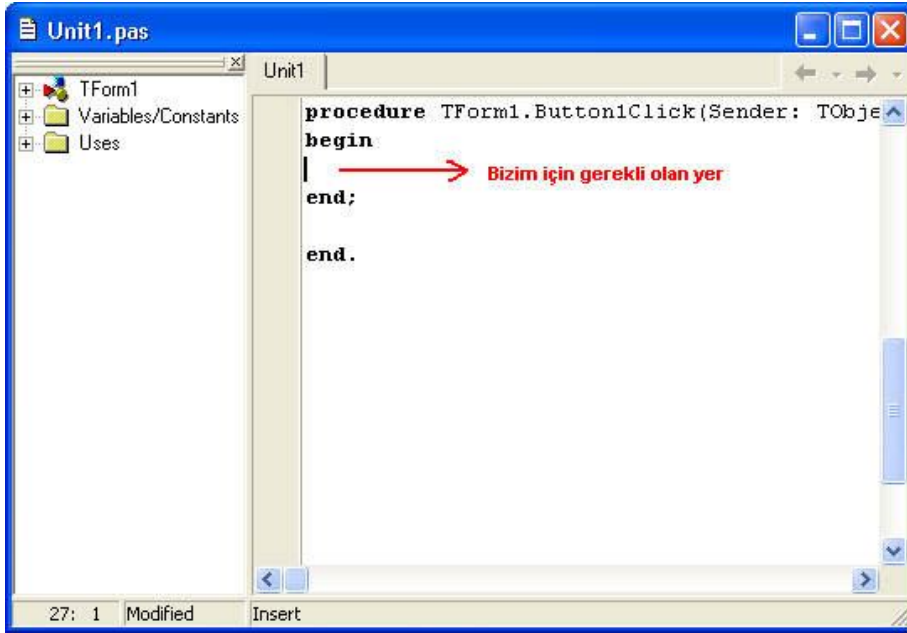
**Image:** "Additional" bölümünde bulunan Image bileşenini kullanarak form üzerine her hangi bir resim dosyası yada ikon (.ico) dosyası koymanız mümkündür. Bir resim koymak için Image bileşenine tıklayın ve form üzerine gelerek boş bir yere tıklayın. Şeffaf bir kare şekli belirecektir. Onun üstüne iki kez tıklayın ve gelen pencerede "Load" ı tıklayarak karşınıza çıkan pencereden istediğiniz resim dosyasını seçin. Resminiz form üzerine yerleştirilecektir... Resmi sürükleyerekte konumun değiştirebilirsiniz...

**Panel:** Bu bileşen form üzerine yerleştirdiğimiz bileşenlerin kullanıcıya daha derli toplu gösterilmesi ve uygulama geliştirme aşamasında konuyla ilgili bileşenlerin bir araya toplanarak daha kolay kontrol edilebilmesi için kullanılır. Uygulamaları geliştirirken bileşenlerimizi form üzerine direkt olarak yerleştirmektense, paneller üzerine yerleştirmemiz, daha sonraki düzeltmelerde bize büyük kolaylık sağlar. Örneğin uygulamalarımız ile ilgili birçok düğmemiz varsa bu düğmelerin tümünü bir panel üzerine, kullanıcının bilgi girişi yapacağı diğer bileşenler ise bir başka panel üzerine yerleştirilebilir...

## ADIM ADIM DELPHI – 2

Artık yavaş yavaş uygulamalarımıza geçeceğiz. Size ilk önce en basitinden form üzerine eklenen bir butondan programın kapatılmasını sağlayan bir uygulama yapacağız...

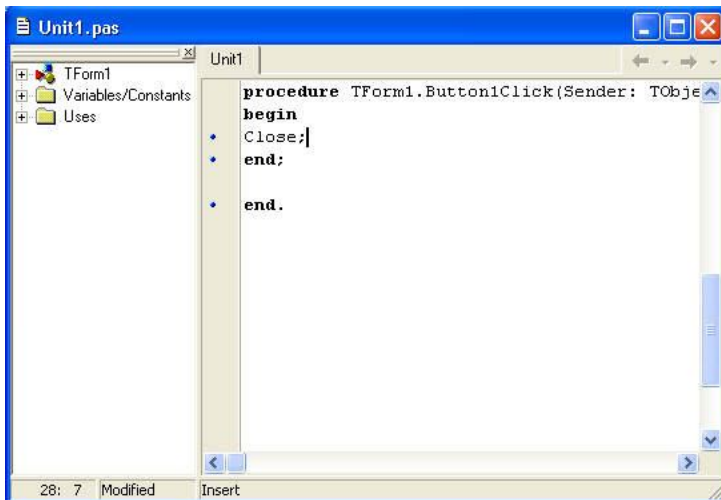
- 1-) İlk önce **File** menüsünden **Close All** diyerek, karşınıza gelen pencereden **No** deyiniz. Bunu yapmamızın sebebi henüz bir çalışan uygulama yapmadığımız için kaydedilmesinin gereksiz olduğu içindir.
- 2-) Tekrar **File** menüsünden **New Application** diyerek karşımıza boş bir form sayfası gelecek...
- 3-) Şimdi form üzerine bir buton ekleyiniz ve isimini "**Kapat**" olarak belirleyiniz. (Tabii isteğinize göre değişir :-)
- 4-) Butonun üzerine iki kez tıklayın. Karşınıza aşağıdaki gibi bir pencere gelecek...




Bu bizim bundan sonra karşımıza çıkacak ve bütün uygulamamızda kullanacağımız kod penceresidir... Her formun ve bileşenin kod penceresi farklıdır. Örneğin bu pencere form üzerindeki butonun kod penceresidir. Bu kod penceresi butona nasıl bir görev verileceğini belirler. Bir görev verebilmek için, o görev için gerekli olan kodu bilmek gerekir. Tabii burası en can alıcı nokta :-). Elimden geldiğince bazı ve sık kullanılan kodları sizlere göstereceğim fakat kodlar o kadar ki çok ve karışık ki hepsini anlatmam imkansız :-). (Kodlar ayrıca Delphi'nin temelini oluşturur. Bunu da belirtiyim :-).

