

74.26
А 14

КЫРГЫЗ РЕСПУБЛИКАСЫНЫН БИЛИМ, ИЛИМ
ЖАНА МАДАНИЯТ МИНИСТРЛИГИ

ОШ ТЕХНОЛОГИЯЛЫК УНИВЕРСИТЕТИ
КЫЗЫЛ-КЫЯ ТЕХНОЛОГИЯ, ЭКОНОМИКА ЖАНА
УКУК ИНСТИТУТУ

ОШ МАМЛЕКЕТТИК УНИВЕРСИТЕТИ

АБЖАПАРОВА У.А., РАЕВ К.Т., РАЕВ А.К.

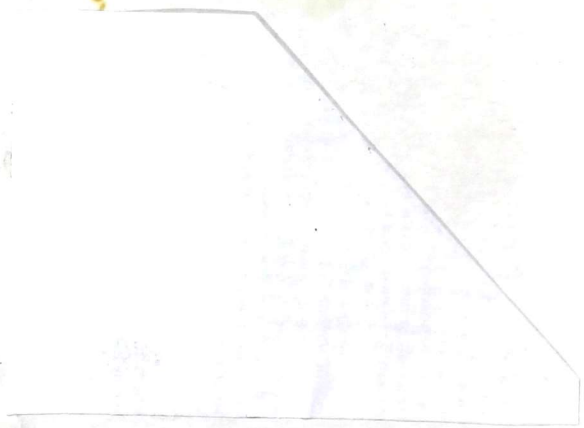


The graphic features a computer window titled 'FoxBase' with a globe on the left and a keyboard below. The main text is centered in the window.

FoxBase

III Plus
системасында
программалоо

Максам.
Александров.
Мажуга В. П.
25.05.2003*



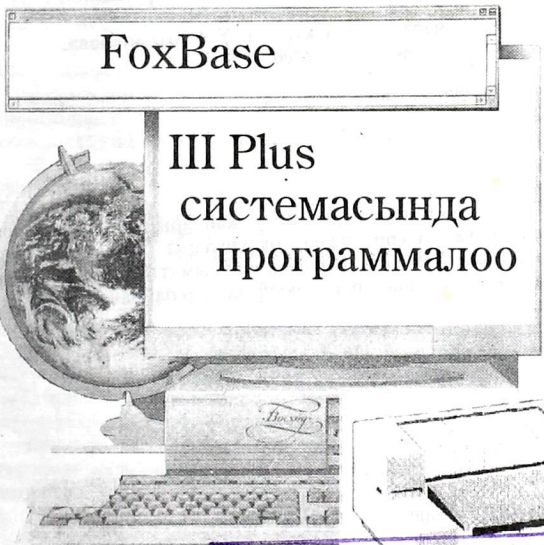
74.26
A 14

с/у/у 2/2

КЫРГЫЗ РЕСПУБЛИКАСЫНЫН БИЛИМ, ИЛИМ
ЖАНА
МАДАНИЯТ МИНИСТРЛИГИ

ОШ ТЕХНОЛОГИЯЛЫК УНИВЕРСИТЕТИ
КЫЗЫЛ -КЫЯ ТЕХНОЛОГИЯ, ЭКОНОМИКА ЖАНА УКУК
ИНСТИТУТУ

ОШ МАМЛЕКЕТТИК УНИВЕРСИТЕТИ
АБЖАПАРОВА У.А., РАЕВ К.Т., РАЕВ А.К.



7350

БИБЛИОТЕКА
 Ошского государственного
 университета
 Ош - 2000
 ЯНД № 932819

ББК 74.26

А—14

Ош мамлекеттик Университетинин Кызыл-кыядагы технологиялык, экономика жана укук институнун илимий кеңешинин чечими боюнча басмадан чыкты

Рецензент: А.О. Кыбыраев Физика –математика илимдеринин кандидаты, доцент.

Абжапарова Ү.А., Раев К.Т., Раев А.К.

А—14 Foxbase III Plus системасында программалоо:

(Окуу – методикалык курал) / Ү.А.Абжапарова, К.Т.Раев, А.К.Раев — Ош : 2000 — 70 б.

ISBN 9967 — 9967 — 0 — 5

Окуу куралын жогорку окуу жайларынын студенттери, мектеп мугалимдери жана окуучулары үчүн жазылды. Ошондой эле бул окуу куралын маалыматтар базасы менен иштөөгө кызыккан көпчүлүк окуурмандар пайдаланса болот.

А 4306010000 — 2000

ББК 74.26

ISBN 9967 — 9967 — 0 — 5

© Абжапарова Ү.А.,
Раев К.Т., Раев А.К.

Кириш сөз

Азыркы мезгилде маалыматтар базасы менен иштөө учурдун негизги талаптарынын бири. Келечекте ар бир инсан информатиканын бул тармагы менен тааныш болусу керек. Ошондуктан, бул окуу-усулдук колдонмонуң максаты маалыматтар базасы менен иштөөнү үйрөнүүгө максат койгон окурмандарга арналды.

Окуу-усулдук куралдын I главасында маалыматтар базасын түзүү боюнча кеңири маалымат берилген. Бул главада маалыматтар базасынын структурасын ачуу, аны толтуруу, киргизилген маалыматтарды окуу, сорттоо, жоюу ж.б. маселелери каралган.

II главада маалыматтарды иштеп чыгуу үчүн керек болгон командаларга токтолдук. Ошондой эле ар бир команда түшүнүктүү болушу үчүн бир нече мисалдарды карап өттүк.

III главада Fox Base системасында сызыктуу, тармактануучу, циклдик программаларды түзүүгө муктаж болгон командалар каралган.

