

dBaseIII

A.NusretGÜÇLÜ

1. GİRİŞ

Bu yazıda, "relational database" diye adlandırılacak bir veritabanı oluşturma paket programı olan dBase III tanıtılacak, bu programla yapılabilecek işlerden söz edilecektir. Daha detaylı bilgi için kullanım kılavuzuna bakılmalıdır.

Bağıntılı veritabanı, çeşitli sıra ve dizelerden oluşmuş bir dizi tablodur. Bu tabloların herbiri bir bağıntıdır. Sıralar kayıt, dizeler ise alan adıdır. Her kaydın bir kayıt numarası vardır. Her alana da ayrıca bir ad verilmiştir. Böylesi bir yapı aşağıdaki gibi örneklenebilir:

	No	(Personel) Adı	Alan Görevi
Kayıt

dBase III gibi bir bağıntılı veritabanı işletim sistemi, bağıntılar

(veritabanı kütükleri) arası ilişkileri kurmaya olanak sağlar. Aşağıdaki tabloyu düşünelim :

MO	Stok NO	(Malın) Adı	Fiyatı	(Stoktan Alan) Personel
...

Personel ve Stok Kütükleri arasındaki bağıntı, personelin adı **alanın** içeriğine göre kurulacaktır.

dBase III, daha önce 8 bitlik bilgisayarcılar için yazılmış dBase II'nin aksine 16 bitlik mikroişlemciler için hazırlanmıştır. IBM PC Ve Corona eşdeğeri mikrobilgisayarlarda ve en az 256Klık bellekle işletilebilir. Komutlar, özellikle tngUizce bilen kullanıcılar için oldukça anlaşılır ve kullanım kolaylığına sahiptir. Bütün komuttan tek tek işletmek

olası olduğu gibi, bir program içinde ardışık olarak işletmek ve veritabanı ile ilgili stok kontrolü, has-tane kayıtları, müşteri, abone işlemleri v.b. gibi her çeşit işi yapmak da olasıdır. Komutları içeren program dBase İÜ tarafından sağlanan kelime işlemciyle yazılabildiği gibi, Wordstar benzeri değişik kelime işlemciler kullanılarak da yazılabilir.

1.1. Teknik Özellikler

a- Her veritabanı kütüğü, bir milyona kadar kayıt içerebilir. Bellek açısından bakıldığında, her kütük iki milyon "byte" içerebilir. Ayrıca her kayıt 4000 bytelik yer kaplayabilir. Oluşturulabilecek alan sayısı ise 128 olarak sınırlanmıştır.

b- dBase IU'te değişik tipte alanlar tanımlanabilir. Bu alanlardan bazdan ve ilgili bellek sınırlaması (en fazla byte olarak) şöyle sıralanabilir :

C : Karakter : 254
D: Tarih : 8
L : Mantık : 1
M : Memo :4096
N : Nümerik : 19

c- Değişik tipte 15 kütük aynı anda açılabilir ve kullanılabilir. Veritabanı kütüğü (DBF) sayısı ise 10 ile sınırlanmıştır, bu sayı çeşitli uygulamalar için yetenlidir. Her aktif DBF için 7 index kütüğü ve 1 format kütüğü açılabilir. Bu kütük tiplerinden ileride sözedilecektir.

d- Nümerik doğruluk 15 ve ondalık olarak da 9 basamağa kadar sağlanmıştır. Kullanılabilecek en büyük sayı 1×10^{308} ve en küçük pozitif sayı da 1×10^{-307} 'dir.

e- Aktif bellek değişken sayı 256'ya kadar çıkabilmektedir. Bellek değişkenleri için toplam byte sayısı 6000 olarak belirlenmiştir.

Şunu da vurgulamak gerekir ki, bütün bu değerler bilgisayarın donanım konfigürasyonu ile sınırlanmış olabilir.

f- İşletim sistemi olarak MS-DOS, yada 2,0 sonrası PC-DOS kul-

lanılmaktadır. Eğer donanım 256K'dan daha geniş bellek içeriyorsa, dBase III içinden ASCII olarak kodlanmış başka programlar ve bu arada DOS komuttan da kullanılabilir. maktadır.