Resimdeki gibi kodlar her zaman genellikle **begin** ve **end;** tagları arasına yerleştirilir... Bizde gerekli olan kodumuzu buraya yerleştireceğiz...

- 5-) Butona tıklayınca programın kapanmasını sağlayan kod **Close;** 'dur. Biz de bunu **begin** ve **end;** arasına yerleştiriyoruz o zaman... Yani kod penceresi aşağıdaki gibi olmalıdır...



6-) Son olarak programınızın çalışır haline görmek için kontrol (bileşen) paletinin yanındaki  (Run) tuşuna basınız. Yada menüden **Run > Run** yaparak da programınızın çalışır haline görebilirsiniz...

7-) Butona tıklayınca programınızın kapandığını göreceksiniz ;-)

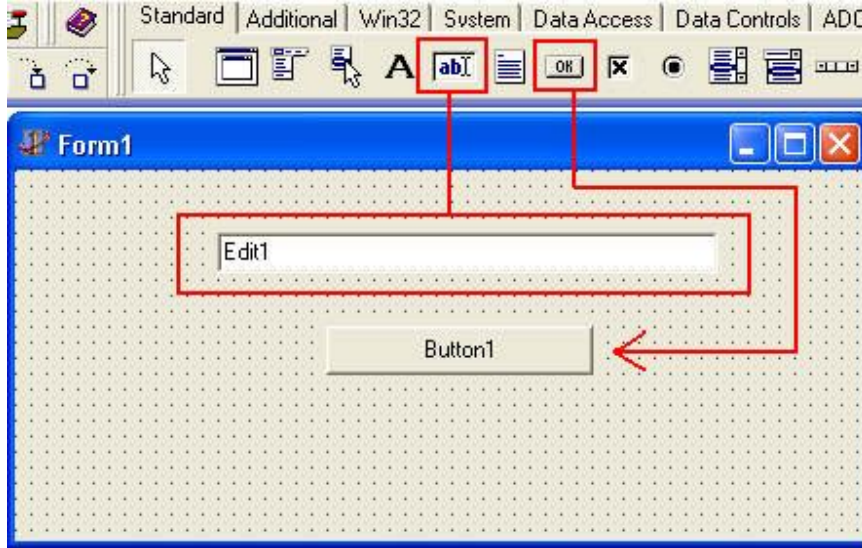
Ayrıca programınızın .exe hali **C:\Program Files\Borland\Delphi5\Projects\Project1.exe** dizinindedir... Bir programı bitirdiğinizde daha sonra bu dizinden yapmış olduğunuz programa bakabilirsiniz...

### **ADIM ADIM DELPHI – 3**

Örnek uygulamalarımıza devam edelim... Bu sefer programda gerçekleştirmek istediğimiz olay, butona tıklayınca **Edit1** bileşenin içinde istediğimiz bir yazının yazması...

1-) Tekrar yeni bir form oluşturalım.

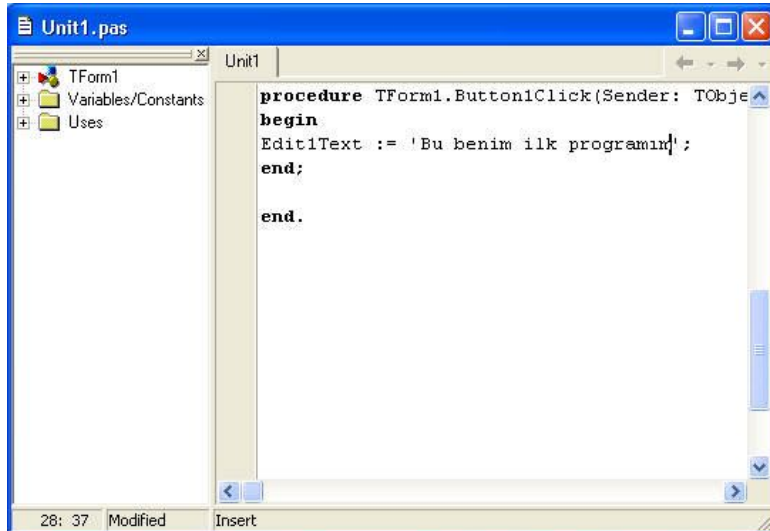
2-) Şimdi ise form üzerine bir **Button** ve **Edit** bileşeni ekleyelim.




3-) Butonun ismini isteğinize göre değiştirebilirsiniz.

4-) Şimdi tekrar butona iki kez tıklayarak butonun görevinin belirneceği kod penceresine geçelim.

5-) Bir bileşene tıklayınca edit kutusunda istenilen yazının çıkmasını sağlayan kod **Edit1Text := 'Mesajınız';** 'dır... Mesajınız yazan yere çıkmasını istediğiniz bir yazıyı yazabilirsiniz. Örneğin " **Bu benim ilk programım**" yazalım ve bu kodu **begin** ve **end;** tagları arasına yerleştirelim...



6-) Son olarak projemizi kaydedelim. File > Save As diyerek dosyaları istediğiniz bir klasöre kaydedin. Eğer kaydetmezseniz hatayla karşılaşabilirsiniz. Benden söylemesi :-)

7-) Şimdi programımıza  tuşuna basarak bakalım. Programımızdaki butona tıklayınca az önce yazdığımız mesaj görünecektir.