Биздин максатыбыз баардык окурмандарга маалыматтар базасы менен иштөөгө багыт берүү болуп саналат. Демек, өз оюңузда болсо, албетте, окуу-усулдук куралды жакшыртуу максатында, анда ОшМУнун информатика жана колдонмо математика, Ош технологиялык университетинин Кызыл-Кыядагы технология, экономика жана укук институтунун математикалык методдор жана программалык камсыздоо, ОшТУнун экономикадагы математикалык методдор кафедраларына жөнөтсөңүздөр болот.

Бул эмгек биздин көп жылдык тажрыйбабыздын негизинде жазылды. Сиздердин оюңузда кутөбүз.

Авторлор

Маалыматтар базасын түзүү

§ 1. Негизги түшүнүктөр

FoxBase III Plus системасы маалыматтар базасы менен ар түрдүү амалдарды аткаруу үчүн өркүндөтүлгөн каражаттарга ээ. Бул системада маалыматтар базасы деп, тик бурчтуу таблицкага окшош структураны түшүнөбүз. Таблицанын ар бир ячейкасы өзү жайгашкан жолчо жана мамыча менен аныкталат. Мындай типтеги маалыматтар базасын релациялык деп атайбыз.

FoxBase III Plus системасында таблицалар жолчолор боюнча сакталат. Ар бир жолчо жазуу деп аталат. Таблицанын мамычаларына карата жазуу талааларга бөлүнөт жана жазуунун ар бир талаасы таблицанын мамычасына дал келет. Ар түрдүү жазуунун жайгашышы жана бир мамычадагы талаанын узундугу бирдей. Мындай таблица .dbf кеңейтилиштүү өз алдынча файлга жайгаштырылат жана мындан ары .dbf файлы деп атайлы. .dbf файлдагы жазуу беш типтеги талааларды камтый алат:

- Календардык дата D (Data);
- символдук C (Character);
- сандык N (Number);
- логикалык L (Logical);
- эскертүү M (Memo).

Символдук жана сандык талаалардын узундуктарын колдонуучу өзү аныктайт. Сандык талаа үчүн ондук бөлүнүшүнүн узундугун көрсөтүү керек. Калган типтеги талаалар стандарттык узундукка ээ. Календардык дата үчүн – 8, логикалык үчүн –1, эскертүү үчүн –10, мемо- 4906 байт талаалардагы эскертүү текисти .dbf кеңейтилишине ээ болгон файлда сакталат. Символдук талаанын максималдык узундугу 254 символ, сандык талаанын максималдык узундугу 18 позиция.

Маалыматтар базасындагы жазуу киргизилген тартиптеги удаалаштыкты карап көрүүгө болот. Бул дайыма эле ыңгайлуу эмес. Ошондуктан, система жазуулардын тартибин өзгөртүп карап көрүү үчүн 3 методду сунуш кылат.

Биринчи ыкма – индекс түзүү, б.а. аныкталган формула боюнча ар бир жазууга жол көрсөтүүчү түзүп, андан кийин жазууларды жолчонун маанилеринин өсүү же кемүү тартибинде жайгаштыруу. Жазуунун жолчолорунун тизмесин индекс, ал эми индекс сакталган файлды, индексстүү файл деп атайбыз.

Индекс түзүү маалыматтар базасын өзгөртпөйт, бирок аны жаңы тартипте карап көрүүгө мүнкүндүк берет. Мындай индекстердин бир нечесин түзүүгө болот жана индекстерди алмаштыруу менен маалыматтар базасын карап чыгуу тартибин алмаштырууга болот. Ар бир индекс өз алдынча файлда сакталат. Ал файл .idx кеңейтилишине ээ болот. .Idx – файлды .dif – файлынан көз карандысыз пайдаланууга болбойт.

Экинчи ыкма – фильтр түзүү, б.а. жазууларды тандап алуу шарттары. Тандап алуу шарттарын канааттандырган жазуулар автоматтык түрдө өткүрүп жиберилет, б.а. көрүнбөгөн сыяктуу болуп калат.

Үчүнчү ыкма – сактоо. Бул ыкма .dbf-файлын тандалган талаа боюнча өзгөртүп түзөт.

§ 2. Маалыматтар базасын түзүү

Бизге студенттер жөнүндө маалыматтар берилсин: фамилиясы, аты, атасынын аты жана математика, физика, информатика, кыргыз тил предметтеринен жарым жылдык баалары берилсин. Бул маселеге маалыматтар базасын түзөлү.

Маалыматтар базасынын жаңы файлын түзүү жана структурасын баяндоо

.CREATE <файлдын аты>

командасы менен аткарылат.

Жазуу үчүн талаанын аттарын, төмөнкүдөй аныктайлы:

Маалыматтар талаанын аттары

Группасы	GR
Фамилиясы	FM
Аты	AT
Атасынын аты	AA
Туулган жылы	TJ
Жынысы	J
Математика	MAT
Физика	FIZ
Информатика	INF
Кыргыз тили	KT

Жогорудагы маселе үчүн маалыматтар базасынын файлынын атын OM деп коелу.

.CREATE

Enter the name of the new file: OM
(Жаңы файдын атын киргиз☺

Командасын аткарабыз. Бул команда түзүлүүчү файлдын жазуусунун талааларынын аттарын, типтерин жана узундуктарын интерактивдүү режимде киргизүүгө мүмкүндүк берет, төмөндөгүдөй форма пайда болот.

OM.DBF		Bytes remaining : 4000	
Files defuned : 0			
Field name	type	width	dec
1	chfrYtext		
Names start with a letter :			
The remainder may de letters, digits,			
Or underscore			

Рамканын ичинде .dbf-файлынын структурасын киргизүү үчүн маалымат берүүчү информация берилген. Мисалы, бир символду четтетүү үчүн Del клавишасын же талааны коюу

үчүн бир учурда Ctrl+End (^ End) клавишаларын басуу керек.

