

T.C.
MİLLİ EĞİTİM BAKANLIĞI



MEGEP

(MESLEKİ EĞİTİM VE ÖĞRETİM SİSTEMİNİN
GÜÇLENDİRİLMESİ PROJESİ)

BİLİŞİM TEKNOLOJİLERİ

VERİ TABANINDA MAKROLAR

ANKARA 2008

Milli Eğitim Bakanlığı tarafından geliştirilen modüller;

- Talim ve Terbiye Kurulu Başkanlığının 02.06.2006 tarih ve 269 sayılı Kararı ile onaylanan, Mesleki ve Teknik Eğitim Okul ve Kurumlarında kademeli olarak yaygınlaştırılan 42 alan ve 192 dala ait çerçeve öğretim programlarında amaçlanan mesleki yeterlikleri kazandırmaya yönelik geliştirilmiş öğretim materyalleridir (Ders Notlarıdır).
- Modüller, bireylere mesleki yeterlik kazandırmak ve bireysel öğrenmeye rehberlik etmek amacıyla öğrenme materyali olarak hazırlanmış, denenmek ve geliştirilmek üzere Mesleki ve Teknik Eğitim Okul ve Kurumlarında uygulanmaya başlanmıştır.
- Modüller teknolojik gelişmelere paralel olarak, amaçlanan yeterliği kazandırmak koşulu ile eğitim öğretim sırasında geliştirilebilir ve yapılması önerilen değişiklikler Bakanlıkta ilgili birime bildirilir.
- Örgün ve yaygın eğitim kurumları, işletmeler ve kendi kendine mesleki yeterlik kazanmak isteyen bireyler modüllere internet üzerinden ulaşılabilirler.
- Basılmış modüller, eğitim kurumlarında öğrencilere ücretsiz olarak dağıtılır.
- Modüller hiçbir şekilde ticari amaçla kullanılamaz ve ücret karşılığında satılamaz.

İÇİNDEKİLER

AÇIKLAMALAR	İ
GİRİŞ	1
ÖĞRENME FAALİYETİ-1	3
1. MAKRO	3
1.1. Makro	3
1.2. Makro Oluşturma ve Komut Düğmelerine Makro Atama	4
1.3. Eylem Türleri	8
1.3.1. Kayıt Gezinme Eylemleri	8
1.3.2. Veri Tabanı Nesneleri Açma Eylemleri	9
1.3.3. Form Eylemleri	9
1.3.4. Uygulama Eylemleri	9
1.4. Makro Tasarım Penceresi	9
1.5. Eylem Değişkenleri	10
1.6. Kod Yazmak	12
UYGULAMA FAALİYETİ	14
ÖLÇME VE DEĞERLENDİRME	15
ÖĞRENME FAALİYETİ-2	16
2. MAKRO DÜZENLEME	16
2.1. Eylemlerin Çalışma Yönü	16
2.2. Bir Makroya Daha Fazla Eylem Ekleme	17
2.3. “Makro Çalıştır” Penceresi	17
2.4. Makrolarda Gruplandırma	18
2.5. Makroyu Bir Olaya Atama	18
2.6. Diğer Önemli Makro Eylemleri	20
UYGULAMA FAALİYETİ	22
ÖLÇME VE DEĞERLENDİRME	23
ÖĞRENME FAALİYETİ-3	24
3. ŞARTLI DEYİMLER	24
3.1. Deyim Oluşturucusu Penceresi	24
3.2. Koşullu Deyimin Parçaları	27
3.3. Klavye Kısa Yolları Yapmak	27
UYGULAMA FAALİYETİ	29
ÖLÇME VE DEĞERLENDİRME	30
MODÜL DEĞERLENDİRME	31
CEVAP ANAHTARLARI	32
ÖRNEK ÇALIŞMA	33
ÖNERİLEN KAYNAKLAR	1
KAYNAKÇA	40

AÇIKLAMALAR

KOD	481BB0039
ALAN	Bilişim Teknolojileri
DAL/MESLEK	Veri Tabanı Programcılığı
MODÜLÜN ADI	Veri Tabanında Makrolar
MODÜLÜN TANIMI	Veri Tabanında makro yapımı ile ilgili öğretim materyalidir.
SÜRE	40/24
ÖN KOŞUL	“Veri Tabanında Formlar ve Raporlar” modülünü bitirmiş olmak
YETERLİK	Veri tabanında makro yapmak
MODÜLÜN AMACI	Genel Amaç Gerekli ortam sağlandığında, veri tabanı programında makro yazabileceksiniz. Amaçlar ➤ Makro oluşturabileceksiniz. ➤ Makro düzenleyip bir olaya atayabileceksiniz. ➤ Şartlı deyimler oluşturabileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	Bilgisayar laboratuvarı ve bu ortamda bulunan; bilgisayar, bilgisayar masaları, kâğıt, kalem, veri tabanı programı, yedekleme için gerekli donanım, raporlama için yazıcı, sayfa için internet bağlantısı ve lisanslı işletim sistemi programı
ÖLÇME VE DEĞERLENDİRME	➤ Her faaliyet sonrasında o faaliyetle ilgili değerlendirme soruları ile kendi kendinizi değerlendireceksiniz. ➤ Modül içinde ve sonunda verilen öğretici sorularla edindiğiniz bilgileri pekiştirecek, uygulama örneklerini ve testleri gerekli süre içinde tamamlayarak etkili öğrenmeyi gerçekleştireceksiniz. ➤ Sırasıyla araştırma yaparak, grup çalışmalarına katılarak ve en son aşamada alan öğretmenlerine danışarak ölçme ve değerlendirme uygulamalarını gerçekleştireceksiniz.

GİRİŞ

Sevgili Öğrenci,

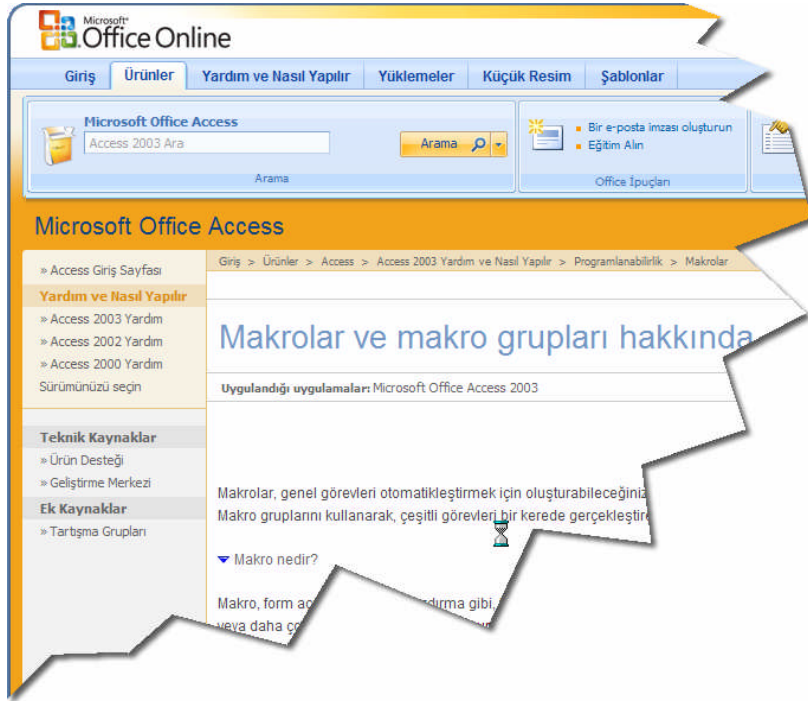
Günümüzde bilgisayar kullanımının yaygın olarak artması, birçok bilginin bilgisayarda daha rahat saklanması “veri tabanı programları”na olan ilgiyi fazlalaştırmıştır.

Bilişimin temel konusu olan veri tabanı; dağınık şekilde duran ve işe yaramaz halde olan bilgilerin, işlenebilir, içinde arama yapılabilir, sıralanabilir hale gelmesini sağlayan bir yardımcı programdır. Bilgilerin kullanıcıya taze ve hızlı bir şekilde ulaştırılmasından da sorumludur.

Bu modülde veri tabanı programını kullanarak makro hazırlayabileceksiniz. Önceki dört modülü bitirerek geldiğiniz “Veri Tabanında Makrolar” modülünde, kullanıcıya yardımcı olabilecek faydalı fonksiyonları hazırlayacaksınız.

“Bilişim Teknolojileri Temelleri” ve “Görsel Programlama” derslerinde “program” yapımı gösteriliyor. Bu sebeple “Veri Tabanında Makrolar” modülünde çok derine inmeden yardımcı programlar yapacağız.

Alışlagelmiş programlama dillerinde bir “ana prosedür” merkezinde işlemler yapılır. Access’te ise “olay tabanlı – *event driven*” programlama vardır. Kullanıcının uygulama ile olan etkileşimine göre işlem silsilesi değişir.



ÖĞRENME FAALİYETİ-1

AMAÇ

Makro oluşturabileceksiniz.

ARAŞTIRMA

- Excel ve Word programında yapılmış makrolar araştırınız. Neden makro içeren bir belge “güvenlik” ayarları ile korunuyor?
- Bir “makro” araç çubuğuna, kısa yol tuşuna veya menüye eleman olarak nasıl eklenebilir?

1. MAKRO



Eğer bazı işlemleri tekrar tekrar yapıyor iseniz, bu işlemleri “Makro” kullanarak kendiliğinden otomatik olarak yapılmalarını sağlayabilirsiniz.

1.1. Makro

Bir “makro”; rutin işlemleri otomatik hale getirmemize yardımcı olan nesnedir. Dizi halinde peş peşe yaptığımız işlemler yığını devamlı yapıyor ve bu bizim zaman kaybetmemize sebep oluyor ise, bu işlemleri yerine getiren tek bir makro ile çalışmalarımızı hızlandırabiliriz.