**Not:** Programa girdiğinizde muhtemelen Edit kutusunda "Edit1" yazacaktır. Eğer orda bir şey yazmasını istemiyorsanız ve bunu silmek isterseniz, form üzerindeki edit bileşenine bir kez tıklayın ve "Object Inspector" penceresindeki "Text" yazan bölümü bulun... Ordaki **Edit1** yazısını silin. İsterseniz kendi istediğiniz bir metnide yazabilirsiniz...

#### ADIM ADIM DELPHI – 4

Şimdi de programımıza klavye kullanılarak nasıl komut verildiğini öğreneceğiz. Örnek olarak programımızı herhangi bir klavye tuşuyla kapatmayı öğreneelim. Mesela sık kullanılan ESC tuşu ile.

1-) Yeni bir form oluşturalım.


2-) "Object Inspector" penceresindeki "Events" bölümüne girelim.

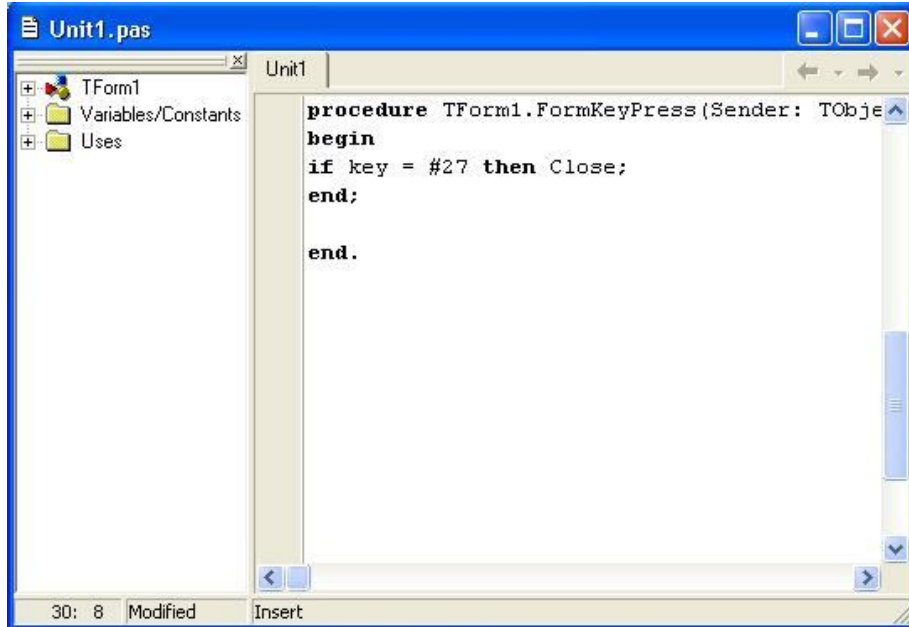
3-) "OnKeyPress" yerini bulun ve iki kez tıklayın. Bu bölüm klavye'den komut vermek için kullanılır. Klavye ile bağlanan görevler buradan ayarlanır... Daha detaylı olarak bir sonraki dersimizde öğreneceksiniz...



4-) Karşımıza yine meşhur kod penceresi gelecektir.

5-) Programı klavyedeki ESC tuşuyla kapatılmasını sağlayan kod **if key = #27 then Close;** 'dır... Buradaki 27 sayısı ESC tuşunu belirtir. Her tuşun sayı kombinasyonu farklıdır. (Diğer harflerin belirtildiği sayılar indirdiğiniz ZIP dosyasının içinde verilmiştir.)

6-) Her zaman ki gibi kodumuzu **begin** ve **end;** tagları arasına yerleştirip,  (Run) diyoruz... Unutmadan her uygulamamızda prejenizi kaydaderseniz daha iyi sonuç alırsınız... (Save Project As)



## ADIM ADIM DELPHI – 5

Umarım yavaş yavaş olayı kavramaya başlamışsınızdır :-). Az önce klavyedeki bir tuşa hangi komutla görev verileceğini öğrendiniz. Şimdi bu komutu başka şekilde kullanacağız... Hatırlarsanız komut **if key = #.. then** 'di. (.. yazan yere klavyedeki bir tuşa karşılık gelen sayı yazılıyordu hatırlarsanız) ve ayrıca **"Close"** yazmıyoruz çünkü o kapatmak için gerekli olan komuttu. Biz şu an kapatma ile ilgili bir şey yapmayacağımız için o komutu burada kullanmayacağız... Bu sefer klavyedeki bir tuşa basarak istediğimiz bir yazıyı form üzerindeki **Memo (Metin Kutusu/ Edit bileşenine benzeyen başka bir bileşen)** üzerinde çıkmasını sağlayacağız... Memo ile Edit bileşenleri birbirine çok benzerler. Bunları karıştırmayın sakın :-)

**1-)** İlk önce yeni form oluşturalım.

**2-)** Form üzerine **Memo** bileşenini ekleyelim.

**3-)** Form üzerindeki **Memo** bileşenini seçili duruma getirip, "Object Inspector" penceresindeki "Events" bölümüne girelim. **"OnKeyPress"**'i bularak üzerine iki kez tıklayalım. Hatırlarsanız bu komut klavye tuşlarına atanan bir görevi gerçekleştirmek için kullanılan bir komuttu. Görevi ne olursa olsun, klavyedeki bir tuşa basılarak yapılan işlemler daima bu komut içinde gerçekleşir. Bunu unutmayın ;-). Bir de görev hangi bileşen üstünde gerçekleşecekse, onun **"OnKeyPress"**'i kullanılmalıdır. Örneğin şu an bizim üzerinde çalıştığımız uygulamada görev hangi bileşen üzerinde gerçekleşecek.. Hadi tahmin edin. Tabii ki **Memo** bileşeninin üzerinde gerçekleşecek. O zaman bizde **Memo** bileşeninin **"OnKeyPress"**'ini kullanacağız. Bunu yapabilmek için form üzerine eklediğimiz **Memo** bileşeninin üzerine bir kez tıklayarak seçili duruma getirin ve "Object Inspector" penceresindeki "Events" bölümüne giderek **"OnKeyPress"**'in üstüne iki kez tıklayın. (Form üzerinde hangi bileşen seçili durumdaysa Events bölümünde onun OnKeyPress komutu görülür...) (Adım adım delphi 4'te programı ESC ile kapatma dersimizi hatırlayın. Burada görev form (program) üzerinde gerçekleşeceği için Form'un "OnKeyPress"ini kullandık) Karşımıza kod penceresi gelecek.