Create командасынан чыгуу үчүн Ctrl+End (^ End) клавишаларын басабыз структура киргизгенден кийин .dbf-файлын толтуруу зарылчылыгы жөнүндө суроо берилет. Эгерде ? жооп берсек, б.а. Y (ооба) клавишасын бассак, анда .dbf-файлын маалыматтар менен толтуруу башталат. Экранда төмөндөгүдөй форма пайда болот:

Record NO	1
GR	
FM	
AT	
AA	
TJ	
J	
FIZ	
MAT	
INF	
KT	

Мында көрүнүп тургандай, экранда талаанын ар бир киргизилүүчү маанисине аныкталган санга позиция бөлүнүп берилген. Ал эми рамканын ичинде сурап билүүчү информация жайгашкан. Эгерде жок деп жооп берсек, анда экранда чакыруу чекити пайда болот.

Структурасы түзүлгөн файл менен иштөө үчүн, ал файл ачык болуш керек.

Файлды ачуу

.USE <файлдын аты >

командасы менен аткарылат. Жаңы файлды ачуу менен, бул команда мурда ачылган файлды жабат.

Ошондой эле .dbf-файлдарын жабуу үчүн

CLOSE DATABASES

атайын командасын колдонсок да болот.

.Dbf-файлынын структурасын баяндоодон мурда түзүлгөн файлдын структурасын да пайдаланууга болот. Ал үчүн COPY STRUCTURE командасын колдонобуз.

Команданын форматы төмөнкүдөй:

.COPY STRUCTURE TO <файлдын жаңы аты >

Мисалы, башка группанын студенттери үчүн жогорудагыдай эле маалыматтар берилсин. Ал .dbf- файлынын аты OMM деп атайлы.

Ал үчүн структурасы пайдаланылуучу .dbf-файлы ачык болуш керек.

Демек, OMM .dbf-файлынын структурасын түзүүдө төмөнкүдөй командаларды пайдаланып аткарабыз.

. USE OM

. COPY STRUCTURE TO OMM

Түзүлгөн .dbf-файлынын структурасынын тууралыгына ишенүү үчүн ал файлды ачып, андан кийин

. DISPLAY STRUCTURE

же LIST STRU командаларынын бирин аткаруу керек. Натыйжада экранда төмөндөгүдөй форма пайда болот.

```
. USE OM
. DISPLAY STRUCTURE
Structure for database : A: OM . dbf
Number of data records : 0
Data of list update : 03Y17Y00
Field Field name type width dec
1 GR C 8
2 FM C 12
3 AT C 12
4 AA C 12
5 TJ D 8
6 J C 1
7 FIZ N 1
8 MAT N 1
9 INF N 1
10 KT N 1
** Total** 57
```

COPY STRUCTURE командасын эски түзүлгөн .dbf-файлынын кээ бир талааларын пайдалануу менен жаңы .dbf-файлынын структурасын түзүү үчүн колдонууга болот. Ал үчүн команда FIELDS опциясы менен толукталып, опциядан кийин жаңы .dbf-файлынын структурасын баяндоого керек болгон талаалардын тизмеси камтылат.

Эскертүү: мындан ары командалардын форматтары берилгенден кийин, аларды кайрадан жазууда биринчи төрт тамгасын гана жазуу жетиштүү.

Команданын форматы төмөндөгүдөй:

```
COPY SRTU TO <файлдын жаңы аты > FIELDS  
<талаалардын аттары>
```

```
.USE OM
```

```
.COPY STRU TO OMM FIELDS FM,AT,AA,TJ
```

OMM .dbf-файлынын структурасы баяндала башташына ынануу үчүн

```
.USE OMM
```

```
.LIST STRU
```

командаларын берүү жетиштүү.

Эгерде LIST STRU же DISP STRU командаларына TO PRIN параметрин кошуп жазсак, анда түзүлгөн структураны принтерден чыгарууну камсыз кылат.

Бул .dbf-файлынын чын түзүлгөндүгүнө жана катуу магниттик «винчестер» тибиндеги дискага жазылышына ишенүү үчүн

```
.DIR
```

командасын аткарсак болот. Анда экранда файлдардын аттарынын тизмеси чыгат б.а. OM, OMM, ANK файлдары чыгарылат.

§ 3. Маалыматтар .dbf-файлынын алгачкы ирет толтуруу

CREATE командасынан кийин аткарылышы мүмкүн экендигин көрдүк. Мындан башка дагы толук экрандуу, интерактивдүү командалар бар. Маалыматтар базасын жаңы жазуулар менен толуктоо

. APPEND

командасы менен аткарылат. Ал үчүн толукталуучу маалыматтар базасы алдын-ала ачылган болуш керек.

Толукталуучу жазуу дайыма маалыматтар базасынын файлынын аягына жазылат. Андан кийин OMM файлын алалы:

. USE OMM

. APPEND

командалары боюнча экранда талаалардын жазылыштарынын маанилерин киргизүү үчүн төмөндөгүдөй формасы пайда болот (1-сүрөт).

Record NO	1
GR	Мое-1-99
FM	Адьялов
AT	Дамир
AA	Муктаралиевич
TJ	17/07/81
J	Э
FIZ	5
MAT	5
INF	4
KT	5

1-сүрөт.

Пайда болгон тик бурчтука маалыматтарды киргиздик. APPEND режиминен чыгыш үчүн Ctrl+ End клавишаларын бир убакытта басуу керек.

§ 4 Маалыматтарды окуу

Ачылган маалыматтардын базасын экранга чыгарыш үчүн LIST (тизме) командасынын жардамы менен көрүүгө болот. Маселен OMM маалыматтардын базасын экранга чыгарыш үчүн төмөнкү командаларды аткаруу керек:

```
.USE OM  
.LIST
```

Экранда төмөнкүдөй информация пайда болот:

Мисалы:

Record #	GR	FM	AT	AA	TJ	J	FIZ	MAT	INF
KT									
1	МОЭ-1-99	Аделов	Дамир	Муктаралиевич	17.07.81	Э	5	5	4 5
2	МОЭ-1-99	Исраилов	Улан	Бурканович	08.04.82	Э	4	5	5 5
3	МОЭ-1-99	Айдаров	Талант	Ураимович	08.04.81	Э	4	4	4 4
4	МОЭ-1-99	Бакиров	Алмаз	Кадырович	12.05.82	Э	3	5	5 5
5	МОЭ-1-99	Рысбаев	Кубаныч	Абсатарович	02.02.80	Э	4	4	3 4
6	МОЭ-1-99	Сартбаева	Махарам	Сагынбаевна	16.11.82	К	5	5	5 5

2-сүрөт.