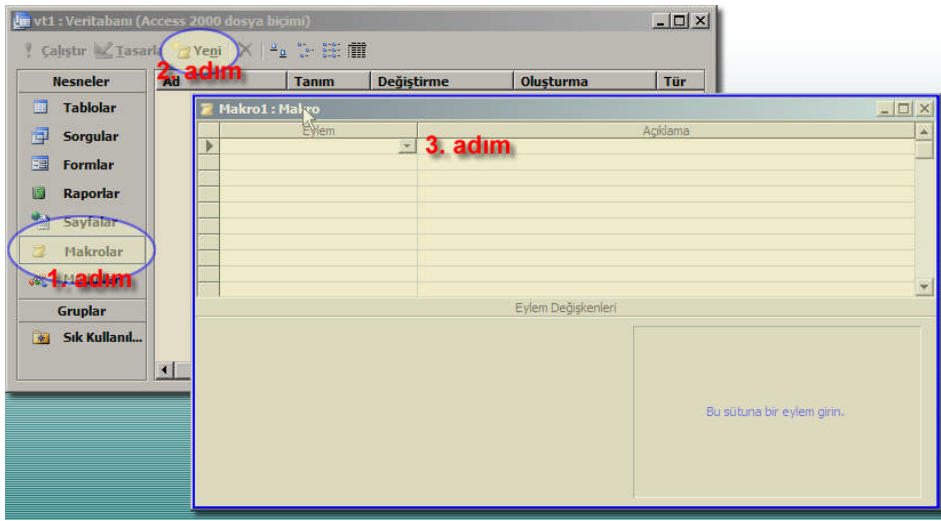
Örneğin bir rapor alma işlemini inceleyelim:

- Veri tabanı penceresinden “Raporlar” kısmına gel
- Çıktısı alınacak raporu bul
- Raporu aç
- Yazdır
- Raporu kapat

Bu beş işlemi sırası ile devamlı yapmak yerine, “Raporu Yazdır” diye tek bir düğmenin tıklanmasına bağlayabiliriz. Böylece bir veya daha fazla komutu “bir komut” haline getirerek makrolar bize yardımcı olur.

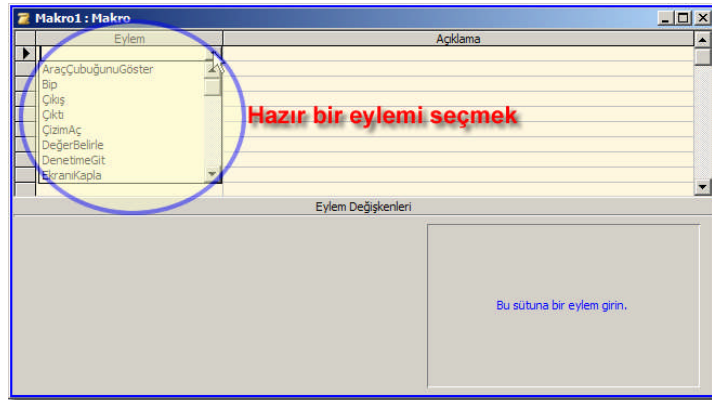
Makro, çok basit olarak programcılığa giriş denilebilir. Makroyu, otomatik hale getirilen karmaşık prosedürler ve görevlerde kullanıyoruz. Nerede ise tek satır kod yazmak zorunda değilsiniz.

1.2. Makro Oluşturma ve Komut Düğmelerine Makro Atama



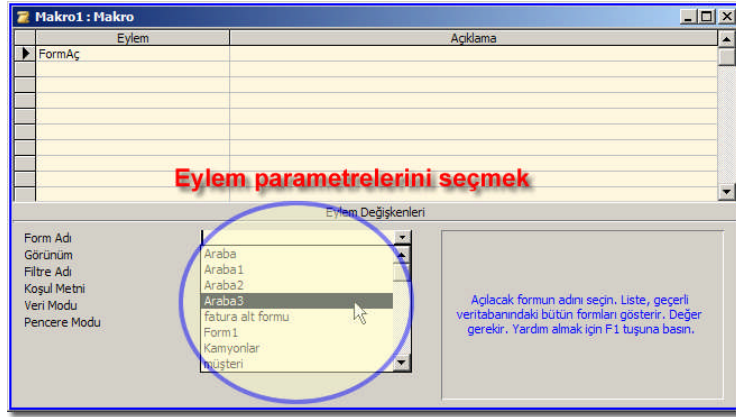
Resim 1. 1: Makro tasarım penceresinin açılması

Daha önceden yaptığımız bir veri tabanı dosyasını açarak, yeni bir makro oluşturalım. “Yeni” komutunu verdiğimizde açılan pencere “Veri Tabanında Tablolar” da gördüğümüz tablo tasarım penceresine benziyor. Burada “alan” eklemek yerine “Eylem” ekliyoruz.

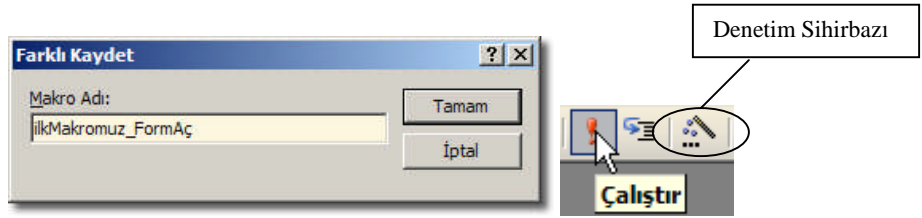


Resim 1. 2: Eylem seçmek

Makrolar genellikle bir düğmenin tıklanma olayına atanır. Yani oluşturduğumuz makroyu bir form üzerindeki düğmeye bağlayabiliriz (Resim 1.3).



Resim 1. 3: Eylemin parametrelerini düzenlemek



Resim 1. 4: Makronun adını belirleyerek kaydetmek ve çalıştırmak

Makromuzun adını “İlkMakromuz_FormAç” olarak belirledikten sonra artık “Çalıştır” komutunu vererek deneyebiliriz (Resim 1.4).

Bir düğmeye atayabileceğimiz basit bir makromuz var. Daha sonra göreceğimiz örneklerde daha fazla işlevi olan makrolarımız olacak.

Boş bir form açıp içine düğme ekliyoruz. Eğer “Denetim Sihirbazı” seçili ise “Komut Düğmesi Sihirbazı” açılacaktır. Kategorilerden “Çeşitli – Makro Çalıştır” seçeriz (Resim 1.5).

Sonraki sihirbaz sayfasından çalıştırılacak makronun adı ve düğmede görüntülenecek resim seçilir (Resim 1.5). Formun ve düğmenin görünümü düzenlendikten sonra örnek formunuzu deneyebiliriz (Resim 1.6).

Aslında arka planda, Office 2003’te tam sürüm olmayan* “Visual Basic 6.3”† vardır. Resim 1.7’de görüleceği gibi, sihirbazlar sayesinde kod yazmadan **makro** yapabiliriz.‡

* Microsoft bu yardımcı dile “Visual Basic for Applications – VBA” ismini vermiştir.

† Office 2007’de “Visual Basic 6.4” sürümü vardır.

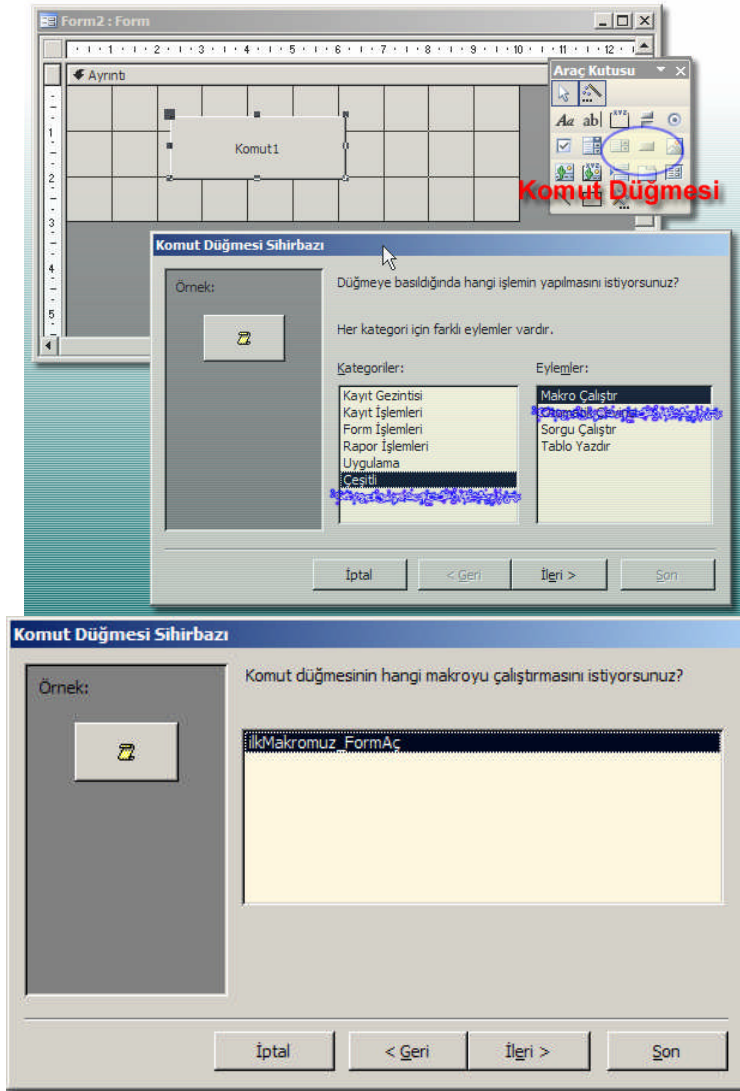
‡ “Araçlar*Makro*Visual Basic Düzenleyicisi (Alt+F11)” ile açabilirsiniz.



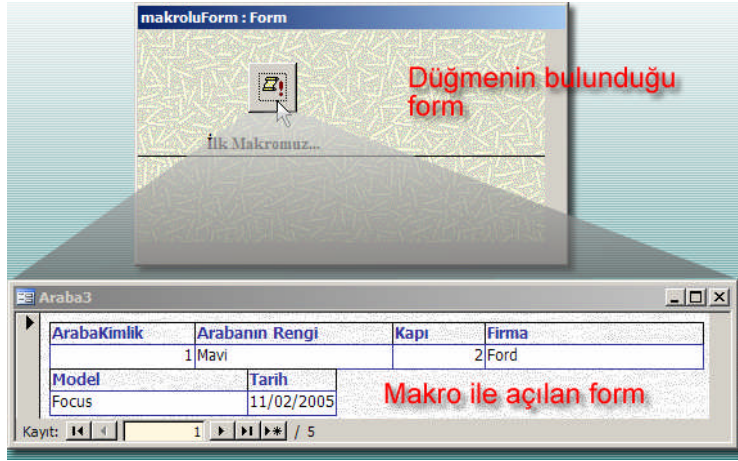
İleri seviye programcılar Visual Basic kodları ile Office programlarına tam hâkimiyet sağlayabilir.