**4-)** Biz yine ESC tuşunu kullanalım. Gerekli komutumuz **if key = #27 then memo1.Lines.Add('Mesajınız');** 'dir. Bu komutları ben size açıklayım tekrar...

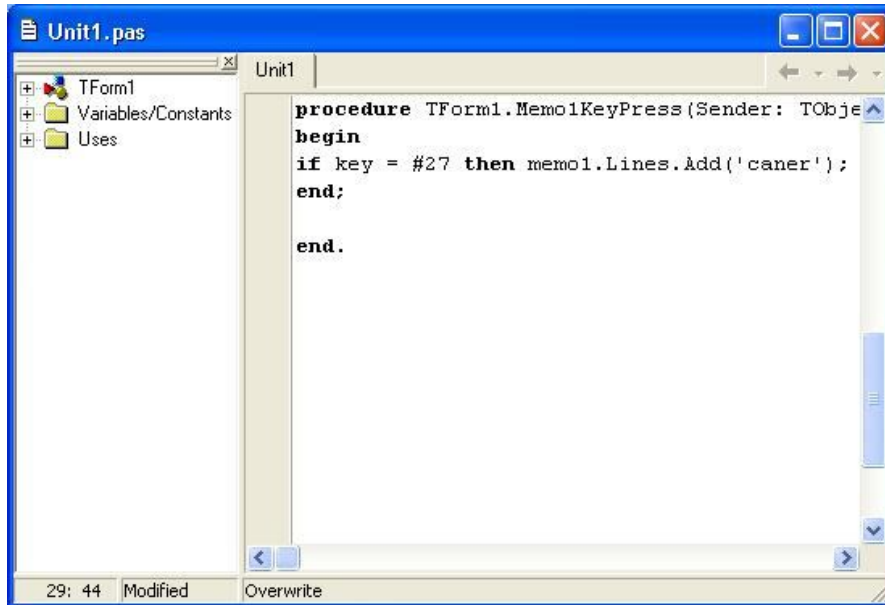
**if key = #27 then** --> Bunu zaten biliyorsunuz. Klavyedeki tuşların kullanımını sağlayan komut.

**memo1.Lines.Add('Mesajınız');** --> Memo bileşeninde mesajınızın görünmesini sağlayan kod.


**Bu kodun Türkçesi şu yani** --> 27 tuşuna basınca memo1 üzerinde ..... mesajını göster

**Not:** Ayrıca formunuzda birden fazla memo bileşeni varsa, bunlar memo2, memo3, memo4 ...vs diye devam eder. Eğer onları kullanmak isterseniz kodumuzdaki **memo1** yerine memo2, memo3 ...vs yazarsanız o bileşen üzerinde gösterilecektir... Ayrıca bu Edit bileşeninde böyledir. Onda da sayıyı değiştirirseniz diğerinde gösterilecektir...

**5-)** Mesajımız "caner" olsun. (Kendi ismin, tabii siz kendi isminizi de yazabilirsiniz :-). Böylece kod penceresi aşağıdaki gibi olmalıdır.



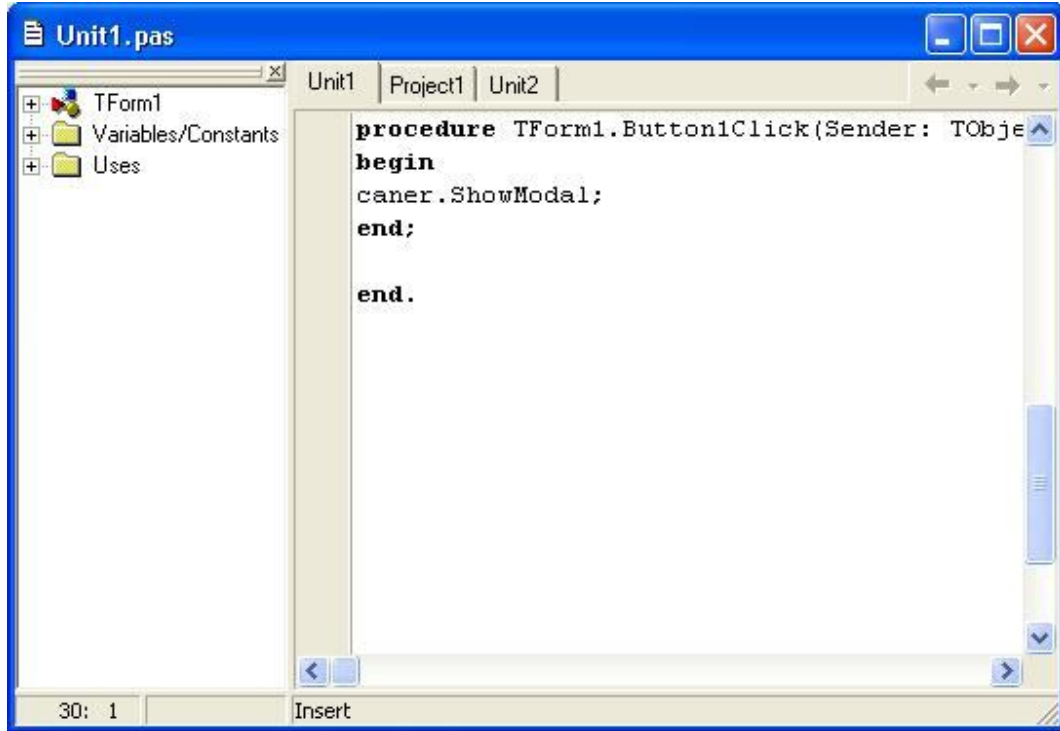


6-) Son olarak  (Run) diyerek programımızı çalıştıralım. ESC tuşuna bastığımızda mesajınız belirecektir...