2. dBASE III YAPILARI

2.1. Kütükler

dBase III, bilgiyi 9 ayn biçimde formatlanmış kütüklerde saklar. Her kütük 8 karakterlik ad, bir nokta ve 3 karakterlik tipleyiciyle adlandırılır. dBase III, kullanılan komutlara göre kütükleri kendisi sınıflar, ancak istenirse noktadan sonraki Uç karakter değişik şekillerde belirlenebilir. Varolan kütük yapılan ve tipleri şöyledir :

a- Veritabanı (. dbf) Kütükleri

Bu tip kütükler, veriyi, teknik özellikleri giriş kısmında açıklanan kayıt (sıra) ve alanlar (dizeler) biçiminde depolar.

b- Memo (. dbt) Kütükleri

Bu tip kütükler, (. dbf) kütüklerine yardımcı yapılar olarak düşünülebilir. Her veritabanı kaydı 128 memo alanı içerebilir. Bütün bu alanlar aynı (. dbt) kütüğünde toplanır. Memo alanları karakterlerden oluşmakla birlikte, üst sınır 4000 karaktere kadar çıkabilmektedir. Her memo alanı kayıt başına 512 byte gerektirir, aynı bir kütükte depolandığı için (. dbf) kütüğünde yalnızca 10'luk bir yer işgal eder.

c- İndeks (. ndx) Kütükleri

İndeks kütükleri, varolan yapıyı fiziksel yerine mantıksal bir dizgede kullanabilmek için oldukça büyük kolaylık sağlar. Fiziksel dizge, kayıtların giriş sırasıdır. Mantıksal dizge ise, istenen bir ya da birkaç alan üzerinde (tanımlanmış bir anahtara göre yapılan) alfabetik, nümerik ya da kronolojik sıralamadır. Anahtar, bir ad ya da tarih alanı olabilir, öyle ki tamamen keyfi girilmiş bir abone listesi, varolan kütük yapısını değiştirilmeden, sözelimi hem alfabetik, hem de abone olma tarihine göre taranabilir.

Kayıt No	Ad	...
1	Hasan	...
2	Ahmet	...
3	Veli	...
4	Hüseyin	...

şeklinde fiziksel yapıya sahip bir kütük (. dbf) "AD"agöreindekslendiğinde, index (. ndx) kütüğü

Kayıt No	Ad	...
2	Ahmet	...
1	Hasan	...
4	Hüseyin	...
3	Veli	...

şeklinde oluşturulacaktır, özellikle uzun bir liste üzerinde işlem yaparken kullanılan indeksleme işleminin arama, v.b. işlemleri büyük ölçüde hızlandıracağı açıktır. Aynı anda 7 indeks kütüğü kullanılabilir, Veritabanı içeriği değiştikçe, açık indeks kütükleri de değişecektir.

d- Komut (. prg) Kütükleri

Bu tür kütükler, yapılması istenen işleri, yani bir dizi dBase III komutunu içeren programlardır. Bu kütükler, ASCII olarak saklanacağı için yalnızca dBase III içinde değil, birçok kelime işlemci kullanarak da yazılabilir.

e- Format (. fmt) Kütükleri

Veri girişlerinin ekranda hep aynı biçimde, belirdi bir forma göre yapılması için kullanılır. Format kütükleri, aynı şekilde yazıcıya da, çeşitli muhasebe, fatura, banka ve kart ile yürüyen stok işlemlerinde olduğu gibi, istenen formatla basılmasını sağlar.

f- Label(.lbl) Kütükleri

Bu kütüklerin içerdiği bilgiyi, LABEL komutu, labelleri basarken kullanacaktır.

g- Bellek (. mem) Kütükleri

256 bellek değişkenine kadar depolayabilir. Daha sonraki kullanımlar için değişkenlerin içeriklerini saklamakta kullanılır. Bir uygulamada, PASSWORD yaratmak ve doğrulamak için kullanılmıştır.