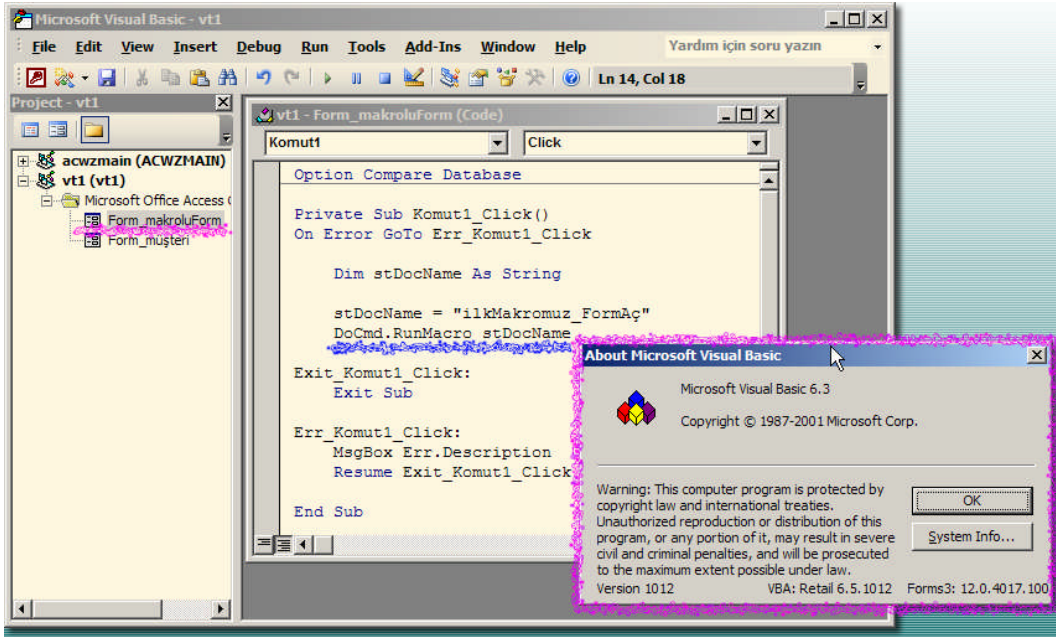
Excel ve Word'de bulunan “Araçlar*Makro*Yeni Makro Kaydet komutu, (Makro Kaydedicisi)” Access'te bulunmaz.



Resim 1. 5: Boş bir form içine komut düğmesi eklemek ve düğmenin çalıştıracacağı makronun adı “İlkMakromuz_FormAç” seçmek



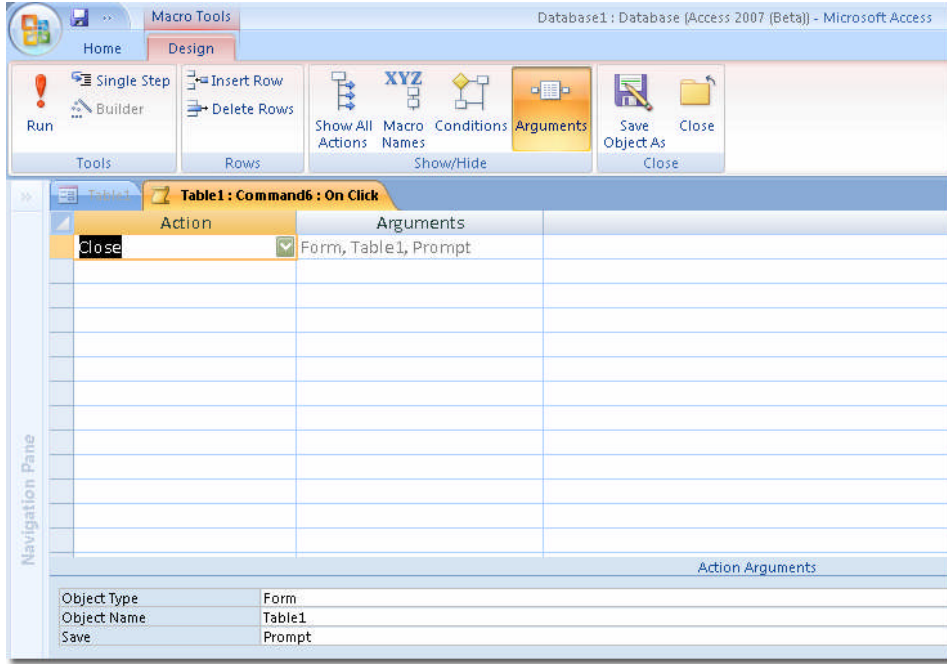
Resim 1. 6: Makronun düğme ile çalıştırılması



Resim 1. 7: Makro atanmış düğmenin Visual Basic kodları



Gelişmiş bir sürüm olan Office 2007'de ise menüler farklı olmasına rağmen, aynı işlemler ve adımları yapabilirsiniz. Eğer bilgisayarınızdaki ofis sürümünü Türkçe değilse sözlük veya çeviri programlarından yardım alarak aynı komutları siz de yapabilirsiniz.



Resim 1. 8: Office 2007’de makro hazırlama

1.3. Eylem Türleri

Fare ve klavye tuşlarının hareketlerini kaydeden “Makro Kaydedicisi” Access’te bulunmasa da, eklenebilen 50’ye yakın “Eylem” türü çoğu faaliyet için yeterlidir. Burada en çok kullanılan eylemlerden bahsedeceğiz.

Tek bir komut için yapılan makroların yapımı kolaydır. Fakat daha karmaşık işlemler için önceden biraz planlama yapılması gerekebilir. Hemen yapımına girişmeden önce makronun bir işlemi, hangi adımları gerçekleştirerek yapacağını düşünmemiz gerekir. Adımları tek tek denemek “hatalı” makro yapılmasını engeller.



Forma ve rapora eklenen nesnelerin kodları, nesne silinse de “Visual Basic Düzenleyicisi”nde dururlar.

1.3.1. Kayıt Gezinme Eylemleri

Kayıtlar içinde gezinme için eklenebilecek eylemler: “Kaydet, KayıtBul, SonrakiniBul, KaydaGit, FiltreyiUygula”

1.3.2. Veri Tabanı Nesneleri Açma Eylemleri

Veri tabanı nesneleri ile ilgili eylemler: “FormAç, SorguAç, TabloAç, RaporAç, VeriErişimSayfasıAç, SaklıYordamıAç, SQLÇalıştır, MakroÇalıştır”

1.3.3. Form Eylemleri

Formlar ile ilgili işlemler: “DenetimeGit, Kapat, İletiKutusu, KumSaati, TaşıBoyutlandır, EkranıKapla, ÖncekiBoyut, SimgeDurumunaKüçült”

1.3.4. Uygulama Eylemleri

Uygulama ile ilgili işlemler: “AraçÇubuğunuGöster, Çıkış, ElektronikTabloAktar, KomutÇalıştır, MakroyuDurdur, MetinAktar, SQLVeriTabanınıAktar, TuşGönder, Veri TabanıAktar, VeriTabanıDosyasıKopyala, YenidenAdlandır”



Soru: Aşağıda bulunan tablodaki boş yerleri uygun bir şekilde doldurunuz:

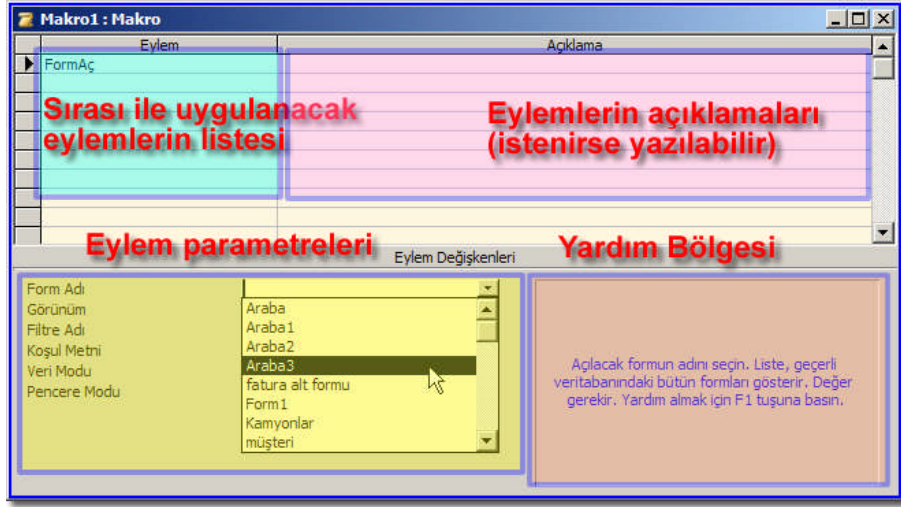
Eylem	Anlamı
Kaydet	
KayıtBul	
FiltreyiUygula	
FormAç	
SorguAç	
RaporAç	
Çıkış	
Kapat	
AraçÇubuğunuGöster	
KomutÇalıştır	

1.4. Makro Tasarım Penceresi

“Makro Tasarım Penceresi” Resim 1,9'daki gibi 4 kısımdan oluşur: “Eylem listesi, açıklama, eylem değişkenleri ve yardım”

Eylem değişkenleri kısmında genel komut ayarları bulunur. Genellikle bu ayarlar yapılmadan diğer komuta geçilmemelidir. Mesela öncelikle “FormAç” komutunun “Form Adı” ayarı yapılmalıdır.

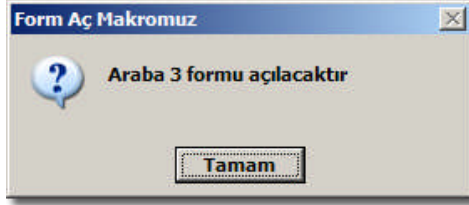
Açıklama kısmında istenirse, komut ile ilgili daha fazla açıklayıcı bilgi sunulabilir. Böylece diğer programcılar programınızı rahat anlayabilir.



Resim 1. 9: Makro düzenleme penceresi



Soru: Resim 1.10'daki gibi bir ileti çıkardıktan sonra, bir formu açtırınız.



Resim 1.10: Bir diyalog kutusu ile kullanıcıya bilgi verilebilir.

1.5. Eylem Değişkenleri

Eylem değişkenleri, eklediğiniz eyleme göre değişir. Mesela “FormAç” eyleminin parametreleri şunlardır: “Form Adı, Görünüm, Filtre Adı, Koşul Metni, Veri Modu, Pencere Modu”



Soru 1: Aşağıda bulunan tablodaki boş yerleri uygun bir şekilde doldurunuz:

Eylem	Eylem Parametresi	Seçenekleri veya örnek bir değer
FormAç	Veri Modu	
FormAç	Pencere Modu	
FormAç	FiltreyiUygula	
İletiKutusu	İleti	
İletiKutusu	Tür	
İletiKutusu	Başlık	



Soru 2: Aşağıdaki uygulamayı yapınız:

- a) Yeni bir makro açınız.
b) Eylem olarak “Metin Aktar” seçiniz.

Aktarım Türü	HTML Verme
Ayrıntı Adı	
Tablo Adı	araba Bir tablonun adını yazınız
Dosya Adı	aktarım.html
Alan Adları Var	Evet
HTML Tablo Adı	
Kod Sayfası	50001: Otomatik seç

- c) Eylem ayarlarını şekildeki gibi yapınız.

- ç) İkinci eylemi “İletiKutusu” seçiniz.

İleti	Aktarım tamamlandı, dosya açılacak
Bip	Evet
Tür	Bilgi
Başlık	HTML Kaydetme

- d) Eylem ayarlarını şekildeki gibi yapınız.