#### **ADIM ADIM DELPHI – 6**

Şimdiki uygulamamızda form üzerindeki bir butona tıklayarak başka bir form penceresinin açılmasını sağlayacağız... Bu uygulama fazlaca kullanılır. Bu yüzden bunu adınız gibi bilin :-)

- 1-) Yeni bir form oluşturun.
- 2-) Form'a bir buton ekleyin.
- 3-) Butonu seçili duruma getirerek "Object Inspector" penceresindeki **Caption** bölümünde butonun üzerinde bize görünen her hangi bir isim yazın.
- 4-) Şimdi **File** menüsünden **New Form** diyerek ikinci formumuzu da oluşturalım...
- 5-) Form2'de "Object Inspector" penceresinde ki "Name" bölümüne giderek birleşik olarak küçük harflerle bir isim girin. Bu isim Form2 üzerinde gözükmeyen, fakat kod yazımında gerekli olan isimdir. Kod yazarken bu ismi kullanacağız. O yüzden kolay bir isim yazın... Örneğin ben **caner** yazıyorum.
- 6-) Form2'nin başlığını değiştirmek için, "Object Inspector" penceresindeki **Caption** bölümüne giderek istediğiniz bir ismi verin...
- 7-) Oluşturduğumuz Form2'yi biraz kaydırarak alttaki Form1 penceresine tıklayın.
- 8-) Form1 penceresindeki butona iki kez tıklayalım. Böylece butonun kod sayfasına gireceğiz... Görevi gerçekleştiren buton olduğu için onun kod penceresine geçmekteyiz...
- 9-) Yeni bir pencere açmak için kullanılan kodlar **caner.Show;** ve **caner.ShowModal;** 'dır. Bu kodda caner yerine isim olarak siz ne yazdıysanız onu kullanın... Bu iki kod arasında fark; ShowModal'da, açılan pencere kapanana kadar Form1 penceresine geri dönebilirsiniz...
- 10-) Her zaman ki gibi yine kodumuzu **begin** ve **end;** tagları arasına yerleştiriyoruz... Ben ShowModal'ı tercih ettim. Bu daha çok hoşuma gidiyor :-)
- 11-) Son olarak en önemlisi yaptığımız işlemleri kaydetmeliyiz. **File** menüsünden **Save Project As** diyerek proje dosyalarını boş bir klasöre kaydedin. Eğer kaydetmezseniz hata verir ;-)



12-) Şimdi projemize bakma zamanı geldi ;-)

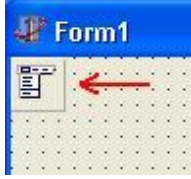
 (Run) diyerek programınızı deneyin. Butona tıkladığınızda form2 penceresi açılacaktır...

## ADIM ADIM DELPHI – 7

Şimdiye kadar ki öğrendiğimiz bilgilerden yola çıkarak, bir not defteri yapacağız. WordPad benzeri bişey olucak fakat bizimkisi basit olacak. Eeee yavaş yavaş öğreniyoruz... ;-) Bu biraz sizi zorlayabilir aman kafanız karışmasın sakın...

**1-)** Yeni bir form oluşturun. (Tüm uygulamalarınızda yeni form oluştururken mutlaka **File > Close All** seçeneğiyle önceki çalışmanızı kapatın)

**2-)** **MainForm** bileşenini ekleyin. **MainForm** bileşeni programımıza menü eklememizi sağlar ve çok önemlidir. (MainForm, form'a eklendiğinde kare şeklinde çıkacaktır. O kareyi sol üst kenara doğru sürükleyin. En köşede kalsın.)




**3-)** Az önce eklediğimiz MainForm bileşenine çift tıklayın (Kareye yani). Aşağıdaki ekran gelecektir...



**4-)** Şimdi bu pencere seçili konumdayken **"Object Inspector"** penceresinde **Caption** bölümüne giderek "Dosya" yazıp Enter'a basın. (Tabii istediğinizi yazmakta özgürsünüz fakat genel olarak "Dosya" yazılır.) Böylece Dosya menüsü oluşmuş olur.

**5-)** Şimdi ise "Dosya"nın üstüne bir kez tıklayın. Altında ufak boş bir pencere açılacaktır. O boş pencereye tıklayın. **"Object Inspector"** penceresindeki **Caption** bölümüne "Yeni" yazıp tekrar Enter'layın.

**6-)** Şimdi ise "Yeni"nin altındaki boşluğa tıklayarak **"Object Inspector"** penceresindeki **Caption** bölümüne giderek "Çıkış" yazın ve son kez Enter'layın. Onun altında bir boşluk daha açılacaktır. Şimdilik ona dokunmayın. İki satırlık

menü yeter bize ;-) Yaptıklarımızın ne olduğunu merak ediyorsanız  (Run) diyerek bir bakabilirsiniz...

**7-)** Run dedikten sonra programın önizlemesini gösterecektir. Önizlemeyi kapatırsanız Form'a geri dönecektir... Menüdekilere her hangi bir komut vermediğimiz için tıklasanız bile bişey olmayacaktır...