COPY STRU командасы боюнча түзүлгөн маалыматтар базасы бош болот. Ал .dbf-файлына жазууларды киргизүү APPEND командасы, же структурасы окшош талааларды камтыган файлдагы жазуулар менен толукталат. Экинчи учурда информацияны киргизүү үчүн APPEND FROM <файлдын аты> командасы, колдонулат. Мисалы, OM файлынан OMM файлына маалыматтарды толуктоо төмөндөгүдөй болот.

```
.USE OM  
.COPY STRU TO OMM  
.APPE FROM OM
```

COPY командасы маалыматтар базасынын структурасы гана эмес, андагы жазууларды да бир учурда көчүрө алат. Мисалы, OM файлынан OMM файлына бир учурда структураны жана жазууларды көчүрүү үчүн төмөндөгүдөй командаларды аткарабыз:

. USE OMM
. APPE FROM OM

Record #	GR	FM	AT	AA	TJ	J	FIZ	MAT	INF	KT
1	MOЭ-1-99	Аделов	Дамир	Муктаралиевич	17.07.81	Э	5	5	4	5
2	MOЭ-1-99	Исраилов	Улан	Бурканович	08.04.82	Э	4	5	5	5
3	MOЭ-1-99	Айдаров	Талант	Ураинович	08.04.81	Э	4	4	4	4
4	MOЭ-1-99	Бакиров	Алмаз	Кадырович	12.05.82	Э	3	5	5	5
5	MOЭ-1-99	Рысбаев	Кубаныч	Абсатарович	02.02.80	Э	4	4	3	4
6	MOЭ-1-99	Сартбаева	Махарам	Сагынбаевна	16.11.82	K	5	5	5	5

. COPY TO OMM FIEL FM,AT,AA,TJ,KR

Алдын-ала ачылган маалыматтар базасындагы жазууларга өзгөртүүлөрдү киргизүү же толуктоо EDIT командасынын жардамы менен да аткарылат. Мисалы, OMM маалыматтар базасынын 1-жазуусуна өзгөртүүлөрдү киргизүү керек болсун. Анда

. USE OMM
. EDIT 1

командаларынын жардамы менен экранга 1-жазуу жөнүндөгү толук маалымат чыгарылат.

Record no 1

! Cursor <->	! UP DOWN	! DEIETE	! I nsert mode: Ins
! Char <->0	! Field ^	! Char:Del	! Exit: ^End
! Word: Hone end	! Page:PgUp PgDn	! Field: ^Y	! Abort: ^Esc
!	!	! Record: ^U	! Text: ^Home

GR	МОЭ-1-99
FM	Аделов
AT	Дамир
AA	Муктаралиевич
TJ	17Y07Y81
J	Э
MAT	5
FIZ	5
INF	4
KT	5

3-сүрөт.

Редакциялоочу жазуунун талааларында информацияларды өзгөртүү APPEND командасы сыяктуу эле аткарылат. ←, ↑, ↓, →, PgDn, PgUp, DEL

Эң акыркы жазууну карагандан кийин PgDn же Return клавишаларынын бирин бассак, анда экранда маалыматтар базасында жаңы жазуунун талааларын толтуруу үчүн форма пайда болот. Ошентип EDIT командасы менен да каалаган сандагы жаңы жазууларды киргизе алабыз. Командадан чыгуу Ctrl +End клавишасынын жардамы менен ишке ашат.

Редакциялоо сеансынын мезгилинде маалыматтардын базасынын жазылыш талааларына кандайдыр бир жаңы маани киргизүү керек болсо, анда CHANGE (өзгөрт) командасы колдонулат. Маселен, жогорку окуу бөлүмүнө, студенттердин математика боюнча жылдык баасы келип түшсүн. Мына ушуга байланыштуу OMM маалыматтардын базасын өзгөртүүгө туура келет. Ал үчүн төмөнкү командаларды киргизүү талап кылынат:

. USE OMM

.CHANGE FIELDS MAT

экранда төмөнкүдөй форма пайда болот:

RECORD NO	1
MAT	5

Бул жазылыштагы MAT талаасына жаңы маанини берүүгө мүмкүнчүлүк берет. Курсор турган жерге жаңы бааны бергенден кийин тизме боюнча кийинки жазылыштарга курсор өтө берет.

Бардык студенттердин баалары киргизилмейинче бул процесс улантыла берет. Бул команданын ыңгайсыз жери экранга студенттердин фамилиясы чыкпайт. Мына ошондуктан студенттин жаңы баасын төмөнкү команда менен берүүгө болот:

.CHANGE FIELDS FM, MAT

.CHANGE командасына <чектегич> опциясын да кошууга болот:

.CHANGE NEXT 3 FIELDS FM, MAT

Командасы 2,3,4- жазылыштарга өзгөртүү кийрүү мүмкүнчүлүгүн түзөт.

Маалыматтарды экрандан бардык талаалары боюнча редакциялоо үчүн BROWSE командасын пайдалануу өтө ыңгайлуу (BROWSE-тартиби жок окуу).