- e) Üçüncü eylemi “UygulamaÇalıştır” seçiniz.

Komut Satırı	explorer "aktarım.html"
--------------	-------------------------

- f) Eylem ayarlarını şekildeki gibi yapınız.

- g) Makroyu kaydediniz.

- ğ) Makroyu çalıştırınız.

- h) Ekran çıktısı tablonuza göre söyle olmalıdır:

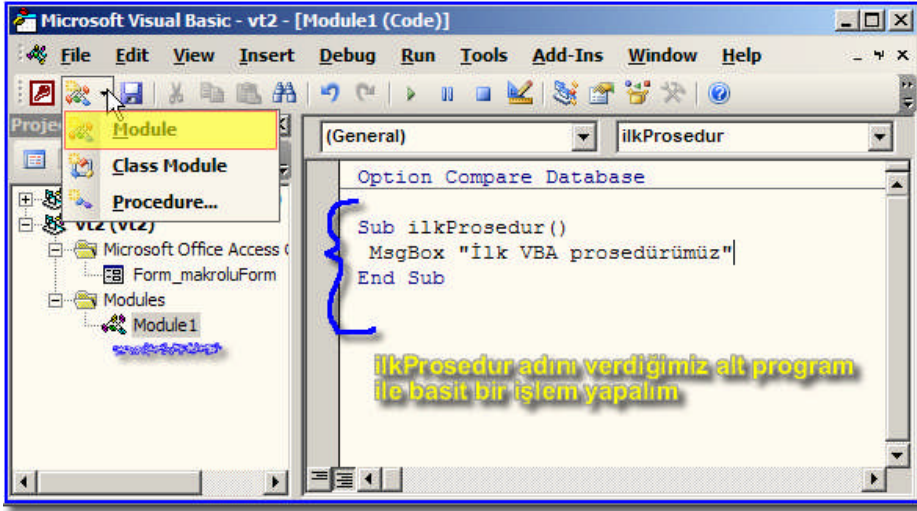
aktarım

ArabaKimlik	Renk	Kapı	Firma	Model	Tarih
1	Mavi	2	Ford	Focus	11/02/2005
2	Sarı	4	Renault	Clio	11/02/2004
12	Siyahsd	4	Renault	Clio	11/02/2006
14	Siyah	4	Ford	Fiesta	11/02/2007
15	Mavi	5	Ford	Focus	11/02/2008

1.6. Kod Yazmak

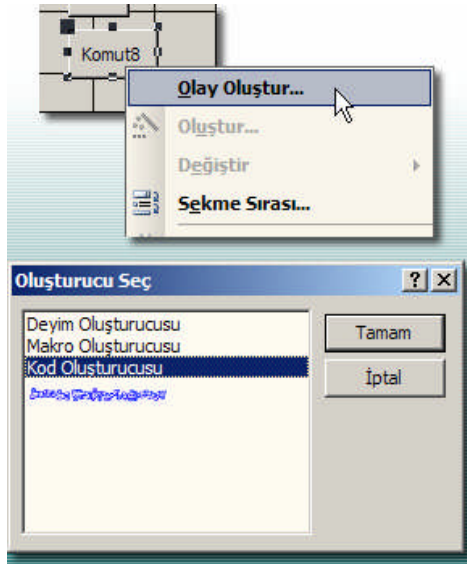
Eylemleri “Makro Tasarım Penceresi”nde yapabildiğimiz gibi daha gelişmiş bir programlama dili olan VBA ile de yapabiliriz. Bir dil kullanarak uygulama yapmak programcı için eylemlere daha fazla seçenek eklenmesini sağlar.

VBA içinde kendi prosedürlerimizi ve fonksiyonlarımızı “Insert*Module” komutu ile yazabiliriz.



Resim 1.11: İçinde bir satır program bulunan bir modül yazıyoruz

“ilkProsedur” adındaki alt programımızı form üzerine eklediğimiz bir komut düğmesinin “Tıklandığında” olayına atayalım:



Resim 1.12: Bir düğmenin kod kısmına geçmek

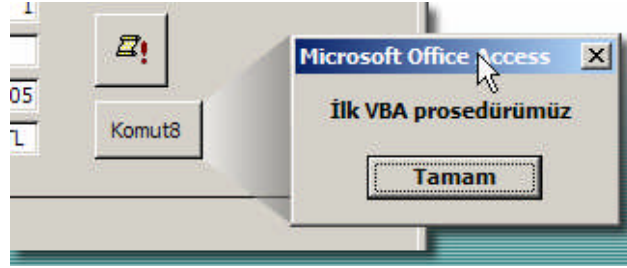
Açılan VB penceresinde bizim için yazılmış hazır iki satır bulunur:

```
Private Sub Komut8_Click()
```

← Düğmenin kodlarımızı buraya yazacağız *

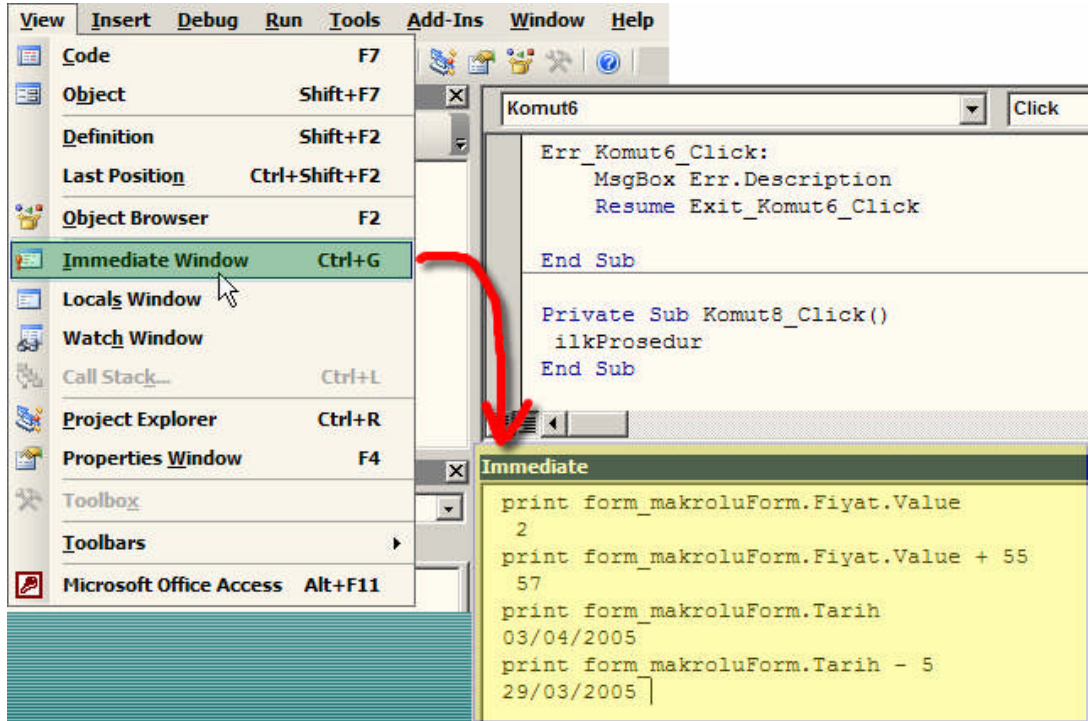
```
End Sub
```

* yazan yere **ilkProsedur** yazalım. Denemek için formunuzu çalıştırabilirsiniz.



Resim 1.13: Kod ile çıkan bir ileti kutusu

VBA içindeki kod hatalarını görmek, değişken değerlerini izlemek ve program hatalarını bulmak, izleme pencereleri ile daha da kolaylaşır. Durak noktası eklemek, adım adım kod çalıştırmak gibi hata yönetim özelliklerini burada kullanabilirsiniz.



Resim 1.14: "Immediate" penceresinde değer izlemek[§]

[§] Kullanılabilecek komutlar kısaca: ?, print, msgbox, DoCmd, atama (a=4 gibi), hesaplama işlemleri (+, -, *, /)

UYGULAMA FAALİYETİ

İşlem Basamakları	Öneriler
➤ Forma bir düğme eklenirken “Denetim Sihirbazları” seçili olmasına dikkat ederek gelen sihirbazdan uygun eylemi seçiniz.	
➤ Sihirbazda düğmenin başlığını ve resmini belirleyerek işlemi tamamlayınız.	
➤ Veri tabanı penceresindeki Makro simgesi tıklanarak “Yeni” komutu veriniz.	
➤ Makro tasarım penceresinden “Makro Adı” sütunundan gereken eylemleri seçiniz.	
➤ Eylemin alt seçeneklerini “Eylem Değişkenleri” kısmından düzenleyiniz.	
➤ Eylemin gerekirse “Açıklama” kısmını açık bir dille doldurunuz	
➤ “Dosya*Kapat” komutu ile makroya isim vererek kaydediniz.	
➤ Makroyu çalıştırmak için, veri tabanı penceresinde makro ismini sağ tıklatıp “Çalıştır” komutu veriniz.	

ÖLÇME VE DEĞERLENDİRME

OBJEKTİF TEST (ÖLÇME SORULARI)

Aşağıdaki sorulardan; sonunda parantez olanlar doğru/yanlış sorularıdır. Verilen ifadeye göre parantez içine doğru ise “D”, yanlış ise “Y” yazınız. Şıklı sorularda uygun şıkkı işaretleyiniz.

1. Microsoft Access’de makro en hızlı şekilde “Makro Kaydedicisi” ile yapılır. (D / Y)
2. Bir ... makronun en basit komut bloğudur.
A) Eylem
B) Deyim
C) Fonksiyon
D) Prosedür
3. Aşağıdaki şıklardan hangisi hatalıdır?
A) “Açıklama” kısmına ek yorumlar yazılabilir.
B) “Eylem Değişkenleri” kısmı ile eylemin çalışması için gereken bilgiler belirtilir.
C) Bir makro birden fazla eylem içerebilir.
D) Makro saf programlama kodlarından oluşur.
4. Bir makro hangisine atanamaz?
A) Düğme
B) Tuş kısa yolu
C) Etiket
D) Araç Çubuğu
5. Bir makrodaki hatayı bulmak için adım adım çalıştırmayı deneyebiliriz. (D / Y)
6. Hangi eylem komutu ile bir rapor açılabilir?
A) FormAç
B) SorguAç
C) RaporAç
D) ÇizimAç

DEĞERLENDİRME

Sorulara verdiğiniz cevap seçeneklerini modül sonunda verilmiş olan cevap anahtarı ile karşılaştırınız.

Yanlış cevapladığınız yada cevap verirken tereddüt ettiğiniz sorularla ilgili konular için bilgi sayfalarına tekrar dönerek eksiklerinizi gideriniz.

ÖĞRENME FAALİYETİ-2

AMAÇ

Makro düzenleyebilecek ve makroyu bir olaya atayabileceksiniz.