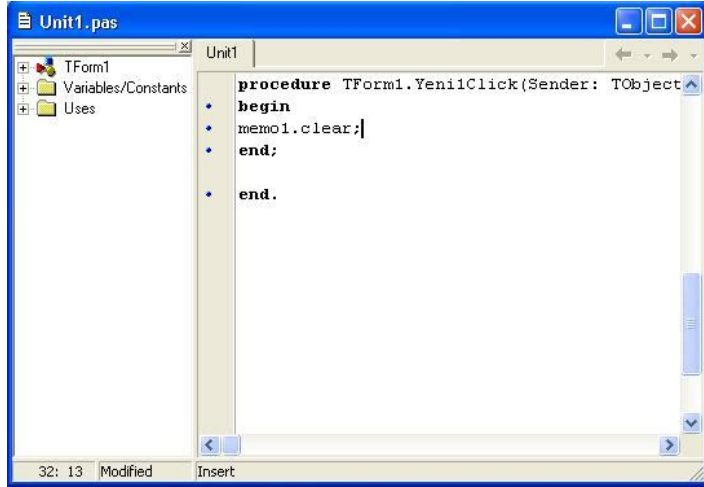
**8-)** Şimdi asıl formumuza geri dönerek (Kareli olan) bileşen paletinden **"Memo"** bileşenini ekleyin.

**9-)** **Memo** bileşenini seçili duruma getirdikten sonra **"Object Inspector"** penceresindeki **"Align"** seçeneğini **AllClient** yapın. Böylece **Memo** bileşeni form kadar genişleyecektir...

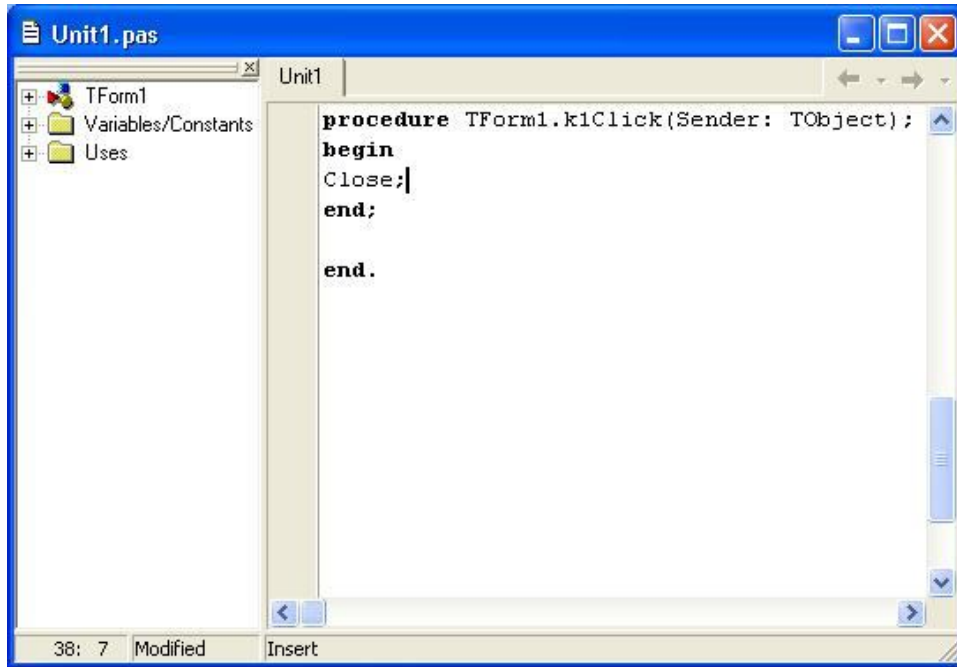
**10-)** Şimdi de az önce oluşturduğumuz "Dosya" menüsündekilere gerekli görev komutlarını yazalım... Asıl formumuz üzerindeki MainForm bileşenine (sol köşeye attığımız kare şeklindeki bileşen) tekrar çift tıklayarak menümüze ulaşalım...

**11-)** "Yeni" yazan yere çift tıklayarak, "Yeni" öğesinin gerçekleştireceği görevin sağlanması için kod penceresi gelecek. Her zaman olduğu gibi **begin** ve **end;** tagları arasına **memo1.clear;** komutunu girelim. Bu kod, "Dosya" menüsündeki "Yeni" öğesine tıklanınca **"Memo1"** bileşenindeki tüm yazıları temizle" anlamına gelmektedir...






**12-)** Şimdi ise MainForm'a tekrar çift tıklayarak "Dosya" menüsüne ulaşalım ve "Dosya"ya tıklayarak "Çıkış" yazan yere çift tıklayalım. Bu seferde "Çıkış" öğesinin görevini belirleyeceğiz. Çıkış öğesi programı kapatmamızı sağlayacak. Daha önceden öğrenmiştik hatırlarsanız. Kapama komutu **Close;** idi. Gerekli kodu yazın ve form penceresine geri dönün...



**13-)** Son olarak da bileşen paletinden **StatusBar**'ı form'a ekleyin. Kendisi zaten otomatik olarak programın (formun) alt tarafına yerleşecektir...

**14-)** Ve tabii ki projemizi kaydetmeyi unutmayalım... **File > Save Project As**

**15-)**  diyerek programımızı deneyelim. Böylece elimizde basit ama kullanışlı bir not defteri oldu... ;-)

**Not:** Programa girdiğinizde, **Memo1** yazısıyla karşılaşmak istemiyorsanız, **Memo** bileşenini seçili duruma getirin. "Object Inspector" penceresindeki **Line** seçeneğine çift tıklayarak gelen penceredeki **Memo1** yazısını silin ve OK ile çıkın.

İşte programımızın son hali. Tabii siz bunu geliştirebilirsiniz öğrendiğimiz bilgiler çerçevesinde...



Ayrıca yaptığınız programların icon'unu (.ico) değiştirmek isterseniz **Project > Options** menüsünden **Application** bölümüne giriniz. Karşınıza programın şu anki iconu gelecektir. Iconun yanındaki **Load Icon** 'a tıklarsanız, çıkan pencerede kendi iconunuzu ayarlayabilirsiniz...