h- Rapor (. frm) Kütükleri

Rapor formları hazırlamakta ve bastırmakta kullanılır. Uygulama olarak, istenen formda stok listesi bastırmak düşünülebilir,

i-Metin (. txt) Kütükleri

Bu kütükler ASCII olarak saklanır ve yalnızca basılabilir karakterlerden oluşur. dBase III ile, satın alınmış diğer yazılımlar arası iletişim ve dBase III işlemlerinin kaydı için kullanılabilir.

2.2. Alanlar

Her alan, harfle başlayan 10 karakterlik bir isim, tip ve alan genişliği ile belirlenir. Bunların herbiri kütük yapısı oluşturulurken (CREATE), kullanıcı tarafından tanımlanacaktır.

dBase III, beş ayrı tipte alan kullanılabilir :

a- Karakter (c)

Klavyeden girilebilen herhangi bir karakter olabilir. Alan genişliği 254 ile sınırlanmıştır.

b-Tarih (D)

Amerikan formatına uygun olarak tasarlandığından, verinin ay/gün/yıl olarak girilmesi gerekir ve yatay çizgilerle ayrılmış toplam 8 karakterlik bir alan işgal eder. dBase III komutlarıyla tarih üzerinde gün ekleme, çıkarma, <v>. birçok işlem yapılabilir.

c-Nümerik(N)

Güvenilirlik limiti 15.9 formatta tam ya da ondalık sayılar için kullanılır.

d-Mantık (L)

Tek bir yer işgal eder, içeriği ya evet (doğru) ya da hayır (yanlış) dır.

e- Memo (M)

Metinler için kullanılır ve daha önce sözünü ettiğimiz (. dbt) kütüklerinde depolanır. 512 bytelık bloklardan oluşur ve bu sayı en fazla 4096'ya çıkabilir.

2.3. Bellek Değişkenleri

Veritabanı yapısı dışında, bağımsız olarak depolanırlar. Sözelimi bir muhasebe programında (iste-

nen koşula uyan) bir GİRDİ, ÇIKTI (burada girdi ve çıktı, alan adı olarak düşünülmelidir) toplamı bellek değişkeni olarak hesaplanıp, daha sonraki işlemlerde kullanılabilir. Değişkenler, karakter, tarih, nümerik ve mantıksal olmak üzere dört tipte sınıflandırılabilir. Aktif olarak 256 değişken, toplam 6000 byte ile sınırlanmıştır. Genellikle geçici depolama için kullanılmalarına karşın, (. mem) kütüklerinde, sonraki kullanımlar için kalıcı olarak da depolanabilirler.

dBase III, bu değişkenlerle ilgili çok çeşitli işlem kolaylıkları sağlayan fonksiyonlar içermektedir. Ayrıca, diğer programlama dillerinde olduğu gibi, değişkenler, isteğe göre global (PUBLIC) ya da lokal (PRIVATE) olarak da tanımlanabilir. Bu tanımlamalar, özellikle alt-programlar ve makro kullanımlar açısından önemlidir.

Değişkenler klavyeden istenilen formatta girilebilir ve program yapısı içinde girilen verilerin doğruluğu kontrol edilebilir.

3. İFADELER, OPERATÖRLER

Çeşitli programlama dillerinde olduğu gibi, dBase III içinde de alanlardan, bellek değişkenlerinden, sabitlerden, fonksiyonlardan ve operatörlerden oluşan değişik ifadeler kurulabilir. Şöyle bir örnek düşünelim, bir mal satışında müşteriye göre özel indirim yapıyor ve maun üzerine KDV eklenip toplam fiyat hesaplanıyor. Elimizde varolan veritabanı yapısında, malın fiyatı ve indirim alanları olsun. KDV % 10luk sabit bir oran ve TOPLAM da bellek değişkenimizin adı olduğunda, aşağıdaki ifadeyi kullanmamız gerekir.