ARAŞTIRMA

- Makro eylemelerinden 10 tanesini örnekler ile arkadaşlarınıza sunu olarak açıklayınız.

2. MAKRO DÜZENLEME

Access'te bazı işlemler peş peşe sıra ile yapılan adımlardan oluşur. Mesela, bir tablo açılır (1), belli bir kayıt seçilir (2), kayıt düzenlenir (3), kayıt içindeki bir alan seçilir (4) ve alan içindeki bilgi panoya kopyalanır (5).

Makro düzenleme penceresinde eklenen eylemler, makro çalıştırıldığında sıra ile uygulanır.



Eğer makronun adını "AutoExec"

Ad
AUTOEXEC
Makro1
Makro2

olarak belirlerseniz, Access uygulamanız "açılır açılmaz" bu makronun içindeki satırlar otomatik çalıştırılır.



Makroların isimlendirilmesinde büyük küçük harf farkı yoktur. Örneğin, AutoExec ile AUTOEXEC aynı anlama gelir.


2.1. Eylemlerin Çalışma Yönü

Makro düzenleme penceresinde işlemler "üstten aşağı" yönde sıra ile çalıştırılır. Eylemin altına veya üstüne bir satır eklenebilir, eylemin sırası değiştirilebilir, eylemin olduğu satır silinebilir.



Makro1 : Makro	
Eylem	
1	FormAç
2	EkranıKapla
3	İletiKutusu

Resim 2. 1: Eylemler sırası ile “üstten aşağı” çalışırlar.

2.2. Bir Makroya Daha Fazla Eylem Ekleme

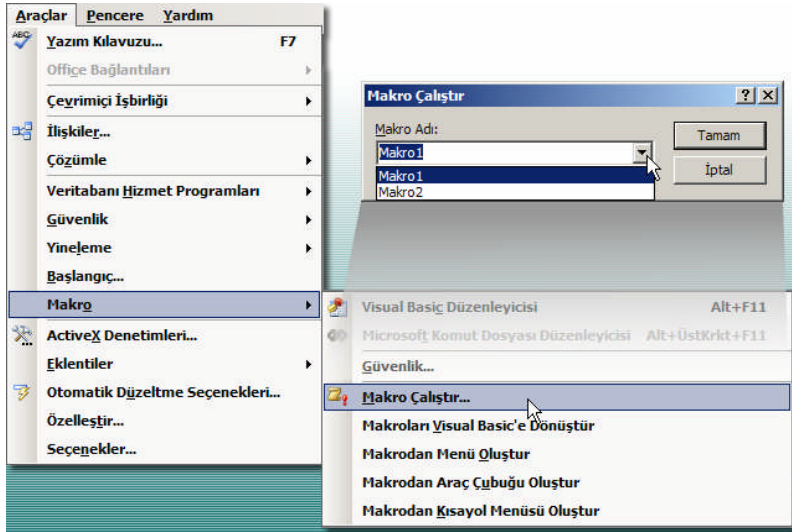
Birden fazla eylem eklemek için Excel veya Word'deki hücre ekleme ile aynı yöntem kullanılır. Ekleme silme işlemleri makro tasarım penceresini kapatmadığımız sürece geri alınabilir  durumdadır.

1. yöntem: “Ekle*Satır” ile imlecin bulunduğu yerin üstüne satır eklenir.

2. yöntem: Makro tasarımı araç çubuğundan  tuşu ile satır eklenebilir. Aynı yerde  komutu ile satır silinebilir.

3. yöntem: İstenen satıra sağ tıklatılıp “Satır Ekle” komutu verilebilir.

2.3. “Makro Çalıştır” Penceresi



Resim 2. 2: Makro çalıştırma penceresi

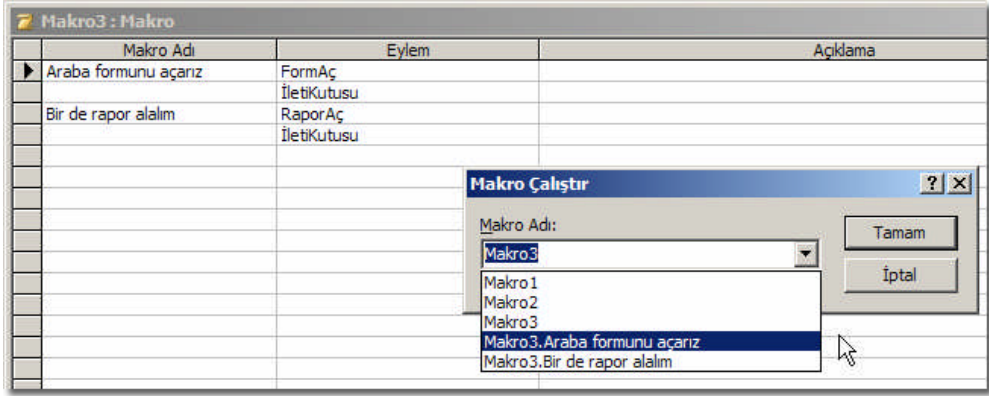
Makroları kısa yoldan çalıştırmak için “Araçlar*Makro*Makro Çalıştır...” penceresi kullanılabilir.

2.4. Makrolarda Gruplandırma

Birçok makro ile uğraşırken makroları gruplayıp yönetimlerini kolaylaştırabiliriz. “Makro grubu” birbiri ile alakalı makroları tek bir makro gibi çalışmalarıdır.

Bir makro grubu oluşturduğumuzda, makronun başlangıç ve bitimini belirten benzersiz (tekil - *unique*) bir isimlendirme yapılır. Bunu yapabilmek için “Görünüm*Makro Adları” seçili olmalıdır.

“Makro Çalıştır” penceresinde Resim 2.3’teki gibi “Makro3” adı altında “Makro3.Araba formunu açarız” ve “Makro3.Bir de rapor alalım” adlı makrolar gözükür. Bu penceredeki alt gruptan bir ismi seçerek çalıştırabiliriz.



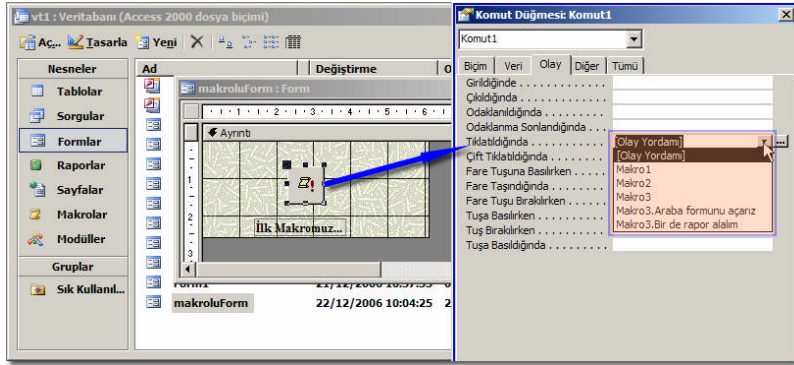
Resim 2. 3: Makro grubu yapmak

2.5. Makroyu Bir Olaya Atama

Önceki faaliyette bir makroyu form üzerindeki bir düğmenin tıklanma anına atamıştık. Bir makro başka nesnenin olaylarına atanabilir. Atanabilen olayları ilk olarak “Veri Tabanında Formlar ve Raporlar” modülünde görmüştük.

Makroları “Makro Çalıştır” penceresinden veya veri tabanı penceresinden çalıştırmak Access’te yaptığınız programı kullanan insanlar için zor olabilir. Bu sebeple makrolar, nesnelere (genellikle düğmelere) atanırlar. Bir düğme tıklanması ile makro aktif hale gelir.

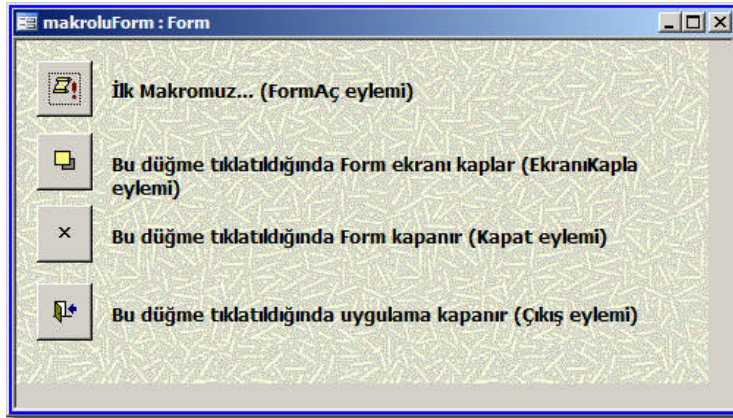
Komut düğmesinin “Tıklatıldığında” olayından başka, “Fare Tuşuna Basılırken, Tuşa Basılırken, Odaklanıldığında” gibi makro atanabilen olayları da vardır.



Resim 2. 4: Bir form nesnesinin bir olayına makro eklemek



Soru 1: Aşağıdaki formun tasarımını ve makrolarını hazırlayınız:



Resim 2. 5: Örnek çalışma sorusu



Soru 2: Aşağıda bulunan tablodaki boş yerleri uygun bir şekilde doldurunuz:

Olay	Tanımı
Güncelleştirme Öncesinde	
Güncelleştirme Sonrasında	
Değiştğinde	
Girildiğinde	
Çıkıldığında	
Odaklanıldığında	
Odaklanma Sonlandığında	

Tıklatıldığında	
Çift Tıklatıldığında	
Fare Tuşuna Basılırken	
Fare Taşındığında	
Fare Tuşu Bırakılırken	
Tuşa Basılırken	
Tuş Bırakılırken	
Tuşa Basıldığında	