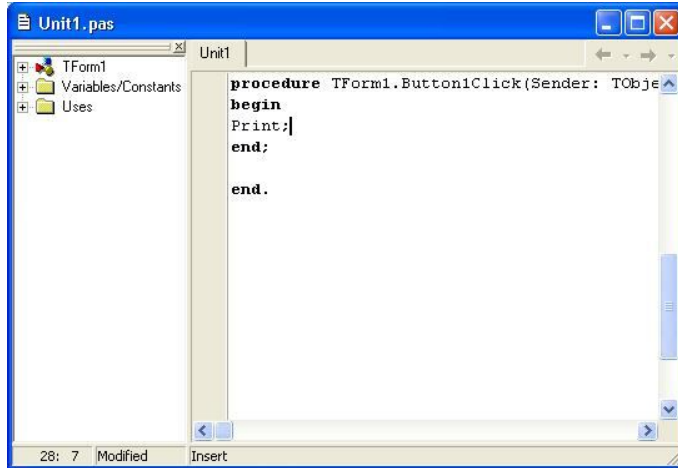
#### **ADIM ADIM DELPHI – 8**

Bu sefer de çok basit ama işe yarar bir şey öğreneceğiz. Forumdaki bir bileşene "Yazdır" komutunu vereceğiz. Yani yazıcımıza yazdırma komutunu göndereceğiz... Örneğin yukarıda bir not defteri uygulaması yapmıştık. Bu yazdır komutunu not defterine eklerseniz, yazılan yazıları yazıcıdan çıkarmasını sağlayabiliriz...

**Artık bu zamanda ben size sadece komutu versem siz ne yapacağınızı bilmeniz gereken bir duruma gelmiş olmalısınız... ;-)**

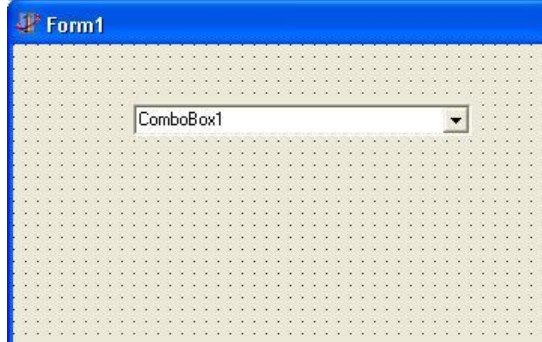
Yazıcıdan çıkartma komutu **Print;** 'dir. Yani bir butona **Print;** kodunu verirsiniz, o butona basıldığında form üzerindeki yazıcıdan çıkarmanızı sağlar...

Kodu butonun kod penceresinde **begin** ve **end;** tagları arasına yerleştirin ve uygulamayı çalıştırın. (Biliyorsunuz ki herhangi bir bileşenin kod sayfasına girebilmek için form üzerinde o bileşene çift tıklamak gerekir.) İşte bu kadar..



## ADIM ADIM DELPHI – 9

Bu yeni dersimizde başka bir bileşeni ve bu bileşen üzerinde gerçekleşen bazı görevleri öğreneceğiz. Bu bileşenimizin adı **ComboBox (Açılan Kutu)** 'dur.



Buna bir **menü kutusu** da diyebiliriz. İçine istediğimiz kadar öğe ekleyip bunlara belirli görevler atayabiliriz... Bu bileşen bize form üzerinde yer kazanmamızı sağlar...

Şimdi ki uygulamamızda **ComboBox** 'un içindeki bir öğeye tıklanınca **Memo** bileşeninde istediğimiz yazının yazmasını sağlayacağız...

- 1-) Yeni bir form oluşturalım. (Tüm uygulamalarınızda yeni form oluştururken mutlaka **File > Close All** seçeneğiyle önceki çalışmanızı kapatın)
- 2-) Bileşen paletinden **"Standard"** sekmesinde **"ComboBox"** adlı bileşeni form üzerine ekleyin. Ayrıca bunun boyutunu form üzerinde istediğiniz gibi ayarlayabilirsiniz.
- 3-) Şimdi de bileşen paletinden form'a **"Memo"** bileşenini ekleyiniz.
- 4-) ComboBox'un içine bir öğe eklemek için **"Object Inspector"** penceresindeki **"Items"** bölümüne gelerek çift tıklayınız. Alt alta eklenmesini istediğiniz öğeleri yazınız. Örneğin biz **"deneme1, deneme2, deneme3"** yazalım. Yazdıktan sonra **"OK"** deyiniz.



4-) Şimdi ise **ComboBox** bileşeninin üzerine gelerek çift tıklayınız. Böylece combobox'un kod penceresine geçmiş olacağız.

5-) Şimdi **begin** ve **end;** arasına aşağıdaki kodu girin. Biliyorum kod biraz uzun ama öğreneceksiniz.

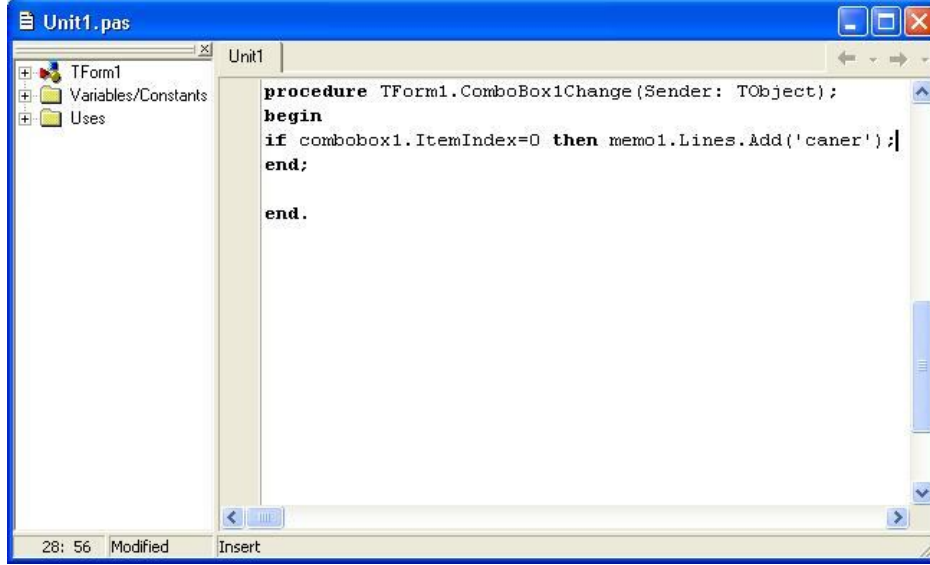
```
if combobox1.ItemIndex=0 then memo1.Lines.Add('caner');
```