Toplam = (Fiyat—Fiyat * İndirim) * 1.1

Eğer maun satışı birden fazla ise, genel toplam

Genel Top.=Top|am * Miktar

olarak hesaplanacaktır. (Klavyeden girilen büyük harflerde l'nın üzerinde

nokta görünmemesine karşın, örneğimizde, ve ğ, ü, ö, ç, ş gibi Türkçe karakterler de son bölümündeki örnek program içinde, izleme/okuma kolaylığı açısından kullanılmıştır).

Daha önce sözünü ettiğimiz format yapıları kullanılarak hesaplanan değerler, bir fatura üzerine de bastırılabilir.

Operatör olarak, yukarıdaki ifadelerde = , () , x ve — kullanılmıştır, öncelerine göre operatörler ** ya da (kuvvet), * ve / , + ve — , + (iki ya da daha fazla karakter dizisini tek bir dizi haline getirme) ve J5 (karakter dizisi karşılaştırma), < (küçük), > (büyük), =, O y a da # (eşit değil), < = , > = . . NOT. (değil), . AND . (ve), . OR . (veya) olarak sıralanabilir.

4. DİĞER ÖZELLİKLER

4.1. Çoklu Kütük Kullanımı

dBase III, aynı anda 10 veritabanı kütüğünün kullanılmasına olanak verir. Her kütük, açık olan tek kütük gibi bağımsız olarak kullanılabilir. Her kütük için ayrı bir çalışma alanı açılır ve l'den 10'a kadar olan çalışma alanları A'dan J'ye kadar harflerle belirlenir. dBase m'e girildiğinde çalışma alanı numarası l'dir. Çalışma alanları sırayla seçilmek zorunda olmadığı gibi, yalnızca A-J olarak değil, kütük adıyla da açılabilir. Bunun için, önce bir alan seçmek (SELECT), sonra da o alanda kullanılacak kütüğü belirlemek (USE) gereklidir.

İki farklı kütük, ortak bir alan ya da ifade tanelinde bağlanabilir. Bunun için, ikinci kütük seçilen ada göre indekslenmiş olmalıdır. Söz konusu özellik, iş uygulamalarında ve rapor hazırlanırken iki kütüğün birlikte kullanılmasını sağlar.

4.2. Diğer Programlarla

İlişki Kurma

Belirli komutlar, dBase III veritabanı kütüklerinden MailMerge gibi programlara veri transferi yapılabilmesini sağlar. Ancak, veri gönderile-

cek programın veri formatı bilinmemiştir. Sözgelimi MailMerge, virgülle ayrılmış alanlar içeren ayrı satırlara yerleştirilmiş kayıtlar kullanır. Bu şekilde, MailMerge tarafından kullanılabilir adres kayıtları dBase III içinde oluşturulabilir ve aynı formu taşıyan bir mektup örneğinin değişik adreslerle çoğaltılmasını sağlar.

Benzer bir şekilde, Multiplan gibi programlardan veri alınabilir. Bunun için söz konusu program, dBase III'ün anlayabileceği formda veri gönderebilmelidir.

Sözü edilen iki tip veri transferi için de detaylı bilgi kullanım kataloğundan elde edilebilir.

4.3. Fonksiyonlar

dBase İÜ, ay, gün, yıl ve tarih üzerine işlem yapma olanağı sağlayan fonksiyonlar içermektedir. Karakter manipülasyonu da çeşitli fonksiyonlarla olaa kılınmıştır. Diğer programlama dillerinde farklı olarak, dBase İÜ, yapısı gereği, matematiksel olarak anıru bir arşive sahiptir, çünkü bir veritabanı programında özel fonksiyonlara gerek duyulmayacaktır.

Bunların yanı sıra, dBase in, birtakım özel kütük test fonksiyonları da içermektedir. Bunlar, kütüğün olup olmadığını, başlangıç ve sonunu, kayıt numarasını test edip, Doğru/Yanlış şeklinde sonuç veren fonksiyonlardır. Ayrıca, karakter dizisinin uzunluğunu, ekran ve yazıcıda sıra ve dizik yerini belirleyen fonksiyonlar da vardır.