2.6. Diğer Önemli Makro Eylemleri

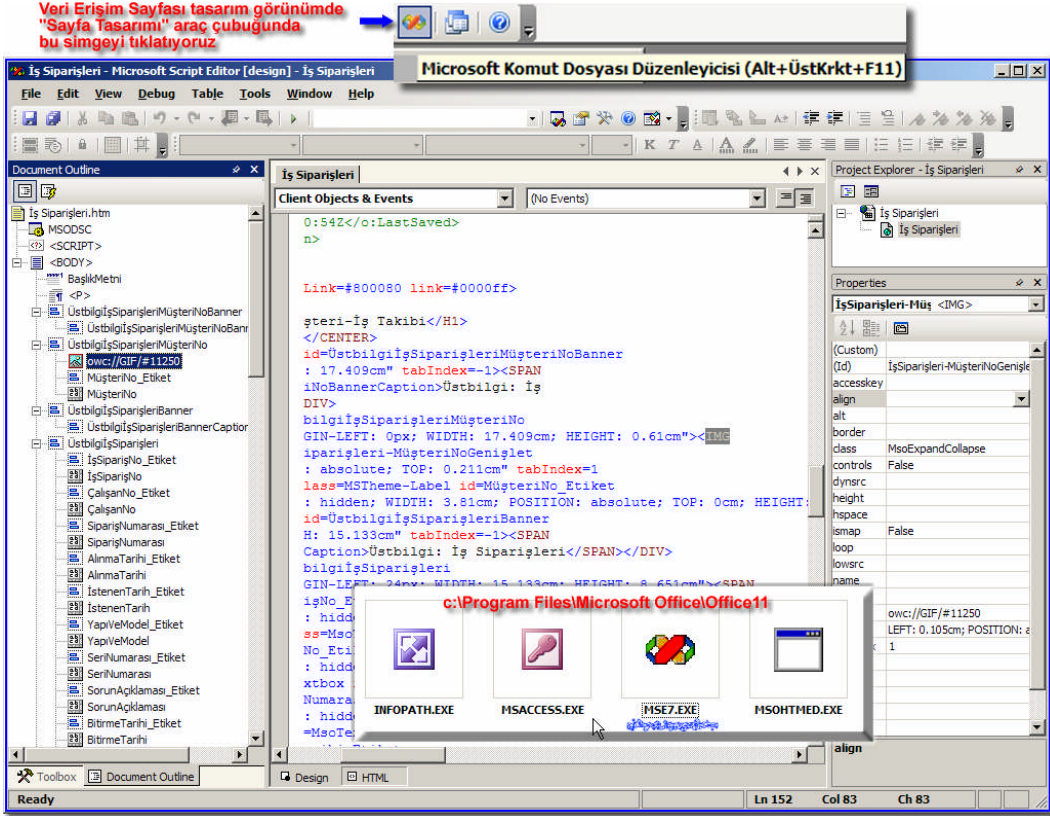
Önceki konudaki olayların dışında hemen her an için yapılmış olaylar vardır. Mesela, formun özelliklerin baktığımızda “Açıldığında, Yüklendiğinde, Yeniden Boyutlandırıldığında, Hata Durumunda, Seçim Değiştiğinde” gibi olayları varken, rapor özelliklerinde de benzer olarak “Açıldığında, Kapandığında, Etkinleştiğinde, Devreden Çıkma, Veri Yokken, Sayfaya Gelindiğinde ve Hata Durumunda” olaylarını görüyoruz.

Veri erişimli sayfalarda makro atanabilen olay yoktur. Fakat Resim 2.6’daki gibi “Microsoft Komut Dosyası Düzenleyicisi” yardımcı programı ile bazı düzenleme işlemleri kolaylaştırılmıştır. Bu yardımcı programı “c:\Program Files\Microsoft Office\Office11” içinde “mse7.exe” ile ayrı bir şekilde çalıştırabiliriz.



“MS Script Editor” kullanabilmek için HTML, VBScript ve ASP gibi web programlama dilleri iyi bilmek gereklidir.

Veri Erişim Sayfası tasarım görünümde
"Sayfa Tasarımı" araç çubuğunda
bu simgeyi tıklıyoruz



Resim 2. 6: "Microsoft Komut Dosyası Düzenleyicisi"ni açmak

UYGULAMA FAALİYETİ

İşlem Basamakları	Öneriler
➤ Sık kullanılan, peş peşe yapılan eylemleri belirleyiniz.	
➤ Makronun simgesine tıklatıp “Tasarla” komutu vererek düzenleyiniz.	
➤ Sırasına göre eylemleri “Eylem” sütununa ekleyiniz.	
➤ Bir makro içine birden fazla birbiri ile ilgili makro ekleyiniz.	
➤ Denemek için tasarım görünümünde araç çubuğundaki “Çalıştır” komutunu veriniz.	
➤ Makroyu “Dosya*Kaydet” ile kaydettikten sonra, makronun atanacağı formu tasarım görünümünde açınız.	
➤ Uygun nesneye sağ tıklayıp “Özellikler*Olay” kısmında belli bir olaya makro seçiniz.	

ÖLÇME VE DEĞERLENDİRME

OBJEKTİF TEST (ÖLÇME SORULARI)

Aşağıdaki sorulardan; sonunda parantez olanlar doğru/yanlış sorularıdır. Verilen ifadeye göre parantez içine doğru ise “D”, yanlış ise “Y” yazınız. Şıklı sorularda uygun şıkkı işaretleyiniz.

1. “Eylemler” ile ilgili hangisi hatalıdır?
A) Her satırda bir eylem vardır.
B) Eylemler üstten aşağı doğru çalıştırılır.
C) Bir eylemi silmek için “Düzen*Satır Sil” komutu verilir.
D) Bir makro F5 kısa yolu ile çalıştırılabilir.
2. Bir makro adından dolayı davranışı ve kullanımı değişebilir. (D / Y)
3. Bir makro istenirse Visual Basic’e dönüştürülebilir. (D / Y)
4. Bir “makro” ile ilgili hangi seçenek hatalıdır?
A) Makronun ismi tekil olmalıdır.
B) Bir makronun içine birden fazla makro yazılabilir.
C) Makronun isminde “boşluk” karakteri kullanılamaz.
D) Program yazarak oluşturulan kod da makrodur.
5. Hangi olay bir “komut düğmesi” için değildir?
A) Tıklatıldığında
B) Değiştirildiğinde
C) Çift tıklatıldığında
6. Hangi olay bir “form”un özelliklerinde bulunmaz?
A) Ekleme öncesinde
B) Açıldığında
C) Veri yokken

DEĞERLENDİRME

Sorulara verdiğiniz cevap seçeneklerini modül sonunda verilmiş olan cevap anahtarı ile karşılaştırınız.

Yanlış cevapladığınız yada cevap verirken tereddüt ettiğiniz sorularla ilgili konular için bilgi sayfalarına tekrar dönerek eksiklerinizi gideriniz.

ÖĞRENME FAALİYETİ-3

AMAÇ

Şartlı deyimler oluşturabileceksiniz.

ARAŞTIRMA

- İçinde çok sayıda kayıt bulunan bir tablodan (Northwind örnek uygulamasından seçebilirsiniz) “belli” kayıtlar nasıl seçilebilir?
- Access’in sunduğu imkânları arkadaşlarınız ile tartışınız. Bir liste haline getiriniz.

3. ŞARTLI DEYİMLER

Şimdiye kadar yaptığımız makrolar tablo veya sorgu içindeki değerlere göre değişen durumlara bakmıyor, sadece eylemi ne ise onu yapıyordu. Bu faaliyette, belli şartlarda “farklı davranabilen” makrolar yazacağız.

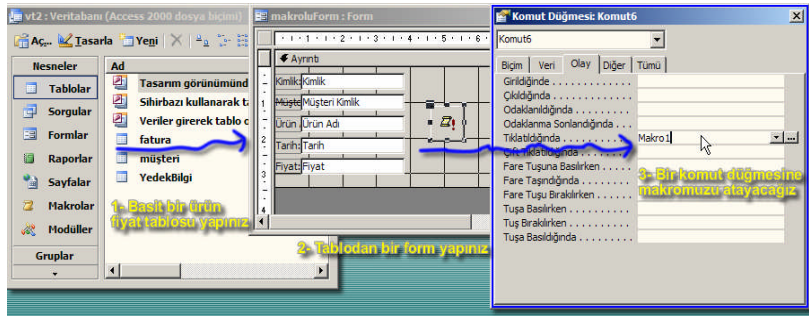
3.1. Deyim Oluşturucusu Penceresi

“Deyim Oluşturucusu” penceresi ile ilk kez “Veri Tabanında Tablolar” modülünde karşılaşmıştık. Kullanımı kolay olan bu pencere ile daha gelişmiş makrolar yazabiliriz.

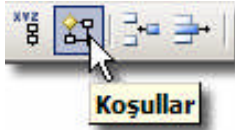
Örnek Çalışma



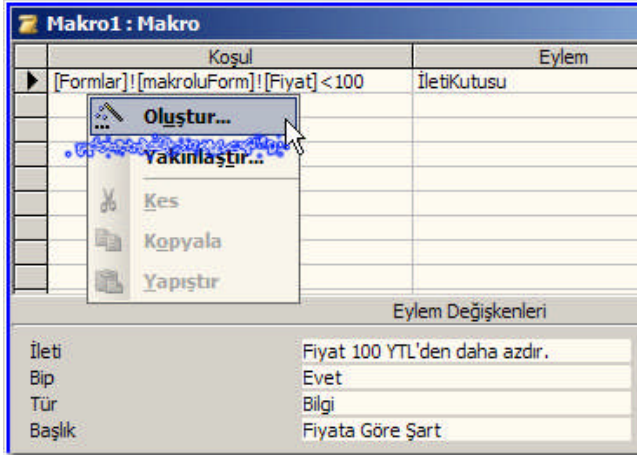
Önceden yaptığınız bir veri tabanından faydalanabilirsiniz. Veri tabanında bir tablo ve bir de buna bağlı form var olmalıdır. Mesela, fiyat listesini saklayan tablonun basit olarak formunu oluşturabilirsiniz.



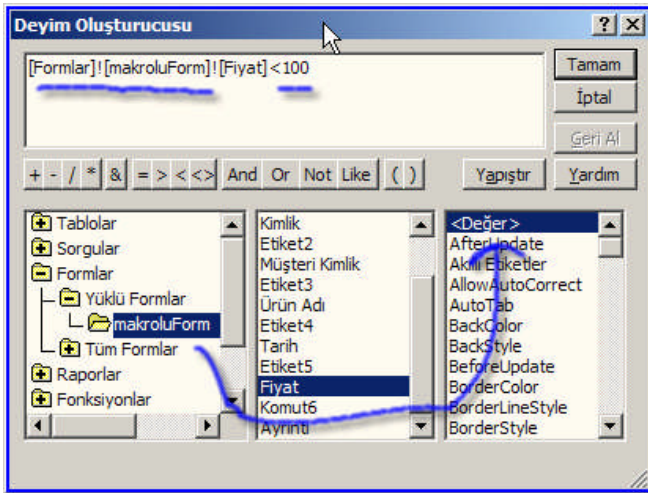
Resim 3. 1: Fiyat alanına sahip bir tablo ve tablonun formunu hazırlayınız.



a) Makro tasarım penceresinde iken, araç çubuğundan “Koşullar”^{**} “Koşul” sütununu açıyoruz.



b) İstenen eylemin “Koşul” sütununda uygun satıra sağ tıklatınız, “Oluştur...” komutu veriniz.



c) Sırası ile istenen bir alanın değeri seçilir. Büyüklük veya eşitlik şartı eklenir.

ç) Makro düğmeye atanmak üzere kaydedilir. Örneğin “Makro1” olarak isimlendirilir.

^{**} “Görünüm*Koşullar” seçilerek de aynı işlem yapılabilir.