**ItemIndex=0** -> Bu bizim **ComboBox** bileşenine eklediğimiz öğeleri temsil ediyor. Örneğin 0 (Sıfır) deneme1'i temsil ediyor. **ItemIndex=1** olursa deneme2'yi temsil edecek. **ItemIndex=2** olursa deneme3'ü temsil edecek. Yani İlk **item** için ('deneme1') bu numara sıfırdır. Daha sonra da birer birer artar...

**memo1.Lines.Add('caner');** -> **memo1 bileşenin içinde "caner" adlı mesajı göster** anlamına gelmektedir. Ordaki caner yazısını isteğinize göre değiştirebilirsiniz...

```
if combobox1.ItemIndex=0 then memo1.Lines.Add('caner');
```

**Anlamı:** ComboBox bileşeninde "....." numaralı öğe seçilirse memo1 bileşeninde ..... mesajını göster.

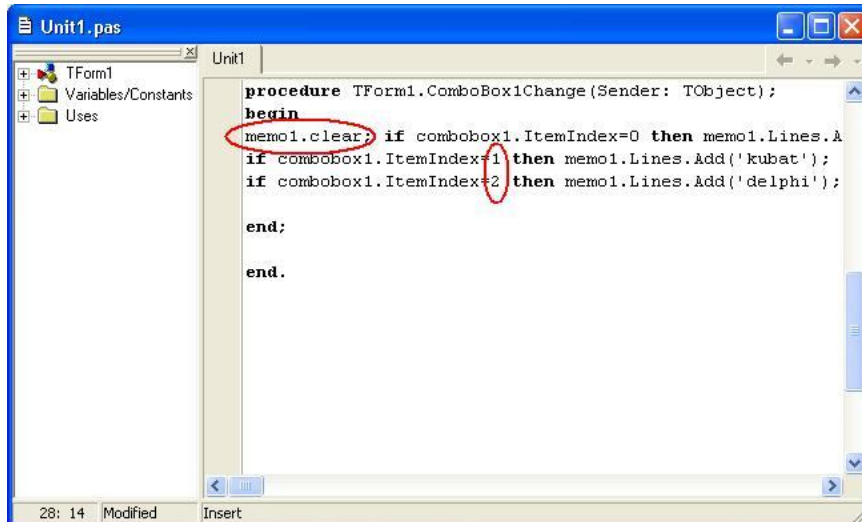


**6-)** Programımızı  (Run) diyerek çalıştıralım. **ComboBox** bileşenindeki **deneme1** öğesini seçersek **memo1** bileşeninde istediğimiz yazının çıktığını göreceğiz...

**Not:** ComboBox bileşeninin üzerindeki "ComboBox" yazısını silmek veya kendi istediğiniz bir yazıyı girmek istiyorsanız bu bileşen seçili durumda iken "**Object Inspector**" penceresindeki **Text** bölümünü bulun. Ordaki yazıyı silin veya kendi mesajınızı ekleyin...

**Not2:** Biz sadece **deneme1** öğesine görev atadık. Siz isterseniz deneme2'ye ve deneme3'e de görev atayabilirsiniz. Bunun için az önce yazdığımız kodun altına aynı kodu tekrar yazınız. Sadece **ItemIndex=0** sayısını 1 arttırarak yazın mesajınızı değiştirin.

**Not3:** Ayrıca çoklu öğe seçimlerinde ilk kodun başına **memo1.clear;** komutunu da mutlaka veriniz. Bu kod şu işe yaramaktadır. Örneğin combobox'tan deneme1 öğesini seçtiniz ve memo bileşeninde "caner" yazısı çıktı. Sonra deneme2'yi seçtiniz ve "kubat" yazısı çıktı. Fakat bu çıkan ikinci yazı ilk yazının altına çıkıyor. Yani ilk çıkan yazı silinmiyor. Böylece mesaj kalabalığı yaratıyor. Eğer bu kodu koyarsanız deneme1'i seçtikten sonra, deneme2'yi seçerseniz ilk çıkan yazı silinir ve deneme2 yazısı tek olarak görünür. ;-)



**Not4:** ComboBox kutusundan seçilen bir öğeyi sadece memo bileşeninde yazı çıkarmak için kullanmayabilirsiniz. **if combobox1.ItemIndex=0 then** komutundan sonra bizim öğrendiğimiz başka komutları yazarsanız öğe o görevi yapacaktır...

Örneğin **if combobox1.ItemIndex=0 then Close;** yaparsanız o öge seçildikten sonra program kapanır.  
Örneğin **if combobox1.ItemIndex=0 then caner.ShowModal;** öge seçildikten sonra başka bir form penceresi açılır.

Yani böyle kodları karıştırarak bileşenlere ve ögelere yeni görevler atayabilirsiniz...

Benden şimdilik bu kadar. Umarım sizlere bir nebze de olsa bir şeyler öğretebilmişimdir. Eğer dersler hoşuna gittiyse ve devamını isterseniz bana ulaşınız...

Mail: [caner.kubat@mynet.com](mailto:caner.kubat@mynet.com)