.BROWSE

командасы боюнча учурдагы жазылыштан баштап удаалаш жазылган жазылыштардын группасын редакциялоого жана карап чыгууга болот:

.USE OMM

.BROWSE

RECORD NO ON

Rec. # GR FM AT AA TJ J FIZ MAT INF

1	MOЭ-1-99	Аделов	Дамир	Муктаралиевич	17.07.81	Э	5	5	4
2	MOЭ-1-99	Исраилов	Улан	Бурканович	08.04.82	Э	4	5	5
3	MOЭ-1-99	Айдаров	Талант	Ураимович	08.03.81	Э	4	4	4
4	MOЭ-1-99	Бакиров	Алмаз	Кадырович	12.05.82	Э	3	5	5
5	MOЭ-1-99	Рысбаев	Кубаныч	Абсатарович	02.02.80	Э	3	4	4
6	MOЭ-1-99	Сартбаева	Махарам	Сагынбаевна	16.11.82	К	5	5	5

4- сүрөт

Эгер маалыматтардын базасынын жазылышы 30 символдон ашып кетсе, анда жазылыш экранга толугу менен батпайт. Акыркы талаа экрандын оң жагында калып калат. Аны көрүш үчүн Ctrl, → клавишаларын бирге басуу керек. Бул учурда бардык жазылыштар солго бир талаага жылышат. Анда көрүнбөгөн 1-талаа экрандын оң жагында пайда болот. Ctrl, → клавишаларын кайталап басканда, сүрөттөлүштү дагы бир талаага солго жылдырат. Мына ошону менен бирге биринчи талаа экрандын сол жагына чыгап кетет да көрүнбөй калат. Сүрөттөлүштү кайра көрүш үчүн Ctrl, ← клавишаларын басуу керек (кайталап басуу менен сол жактагы көрүнбөгөн талааларды көрүүгө болот). Ирети менен кийинки жазылыштардын группасын көрүш үчүн PgDn (бир бетке төмөн) клавишасын басуу керек. Мурунку жазылыштардын группасына кайрылыш үчүн PgUp клавишасын басуу керек.

BROWSE командасы менен кошо FIELDS опциясын да колдонууга болот. Бул болсо редакциялануучу талаалардын тизмесин көрсөтүүгө мүмкүнчүлүк берет.

. BROWSE FIELDS GR, FM, AT, AA, MAT

Командасы боюнча OMM ачылган маалыматтардын базасынан экранга «Фамилия», «Аты», «Атасынын аты» жана

«Математика» деген талаа чыгат. Жазылыштарды кошууга да болот. Ал үчүн курсорду акыркы жазылыштын каалаган позициясына коюп, Enter клавишасын басабыз. Экрандын жогорку жолчосунда төмөнкүдөй талаа коюлат:

Add new record ? (Y/N):

(Жаңы жазылышты кошосузбу? (Ооба/Жок)):

« Y » ти алсак, маалыматтардын базасынын акыркы жазылышынан кийин бош болгон жазылыш пайда болот. Аны жогоруда баяндаган редакциялоо командасынын жардамы менен толтурууга болот. BROWSE режиминен чыгыш үчүн Ctrl+End клавишаларын басуу керек. ESC клавишасын бассак редакцияланган өзгөртүүлөрдүн бардыгы жок болот. Маалыматтардын базасы BROWSE командасы берилгенге чейинки абалга келет.

§ 5 Жазылыштарды киргизүү жана чыгаруу

Маалыматтардын базасынын акырына жаңы жазылыштарды кошуу APPEND, EDIT, BROWSE режимдеринде аткарууга болот. Эгер жазылыштардын арасына жаңы жазылышты киргизүү керек болсо, анда INSERT(коюу) командасы колдонулат. Ал төмөнкү көрүнүштө жазылат:

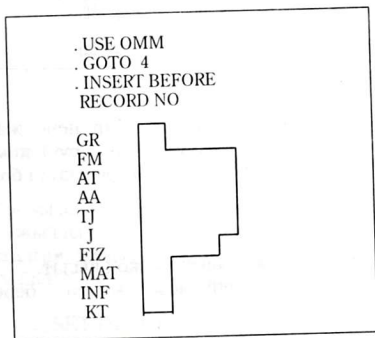
.INSERT [BLANK] [BEFORE]

Эгер BEFORE (алдына) опциясы командада көрсөтүлсө, анда жазылыш учурдагы жазылыштын алдына коюлат. Эгер BEFORE опциясы жазылбаса, анда жазылыш учурдагы жазылыштан кийин коюлат. Эгер BLANK (бош) опциясын командага жазсак, анда INSERT командасы боюнча бош жазылыш кийрилет да EDIT же BROWSE командасынын жардамы менен ал жазылыш толтурулат. Эгер BLANK опциясы командада жазылбаса, анда INSERT командасы боюнча экранда форма пайда болот. Андан кийин ал форманы талаалардын маанилери менен толтуруу керек. Форманы толтуруш үчүн EDIT режиминдеги башкаруучу клавишалар эле колдонулат.

1-мисал.

```
. USE OMM  
. GOTO 4  
. INSERT BEFORE
```

Натыйжада экранда 4- жазылыш үчүн форма пайда болот (1-сүрөт).

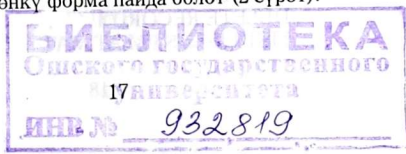


1-сүрөт.

2- мисал

```
. USE OM  
. LIST FIELDS GR,FM,AT,TJ  
. GOTO 4  
. INSERT BLANK  
. GOTO TOP  
. LIST FIELDS GR,FM,AT,TJ
```

Бул мисалда INSERT командасы 4- жазылыштан кийин бош жазылыш кийрийт. Жогорку программа аткарылгандан кийин экранда төмөнкү форма пайда болот (2-сүрөт).



.USE OMM

.GOTO 4

.INSERT BLANK

.GOTO TOP

.LIST FIELDS GR,FM,AT,TJ

Record #	GR	FM	AT	TJ
1	МОЭ-1-99	Аделов	Дамир	17.07.81
2	МОЭ-1-99	Исраилов	Улан	8.04.82
3	МОЭ-1-99	Айдаров	Талант	08.03.81
4	МОЭ-1-99	Бакиров	Алмаз	12.05.82
5				
6	МОЭ-1-99	Рысбаев	Кубаныч	02.02.80

2-сүрөт.