4.4. Sistem Konfigürasyonu

dBase m, standart ASCII metin kütüğü olan bir konöürasyon (CONFIG.DB) kütüğü kullanır. Bu kütük, herhangi bir editör programla ya da kelime işlemciyle yazılıp "SOFTKEY" olarak da adlandırılan fonksiyon tuşlarının, ON/OFF komutlarının kontrolü gibi olanaklar sağlar.

5. ÖRNEK

Komut ve fonksiyon listesi, bu

yazıda verilmemekle birlikte, bir stok kontrol programında altprogram olarak kullanılan stoktan malzeme arama menüsüne giriş olası olduğu ölçüde aşağıda açıklanmıştır. * ile başlayan satırlar dBase III tarafından açıklama olarak kabul edilip işleme konmaz.

```
* ARAMA PROGRAMI
* bu program tüm stok listesini tarayarak
* verilen özelliklere uygun malzeme bulur
*
* değişkenleri sil
clearall
* stok arama kütüğünü kullan
usearastok
* ana menüye dönme isteği gelene dek
*
* döngü içinde kal
do while . T .
* ikinci düzey altprogramlara gönderilecek değişkenleri belirle
public..., ...
* ekranı sil
dear
* sıra, dizik olmak üzere ekrana yaz
a 7,5 say 1 - Tanım kodlarına göre arama'
a 9,5 say '2 - Adına göre arama'
a 11,5 say '3 - Miktan en az stok düzeyinin altındakileri arama'
a 13,5 say '4 - Bütün stoğun listesi'
a 15,5 say '0 • Ana menüye dönüş'
* cursoru taşı
a 20,5
wait lütfen yukandakilerden birini seçiniz ' to ara
* seçime uygun işlemi yap
docase
  caseara = 1
  do arama 1
  caseara = 2
  do arama 2
  caseara = 3
  zap
  append from stok for uyan
  list off to print
  caseara = 4
  use stok
```

```
report form stok to print
caseara = 0
return
other wise
loop
endcase
enddo
```

Aramal ve arama2 ayrıca yazılması gereken iki alt programdır. UYARI, stok yapısında bulunan ve stok işlendikçe (giriş-çıkışa göre) değişen bir mantık değişkeni olup en az stok düzeyini kontrol etmektedir. Program, menüdekilerden farklı bir rakam girildiğinde döngünün başına dönecektir. Program işletilmeden önce, stok ve aynı yapıdaki arastok kütükleri oluşturulmuş (CREATE), ayrıca basılacak format hazırlanmış (MODIFY REPORT) olmalıdır. Program, dBase III kelime işlemcisi kullanılarak (MODIFY COMMAND) yazılabilir. Programı koşmak için ise, do<programadı> komutu yeterlidir.

Gerekli tüm programlar hazırlanıp, diskete sistem, COMMAND.COM kütüğü, dBase III sistem programları (.exe .ovr) yüklenip, bir "batch" kütüğü yazılırsa, bilgisayar açıldığı zaman stok kontrol programı koşmaya başlanacaktır. Ayrıca, disk sürücüler, ya da harddisk ile disk sürücüler arası geçişler de program içinde belirlenebilir, önerilen, hardiskli sistemde, program ve kütükler diskte, backup disketi de A sürücüsünde; çift disk sürücülü sistemde program A, veritabanı kütükleri de B sürücüsünde olacak şekilde çalışacaktır. Böylece kayıtlar aynı diskette tutulacak ve disketin bellek-yer sınırlaması sorun yaratmayacaktır.

önemle vurgulanması gereken bir nokta, sık ak backup (copy) alınması üzerinedir. Yapılar ve programlar değiştikçe veri korunması ve güvenilirlik açısından kütüklerin kopyalarının bir köşede korunmasının yararları büyük olacaktır.