3.2. Koşullu Deyimin Parçaları

Önceki örnek çalışmadaki koşul kısmını inceleyelim:

[Formlar]![makroluForm]![Fiyat] < 100

1 2 3 4

- 1= Veri tabanı nesnesi (Formlar veya Raporlar yazılır.)
- 2= Nesne adı (makroluForm form nesnemiz)
- 3= Kontrol adı (Fiyat değer saklayan formdaki nesne)
- 4= Şart deyimi (<, >, <>, =, >=, <= gibi)

3.3. Klavye Kısa Yolları Yapmak

Uygulamamıza kendi kısa yol tuş basımlarını ekleyebiliriz. Access içinde kullanılabilen tuş basımları Visual Basic dili ile benzerlik gösterir.

Eğer sistem için çok gerekli bir kısa yol değiştirilmeye çalışılırsa, Access bir uyarı vermez. Fakat örneğin, Ctrl+Z “geri al” eyleminde kullanıldığından, başka bir komut için kullanılmamalıdır.

Örneğin, Ctrl ile L tuşları basılı iken bir makro çalıştırılabilir. Tuş basımlarının hepsini “**AutoKeys**” adı verilen makro içine kaydederiz.



Makro tasarım penceresinde iken, “Makro Tasarımı” araç çubuğundaki “Makro Adları” seçeneği aktif edilir. Açılan “Makro Adı” sütununa kısa yol tuşu yazılır.

Yazılabilecek klavye kısa yolları şu şekildedir:

Tuş	Yazımı
Ctrl + Tuş	^ (Örneğin, ^E'nin anlamı <Ctrl> + <E>)
Shift + Tuş	+ (Örneğin, +E'nin anlamı <Shift> + <E>)
Alt	% (Örneğin, %E'nin anlamı <Alt> + <E>)
Enter	{ENTER}
Esc	{ESC}
Tab	{TAB}
Insert, Delete	{INSERT} veya {INS}, {DELETE} veya {DEL}
Page Down, Page Up	{PGDN}, {PGUP}
Home, End	{HOME}, {END}
Ok Tuşları	{UP}, {DOWN}, {LEFT}, {RIGHT}
Caps Lock	{CAPSLOCK}
Fonksiyon Tuşları	{F1}, {F2}, {F3} ... gibi

Fonksiyon tuşları ve Ctrl ile yapılan tuş basımlarını çok sık çağrılan makroların kısa yollarına atayabiliriz.



Soru: Aşağıda bulunan tablodaki örnekleri yapınız:

Eylem	Kısa yol tuşları
Hesap makinesi programını çalıştır	Ctrl+H
Tüm makroları durdur	F9
Bir tabloyu HTML olarak aktar	Ctrl+Shift+H
Yardım için bir ileti kutusu göster	Ctrl+F1
Bir tablo içinde belli bir kaydı bul	Ctrl+G
Uygulamadan çık	F10



UYGULAMA FAALİYETİ

İşlem Basamakları	Öneriler
➤ “Görünüm*Koşullar” komutunu veriniz.	
➤ Makro tasarımı araç çubuğundaki “Oluştur” komutunu veriniz.	
➤ “Deyim oluşturucusu” ile istenen şartı oluşturunuz.	
➤ Bir makronun düğmeye basarak çalıştırılması yerine, klavyeden kısa yol ile çalışması için “Otomatik tuşlar - AutoKeys” adında makro oluşturunuz.	
➤ Makronun “Makro adı” hücresine klavye kısa yolu yazınız.	

ÖLÇME VE DEĞERLENDİRME

OBJEKTİF TEST (ÖLÇME SORULARI)

Aşağıdaki sorulardan; sonunda parantez olanlar doğru/yanlış sorularıdır. Verilen ifadeye göre parantez içine doğru ise “D”, yanlış ise “Y” yazınız. Şıklı sorularda uygun şıkkı işaretleyiniz.

1. Bir makroya şart eklemek için tasarım görünümünde hangi sütun açık olmalıdır?
A) Koşullar 
B) Makro Adları 
2. Hangi simge “koşul” içinde kullanılamaz?
A) < B) <>
C) != D) =
3. Bir koşul içinde “**And** ve **Or**” deyimlerini kullanarak birden fazla şart ekleyebiliriz.
(D / Y)
4. Bir koşul ifadesini, programlama dillerinde kullanılan “**If Then**”e benzetebiliriz.
(D / Y)
5. Aşağıdaki kısa yollardan hangisi Access’te kullanılmaya uygundur?
A) Ctrl+Z B) Ctrl+T
C) Ctrl+X D) Ctrl+Alt+Del
6. Tuş basımları hangi makro adı altında saklanırlar?
A) AutoExec
B) AutoKeys
C) AutoKey

DEĞERLENDİRME

Sorulara verdiğiniz cevap seçeneklerini modül sonunda verilmiş olan cevap anahtarı ile karşılaştırınız.

Yanlış cevapladığınız yada cevap verirken tereddüt ettiğiniz sorularla ilgili konular için bilgi sayfalarına tekrar dönerek eksiklerinizi gideriniz.

MODÜL DEĞERLENDİRME

PERFORMANS TESTİ (YETERLİK ÖLÇME)

Modül ile kazandığınız yeterliği, öğretmeniniz işlem basamaklarına göre 0 ile 5 puan arasında olacak şekilde değerlendirecektir.

DEĞERLENDİRME ÖLÇÜTLERİ	Puan
1. Forma bir düğme ekleme	
2. “Denetim Sihirbazı”ndaki uygun eylemi seçme	
3. Sihirbazda düğmenin başlığını ve resmini belirleyerek işlemi tamamlama	
4. Veri tabanı penceresindeki Makro simgesi tıklanarak “Yeni” komutu verme	
5. Makro tasarım penceresinden “Makro Adı” sütunundan gereken eylemleri seçme	
6. Eylemin alt seçeneklerini “Eylem Değişkenleri” kısmından düzenleme	
7. Eylemin gerekirse “Açıklama” kısmını açık bir dille doldurma	
8. Makroyu çalıştırmak için, veri tabanı penceresinde makro ismini sağ tıklatıp “Çalıştır” komutu verme	
9. Sık kullanılan, peş peşe yapılan eylemleri belirleme	
10. Makronun simgesine tıklatıp “Tasarla” komutu vererek düzenleme	
11. Sırasına göre eylemleri “Eylem” sütununa ekleme	
12. Bir makro içine birden fazla birbiri ile ilgili makro ekleme	
13. Makroyu denemek için tasarım görünümünde araç çubuğundaki “Çalıştır” komutunu verme	
14. Makroyu “Dosya*Kaydet” ile kaydettikten sonra, makronun atanacağı formu tasarım görünümünde açma	
15. Uygun nesneye sağ tıklatıp “Özellikler*Olay” kısmında belli bir olaya makro seçme	
16. “Görünüm*Koşullar” komutunu verme	
17. Makro tasarımı araç çubuğundaki “Oluştur” komutunu verme	
18. “Deyim oluşturucusu” ile istenen şartı oluşturma	
19. Bir makronun düğmeye basarak çalıştırılması yerine, klavyeden kısa yol ile çalışması için “Otomatik tuşlar - AutoKeys” adında makro oluşturma	
20. Makronun “Makro adı” hücrelerine klavye kısa yolu yazma	
Toplam (en fazla 100 puan olabilir)	

DEĞERLENDİRME

Yaptığınız değerlendirme sonucunda eksikleriniz varsa öğrenme faaliyetlerini tekrarlayınız.

Modülü tamamladınız, tebrik ederiz. Öğretmeniniz size çeşitli ölçme araçları uygulayacaktır, öğretmeninizle iletişime geçiniz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ-1'İN CEVAP ANAHTARI

1	Y
2	A
3	D
4	C
5	D
6	C

ÖĞRENME FAALİYETİ-2'NİN CEVAP ANAHTARI

1	D
2	D
3	D
4	C
5	B
6	C

ÖĞRENME FAALİYETİ-3'ÜN CEVAP ANAHTARI

1	A
2	C
3	D
4	Y
5	B
6	B

Cevaplarınızı cevap anahtarları ile karşılaştırarak kendinizi değerlendiriniz.

ÖRNEK ÇALIŞMA

Makrolar sayesinde belli bir konuda proje (Windows uygulaması) rahatlıkla yapabiliriz. Makro seçenekleri çoğu işlemin yapımında yararlı olsa da, programcılar “kod yazarak” yaptıkları programa tam olarak hâkim olmak ister.

VBA (*Visual Basic for Applications*) ile Access içinde bir nesnenin bir olayına program kodları yazabiliriz. Bunu yapabilmek için iyi bir şekilde algoritma ve akış şeması kullanımını bilmeliyiz. Müşterinin talep ettiklerini yerine getirmek için istenenleri iyi anlamalı ve değerlendirmeliyiz.

Örnek olarak spor takımlarının haftalık maç listesini hazırlayan bir fikstür programı yazalım. Programın yapımından önce aşağıdaki tablo yapılarını hazırlayalım. Daha sonra da formları ve kodlarını yazalım.

En az iki takım veya oyuncu seçilerek “Seçili Oyunculardan Fikstür Hazırla” düğmesine basılır. Daha sonrada hazırlanan maçlar takip edilerek sonuçları kaydedilir.

Temelde oyuncular seçilen bir joker takımın oyununa göre dağıtılıyor. Biz burada son takımı joker takım olarak belirliyoruz. Önce bu takımın oynayacağı maçlar belli bir döngü ile bulunur. (1, 10, 2, 11, 3, 12, 4, 14, ... gibi) İlk hafta joker takımımız 1. takım ile ikinci hafta 10. takım ile maç yapacaktır. Bu şekilde diğer takımların maçları hazırlanır.

maçlar : Tablo		
	Alan Adı	Veri Türü
🔑	Kimlik	Otomatik Sayı
	IlkOyuncuVeyaTakim	Metin
	IkinciOyuncuVeyaTakim	Metin
	MacSonucuIlkOyuncu	Sayı
	MacSonucuIkinciOyuncu	Sayı
	MacTuru	Metin
	MacTarihi	Tarih/Saat
	MacYeri	Metin
	Kazanan	Metin
	MacHaftasi	Metin
	GrupNumarasi	Metin

oyuncular : Tablo		
	Alan Adı	Veri Türü
🔑	Kimlik	Otomatik Sayı
	OyuncuAdiVeyaTakimAdi	Metin
	Secili	Evet/Hayır
	OyunTuru	Metin
	Puan	Sayı
	OyuncununGrubu	Sayı
	Sinifi	Metin

oyunTurleri : Tablo		
	Alan Adı	Veri Türü
🔑	Kimlik	Otomatik Sayı
	OyunTuru	Metin

Tabloların yapıları

Alan Adı	Veri Türü
Kimlik	Otomatik Sayı
OyuncuAdiVeyaTakimAdi	Metin
Secili	Evet/Hayir
OyunTuru	Metin
Puan	Sayı
OyuncununGrubu	Sayı
Sinifi	Metin

Genel	Arama
Denetimi Görüntüle	Açılan Kutu
Satır Kaynak Türü	Tablo/Sorgu
Satır Kaynağı	SELECT oyunTurleri.OyunTuru FROM oyunTurleri;
İlişkili Sütun	1
Sütun Sayısı	1
Sütun Başları	Hayır
Sütun Genişlikleri	
Satırları Listele	8
Liste Genişliği	Otomatik
Listeye Sınır	Hayır

“oyuncular” tablosundaki “OyunTuru” bilgilerini “oyunTurleri” tablosundan alır.