Маалыматтардын базасын бир же бир нече жазылыштарды чыгарып ташташ үчүн DELETE (чыгаруу) командасы колдонулат. DELETE командасы жөнөкөй форматта болсун:

.DELETE

Бул команда учурдагы жазылышты жок кылат.

DELETE <чектегич> опциясы менен берилгенде төмөнкүдөй болот:

..DELETE RECORD N

Мында бул команда N- номерлүү жазылышты жок кылат:

.DELETE ALL

Командасы маалыматтардын базасын бардык жазылыштарын жок кылат. DELETE командасы дискадан жазылыштарды физикалык жактан өчүрбөйт. Ал болгону DELETE командасы тиешелүү болгон жазылыштарга гана белги коюп коет:

.USE OMM

.DELETE

.DELETE RECORD 4

.LIST FIELDS FM,AT,MAT

Командаларын аткаргандан кийин 3-сүрөттөгү экранда пайда болот:

```
.USE OMM
.DELETE
  1 rekord deleted
.DELETE RECORD 4
  1 rekord deleted
.LIST FIELDS FM,AT,MAT
Record #  FM      AT      MAT
  1  *Аделов  Дамир   5
  2  Исраилов Улан   4
  3  Айдаров  Талант  4
  4  *Бакиров Алмаз  3
  5
  6  Рысбаев  Кубаныч 4
```

3-сүрөт.

1-жана 4- жазылыштар чыгарылып ташталганы менен печатта калып, «жылдызча» (*) белги менен белгиленет.

Ташталган жазылыштарды экрандан көрүнбөй тургандай кылууга болот. Ал үчүн төмөнкү команданы аткаруу керек:

```
. SET DELETE ON
```

Бул командадан кийин

```
. LIST FIELDS FM,AT,MAT
```

Командасын берсек, анда (*) белгиси коюлган жазылыштар экранда көрүнбөй калат (4-сүрөт).

```
.SET DELETE ON
.LIST FIELDS FM,AT,MAT
rekord #  FM      AT      MAT
  2  Исраилов Улан   4
  3  Айдаров  Талант  4
  5
  6  Рысбаев  Кубаныч 4
```

4-сүрөт.

Экрандан чыгарылып ташталган жазылыштарды кайра экранга чыгарыш үчүн төмөнкү команданы аткаруу жетиштүү:

. SET DELETE OFF

Чыгарылып ташталган катары белгиленген жазылыштарды RECALL (калыбына келтирүү) командасы менен калыбына келтирүүгө болот.

. RECALL

командасы учурдагы жазылышты,

. RECALL RECORD 4

командасы 4- жазылышты,

. RECALL NEXT 3

Командасы учурдагы жазылыштан баштап бардык чыгарылып ташталгандардын үчөөсүн,

. RECALL ALL

командасы мурда чыгарылып ташталган жазылыштардын бардыгын калыбына келтирет. RECALL командасынын кылган кызматы 5-сүрөттө көрсөтүлгөн:

.SET DELETE OFF

.LIST FIELDS FM,FIZ,MAT

record #	FM	FIZ	MAT
1.	*Аделев	5	5
2.	Исраилов	4	5
3.	Айдаров	4	4
4.	*Бакиров	3	5
5.		0	0
6.	Рысбаев	4	4

.RECALL RECORD 4

1 record recalled

.LIST FIELDS FM,FIZ,MAT

Record#	FM	FIZ	MAT
1.	*Аделев	5	5
2.	Исраилов	4	5
3.	Айдаров	4	4
4.	Бакиров	3	5
5.		0	0
6.	Рысбаев	4	4

5-сүрөт.

Акырында жазылыштарды дискадан өчүрүш үчүн (чыгарылып ташталган катары белги коюлган жазылыштар) PACK (упаковкалоо) командасы колдонулат:

.PACK

командасын ОММ маалыматтардын базасын колдонгондо (5-сүрөттөгү абалында колдонгондо) төмөнкү информация экранга чыгат (6-сүрөт).

6-сүрөт.

5, 6-сүрөттөрдү салыштырып, 5- сүрөттөгү * коюлган жазылышты 6-сүрөттү жок экендиги байкалат. 6-сүрөттөгү жазылыштар кайрадан номерленгени көрүнүп турат.

Маалыматтардын базасындагы бардык жазылыштарды тазалаш үчүн төмөндөгү командаларды аткаруу керек:

```
.DELETE ALL
.PACK
```

Жогорку командалардын кызматын төмөндөгү команданы аткаруу керек:

```
.ZAP
```

Эми дискадан маалыматтардын базалык файлын өчүрүш үчүн төмөнкү команданы аткаруу керек:

```
.ERASE KARTUCH.DBF
```

өчүрүлө турган файл жабылуу бойдон болуш керек. Эгер жок кылынуучу файл ачык болсо, анда аны алдын ала жабуу керек. Ал үчүн төмөнкү команданы аткаруу керек:

Бул команда боюнча мурда ачылган бардык маалыматтардын базасы жабылат.

§ 6 Структураны модификациялоо

Структураны модификациялоо зарылчылыгы маалыматтарды редакциялоо мезгилинде келип чыгат. Себеби тигил же бул талаанын маанисин жазгыла, структураны баяндаганда жетишээрлик позициялар калтырылган болот. Маалыматтарды иштетип жаткан мезгилде типтерди алмаштырууга туура келиши мүмкүн. Маселен, «символдук» типти «сандык» типке алмаштыруу туура келет. Маалыматтардын структурасын өзгөртүү үчүн MODIFY STRUCTURE (структураны модификациялоо) командасын берүү керек: Бул команданын жардамы менен төмөнкүдөй өзгөртүп түзүүлөрдү жүргүзө алабыз:

- талаалардын аттарын өзгөртүү;
- талаалардын узундуктарын өзгөртүү;
- талаалардын типтерин өзгөртүү;
- жаңы талааны киргизүү жана керексиз талааларды жок кылуу.

Модификациялануучу маалыматтардын базасы алдын ала ачылган болушу керек. Маселен, OM маалыматтардын базасынын структурасын модификациялоо үчүн төмөнкү командаларды киргизүү зарыл:

```
USE OM  
MODIFY STRUCTURE
```

Натыйжада экранда төмөнкүдөй форма пайда болот (1-сүрөт):

	Имя поля	тип	длина	дес
1.				
2.	FM	СимвУтекст	12	
3.	AT	СимвУтекст	12	
4.	AA	СимвУтекст	12	
5.	TJ	Дата	8	
6.	J	СимвУтекст	1	
7.	FIZ	Число	1	
8.	MAT	Число	1	0
9.	INF	Число	1	0
10.	KT	Число	1	0

Имена начинается с буквы;

Остальные символы могут быть буквами, цифрами или знаком

Подчеркивание

1-сүрөт.

Эскертүү. 1- сүрөттөгү информация ЕС-1040, ЕС- 1041 компьютерлеринде пайда болот.

Бул структура бош эмес экендиги көрүнүп турат. Ал мурун CREATE командасы боюнча колдонулган маалымат тардын базасынын структурасы болуп эсептелет. Каалаган параметрди өзгөртүш үчүн ←,↑,↓,→ клавишаларды пайдаланса болот.

Курсорду оңдоло турган параметрге алып келип, аны кайрадан киргизүү керек. Структураны модификациялоону бүтүш үчүн Ctrl+End клавишаларын бирге басуу керек. Өзгөрүлгөн структурадан баш тартыш үчүн ESC клавишасын басуу жетиштүү. Модификацияланган структураны көрүү үчүн төмөнкү командалардын бирин аткаруу жетиштүү:

- . DISPLAY STRUCTURE же
- . LIST STRUCTURE

Тандалуу операцияларын жана информацияларды иштетүүдө берилген структурага окшош болгон жаңы маалыматтардын базасын түзүү талап кылынат. Мына ушундай учурда CREATE командасынын жардамы менен жаңы структураны түзүштүн кереги жок. Каалаган

структуранын баяндалышын толук же жарым-жартылай жаңы маалыматтардын базасына копиялоого болот. Ал COPY STRUCTURE (структураны копиялоо) командасы менен ишке ашат.

Жаңы маалыматтардын базасын түзүш үчүн (маселен BD2 аты менен) эстеги маалыматтардын базаларынын бирине окшош болгон (маселен BD1) структураны ачуу керек:

. USE BD1

Андан кийин

. COPY STRUCTURE TO BD2

командасын аткаруу керек. Жаңы түзүлгөн маалыматтардын базалык структурасын экрандан карап чыгуу үчүн ал структураны ачып, андан кийин DISPLAY STRUCTURE командасын аткаруу керек:

.USE BD2

.DISPLAY STRUCTURE

Жаңы түзүлгөн маалыматтардын базасынын аты буга чейин болгон маалыматтардын базаларынын биринин аты менен да аталышы мүмкүн (Маселен, BASADAN). Бул учурда экранга төмөнкүдөй өтүнүч-билдирүү пайда болот:

BASADAN.dbf already exists, overurite if ? (Y/N):

BASADAN.dbf бар аны кайра жазасыңбы? (Ооба/Жок):

«Y» менен жооп берсек, анда мурдагы маалыматтардын базасындагы маалыматтар өчөт да, структура копияланат.

COPY STRUCTURE командасы жаңы маалыматтар базасына мурунку структуранын талааларынын баяндалыштарын тандап копиялайт. Ал үчүн бул команда FIELDS деген ат менен толтурулат. FIELDS опциясы жаңы маалыматтардын базасына которулуучу талаалардын тизмесин кармап турат.

Маселен, фамилияларды аттарды, атасынын аттарын, туулган даталарын жана окуучунун жынысын белгилөөчү тамганы

кармап турган ANK деген маалыматтардын базасын түзүү керек болсо, анда OM маалыматтардын базасын ачуу керек.

.USE OM

Андан кийин структураны копиялоочу төмөнкү команданы киргизүү керек:

.COPY STRUCTURE TO ANK;
FIELDS GR,FM,AT,AA,TJJ

Төмөнкү команда OM маалыматтардын базасын пайдаланып, студенттердин жетишүүсү жөнүндөгү билдирүүнү JET маалыматтардын базасын түзүүгө мүмкүндүк берет:

.COPY STRUCTURE TO JET;
FIELDS GR,FM,FIZ,MAT,INF,KT

Студенттердин анкеталык маалыматтарынын ичинен JET маалыматтарын базалык жазылыштарынын структурасына студенттин фамилиясы кошулду. Бул болсо ANK жана JET маалыматтардын базалары OM маалыматтардын базасы кандай билдирүүлөрдү кармаса ошондой эле билдирүүлөрдү кармап турат. Төмөнкү сүрөттө (2-сүрөттө) копиялоо командасынын аткарган иши, ANK жана JET маалыматтардын базаларынан келип чыккан структуралардын баяндамасын көрүүгө болот.

```

.USE OM
.COPY STRUCTURE TO ANK FIELDS GR,FM,AT,AA,TJJ
.USE ANK
.LIST STRUCTURE
Structure for data base:A: ANK.dbf
Number of data records: 0
Data of last update: 04/15/93

```

Field	Field name	Type	Width	Dec
1.	GR	Character	8	
2.	FM	Character	12	
3.	AT	Character	12	
4.	AA	Character	12	
5.	TJ	Data	8	
6.	J	Character	1	
**Total **			53	

a)

```

.USE OM
.COPY STRU TO JET FIELDS FM,FIZ,MAT,INF,KT
.USE JET
.LIST STRUCTURE
Structure for data base:A: JET.dbf
Number of data records: 0
Data of last update: 04/15/00

```

Field	Field name	Type	Width	Dec
1.	FM	Character	12	
2.	FIZ	Numeric	1	
3.	MAT	Numeric	1	
4.	INF	Numeric	1	
5.	KT	Numeric	1	
**Total **			16	

б)

2-сүрөт.