Fikstür Yönetim Programı

Fikstür Yönetim Programı

Komut21

Seçili	Sınıfı	Oyuncu Adı	Oyun Türü	Oyuncu Grubu
<input checked="" type="checkbox"/>	T12C	Şadullah	Masa Tenisi	1
<input checked="" type="checkbox"/>	M11G	Veli Gelme	Masa Tenisi	1
<input checked="" type="checkbox"/>	T10C	Tako Bako	Masa Tenisi	1
<input checked="" type="checkbox"/>	T11C	Altan Ural	Masa Tenisi	1
<input checked="" type="checkbox"/>	A10C	Cem Git	Masa Tenisi	1
<input checked="" type="checkbox"/>	A11C	Eren Gel	Masa Tenisi	2
<input checked="" type="checkbox"/>	A12C	Ayşe	Masa Tenisi	2
<input checked="" type="checkbox"/>	M11F	Ali Gel	Masa Tenisi	2
<input checked="" type="checkbox"/>	T12C	Mehmet	Masa Tenisi	2
<input checked="" type="checkbox"/>		T12C	Futbol	3
<input checked="" type="checkbox"/>		A12C	Futbol	3
<input checked="" type="checkbox"/>		T10C	Futbol	3

Seçili Oyunculardan Fikstür Hazırla! Maçları Listele Puan Durumu +

Komut8 Komut9 Komut11 Komut10

“oyuncular” tablosuna bağlı olan ana formun tasarımı (Düğme isimlerine kod yazımında dikkat edilmelidir.)

Maç Haftası	1. Oyuncu	2. Oyuncu	1. Sonuç	2. Sonuç	Maç Türü	Maç Tarihi	Maç Yeri	Kazanan
1	Tako Bal	Sadullal	0	0	Masa Teni			
1	Altan Ur	Cem Git	0	0	Masa Teni			
2	Veli Gel.	Tako Bal	0	0	Masa Teni			
2	Altan Ur	Sadullal	0	0	Masa Teni			
3	Veli Gel.	Altan Ur	0	0	Masa Teni			
3	Cem Git	Sadullal	0	0	Masa Teni			
4	Veli Gel.	Cem Git	0	0	Masa Teni			
4	Tako Bal	Altan Ur	0	0	Masa Teni			
5	Veli Gel.	Sadullal	0	0	Masa Teni			
5	Tako Bal	Cem Git	0	0	Masa Teni			
*			0	0				

Kayıt: 1 / 10

Maçları silmek için Ctrl+A ile tümünü seçip, ardından Del tuşuna basınız...

Maçları listeyen form ("maclarveSonuclari" adında)

Üç formun isimleri sırası ile:
OyunculardanFikstürYapma, maclarveSonuclari, PuanDurumu

Ana formda (OyunculardanFikstürYapma) iken yazılacak kodlar aşağıdaki gibidir:

```

Option Compare Database
Option Base 1
Option Explicit

Private Sub Komut10_Click()
    DoCmd.Close
End Sub

Private Sub Komut9_Click()
    DoCmd.OpenForm "maclarveSonuclari"
End Sub

Private Sub Komut21_Click()
    DoCmd.Quit
End Sub

Private Sub Komut11_Click()
    DoCmd.OpenForm "PuanDurumu", , , "[secili]=true"
End Sub

```

```

Private Sub Komut8_Click()

    Dim kacOyuncuSecili, oynayacaklar, turu, grubu, filtreDeyimi, i, j, k

    If Me.Dirty Then Me.Dirty = False

    turu = Me.OyunTuru
    grubu = Me.OyuncununGrubu
    filtreDeyimi = "[secili]=True and [oyunturu]='" & turu & "' and [oyuncunungrubu]='" & grubu

    kacOyuncuSecili = DCount("*", "oyuncular", filtreDeyimi)
    oynayacaklar = kacOyuncuSecili
    If kacOyuncuSecili Mod 2 = 1 Then oynayacaklar = oynayacaklar + 1

    If kacOyuncuSecili > 1 Then
        If MsgBox(kacOyuncuSecili & " Oyuncu/takım Seçilidir. " & vbCrLf & vbCrLf & "Fikstür Hazırlansın Mı?", vbYesNo + vbDefaultButton2, "Fikstür") = vbYes Then
            Dim rs, rs2 As DAO.Recordset
            Dim haftaMacSayisi, haftaSayisi, joker, sayac As Integer
            Dim oyuncuListesi() As String
            Dim macListesi(), jokerinListesi(), macSiraListesi()
            Dim MacTuru, oyuncuListesif As String

            ReDim oyuncuListesi(oyunacaklar, 3)

            Set rs = CurrentDb().OpenRecordset("oyuncular", dbOpenDynaset)
            Set rs2 = CurrentDb().OpenRecordset("maclar", dbOpenDynaset)

            With rs
                .FindFirst filtreDeyimi
                For i = 1 To kacOyuncuSecili
                    oyuncuListesi(i, 1) = i
                    oyuncuListesi(i, 2) = 0
                    oyuncuListesi(i, 3) = rs![OyuncuAdiVeyaTakimAdi]
                    MacTuru = rs![OyunTuru]
                    oyuncuListesif = oyuncuListesif & vbCrLf & i & ". " & oyuncuListesi(i,
3)
                .FindNext filtreDeyimi
                Next
            End With

            If kacOyuncuSecili Mod 2 = 1 Then
                oyuncuListesi(oyunacaklar, 1) = oynayacaklar
                oyuncuListesi(oyunacaklar, 2) = 0
                oyuncuListesi(oyunacaklar, 3) = "silinecek"
            End If

            rs.Close

            If oynayacaklar Mod 2 = 0 Then haftaSayisi = oynayacaklar - 1 Else
            haftaSayisi = oynayacaklar
            If oynayacaklar Mod 2 = 0 Then haftaMacSayisi = oynayacaklar \ 2 Else
            haftaMacSayisi = (oyunacaklar - 1) \ 2
            MsgBox "Fikstür Tamamlandı!" & vbCrLf & vbCrLf & "Oynayacakların
Listesi:" & vbCrLf & oyuncuListesif & vbCrLf & vbCrLf & "Oynanacak Hafta Sayısı:
" & haftaSayisi & vbCrLf & vbCrLf & "Bir Haftadaki Maç Sayısı: " & haftaMacSayisi
& vbCrLf & vbCrLf & "Toplam Yapılacak Maç Sayısı: " & (haftaSayisi *
haftaMacSayisi)

            ReDim macListesi(haftaSayisi, haftaSayisi, 2) ' maçların dizisi
            ReDim macSiraListesi(oyunacaklar - 1, oynayacaklar - 1) ' maç sıra listesi
            ReDim jokerinListesi(oyunacaklar - 1)

            joker = Int(oyuncuListesi(oyunacaklar, 1))

```



```

sayac = 0
For i = 1 To oynayacaklar - 1
    ' jokerin maçları tekler
    If i Mod 2 = 1 Then
        sayac = sayac + 1
        jokerinListesi(i) = sayac
    End If
Next i
For i = 1 To oynayacaklar - 1
    ' jokerin maçları çiftler
    If i Mod 2 = 0 Then
        sayac = sayac + 1
        jokerinListesi(i) = sayac
    End If
Next i

sayac = 0
Dim yeri
' oyuncu sıralarına göre maç listesi
For j = 1 To haftaSayisi ' satır döngüsü

    For k = 1 To haftaSayisi
        If k = j Then
            yeri = k: Exit For
        End If
    Next k

        For i = 1 To haftaSayisi ' sütun döngüsü
            If i + sayac <= haftaSayisi Then
                If i = yeri Then
                    macSiraListesi(j, i + sayac) = joker 'joker ile oynarsa
                Else
                    macSiraListesi(j, i + sayac) = i
                End If
            Else
                If i = yeri Then
                    macSiraListesi(j, ((i + sayac) Mod joker) + 1) = joker
                Else
                    macSiraListesi(j, ((i + sayac) Mod joker) + 1) = i
                End If
            End If
        Next i
        sayac = sayac + 1
    Next j

    For i = 1 To haftaSayisi
        For j = 1 To haftaSayisi

            If i < macSiraListesi(i, j) Then
                macListesi(j, i, 1) = i
                macListesi(j, i, 2) = macSiraListesi(i, j)
            End If

        Next j
    Next i

    For j = 1 To haftaSayisi ' maclar tablosuna kayıt
        For i = 1 To haftaSayisi

            If macListesi(j, i, 1) <> "" Then
                If oyuncuListesi(macListesi(j, i, 1), 3) <> "silinecek" Then
                    If oyuncuListesi(macListesi(j, i, 2), 3) <> "silinecek" Then
                        rs2.AddNew
                        rs2![IlkOyuncuVeyaTakim] = oyuncuListesi(macListesi(j, i, 1), 3)
                    End If
                End If
            End If
        Next i
    Next j

```

```

3)          rs2![IkinciOyuncuVeyaTakim] = oyuncuListesi(macListesi(j, i, 2),

          rs2![MacHaftasi] = j
          rs2![MacTuru] = MacTuru
          rs2.Update
          End If
          End If
          End If

          Next i
          Next j

          rs2.Close
          Set rs = Nothing
          Set rs2 = Nothing

          End If ' vbYes
          Set rs = Nothing
          End If ' oynayacaklar > 1

          End Sub

```



Puan Durumu kısmı hesaplanmamıştır. Siz de eksik kısımları benzer şekilde geliştirebilirsiniz.

ÖNERİLEN KAYNAKLAR

- www.databasedev.co.uk/macro_tips.html
- www.er.uqam.ca/merlin/xw999998/eng/eaccm.htm
- www.excel.web.tr/archive/index.php/f-58.html
- www.fixturelist.com
- www.mcsetr.com/ornek.html
- mis.bus.sfu.ca/tutorials/MSAccess/HTML/main.book.htm
- office.microsoft.com/tr-tr/access/HP051868041055.aspx
- www.sorucevap.com/bilisimteknolojisi/veritabanlari/access
- www.teknomerkez.net/makale.asp?konu=153
- yunus.sourceforge.net

KAYNAKÇA

- **BAĞRIYANIK** Tarık, **Programlama Ders Notları ve Uygulamalı Genel Programlama Kitabı** (www.yunus.projesi.com)
- Microsoft Computer Courseware, **Access 2003 Student Edition**, Custom Guide, Minneapolis, 2004.