COPY STRUCTURE командасы менен түзүлгөн маалыматтардын базасы бош, б.а. информациялык жазылыштарды кармабаган база болот. Аларга жазылыштарды киргизүү APPEND командасынын жардамы менен ишке ашат же толтурулган жана толтурула элек маалыматтардын базаларынын талаалары окшош болгондо ишке ашат.

Экинчи учурда информацияны киргизиш үчүн төмөнкү команда колдонулат:

.APPEND FROM <маалыматтар базасынын аты>

Мында, <маалыматтар базасынын аты> -USE командасы менен ачылган маалыматтардын базасына, маалыматтары кошулуучу маалыматтардын базасынын аты.

OM маалыматтардын базасынан студенттердин анкеталык маалыматтарын COPY STRUCTURE командасы боюнча ANK маалыматтардын базасына көчүрүү үчүн төмөнкү командаларды киргизүү керек:

```
.USE ANK
.APPEND FROM OM
.LIST
```

Ушул командаларды бергенден кийин төмөнкүдөй информация экранда пайда болот (3-сүрөт):

```
.USE ANK
.APPEND FROM OM
  6  Recorde added
.LIST
Record #  GR      FM      AT      AA      J
1  МОЭ-1-99  Аделов  Дамир  Муктаралиевич  Э
2  МОЭ-1-99  Исраилов  Улан  Бурканович  Э
3  МОЭ-1-99  Айдаров  Талант  Ураинович  Э
4  МОЭ-1-99  Бакиров  Алмаз  Кадырович  Э
5  МОЭ-1-99  Рысбаев  Кубаныч  Абсатарович  Э
6  МОЭ-1-99  Сартбаева  Махарам  Сагынбаевна  К
```

3-сүрөт.

Жогоркуга окшош эле JET маалыматтардын базасын маалыматтар менен толтурууга болот (4-сүрөт):

```
.USE JET
.APPEND FROM OM
```

6 records added

```
.LIST
```

Record #	FM	FIZ	MAT	INF	KT
1	Аделов	5	5	4	5
2	Исраилов	4	5	5	5
3	Айдаров	4	4	4	4
4	Бакиров	3	5	5	5
5	Рысбаев	4	4	3	4
6	Сартбаева	5	5	5	5

4-сүрөт.

COPY командасы жазылган гана структураны копиялабастан, маалыматтардын базасындагы жазылыштардагы сакталгандарды бир убакытта копиялайт. Жогоруда көрсөтүлгөндөй төмөнкү команданы киргизүү керек:

```
.USE OM
.COPY STRUCTURE TO ANK;
  FIELDS GR,FM,AT,AA,TJJ
.USE ANK
.APPEND FROM OM
```

Ушул эле жыйынтыкты төмөнкү командаларды киргизүү менен да алууга болот:

```
.USE OM
.COPY TO ANK;
  FIELDS GR, FM, AT, AA, TJJ
```

Мында маалыматтардын базасы, OM маалыматтардын базасы кандай структурага ээ болсо ошондой эле структурага ээ болот. Ал талаа командасына келтирилген жана ошондой эле OM маалыматтардын базасы канча жазылыштарга ээ болсо, ошондой эле жазылыштарга ээ болот.

Эгер биз учурдагы JET маалыматтардын базасынан «орточо

жетишүүсү» деген талаа менен толтурулган жаңы JET_OR маалыматтардын базасына копиялоо керек:

```
. USE JET  
. COPY STRUCTURE TO JET_OR
```

2. JET_OR маалыматтардын базасын OR (орточо жетишүү) талаасы менен толуктап, структураны модификациялоо керек:

```
. USE JER_OR  
. MODIFY STRUCTURE
```

модификациялоонун жыйынтыгы 5-сүрөттө келтирилген.

3. JET_OR маалыматтардын базасына информацияны которуу:

```
. APPEND FROM JET
```

Аткарылган амалдардан кийин JET_OR маалыматтардын базасынын көрүнүшү 6-сүрөттө көрсөтүлгөн. Ушул эле жыйынтыкты COPY командасын пайдаланып алса да болот. Ал үчүн алды менен маалыматтарды JET_OR го копиялоо керек.

```
. USE JET  
. COPY TO JET_OR
```

Андан кийин жаңы маалыматтардын базасынын структурасына OR талаасын кошуп, структураны модификациялоо керек:

```
. USE JER_OR  
. MODIFY STRUCTURE
```

```

.USE JET_OR
.DISP STRUCTURE
Structure for database: A: JET_OR.dbf
Number of data records:
Data of last update: 04Y15Y00
Field Field name      Type      Width      Dec
1.   FM      CHARACTER  12
2.   FIZ     NUMERIC   1
4.   MAT     NUMERIC   1
5.   INF     NUMERIC   1
6.   KT     NUMERIC   1
**Total **           16

```

5-сүрөт.

```

.USE JET_OR
.APPEND FROM JET
6 rrecords added
.LIST
Record #      FM      FIZ  MAT  INF  KT
1           Аделов   5    5    4    5
2           Исраилов 4    5    5    5
3           Айдаров  4    4    4    4
4           Бакиров  3    5    5    5
5           Рысбаев 4    4    3    4
6           Сартбаева 5    5    5    5

```

6-сүрөт.

6-сүрөттө JETOR маалыматтардын базасынын жазылыштарында OR талаасы киргизилген. Бирок ал талааны толтуруш үчүн татаал эмес арифметикалык амалдарды аткаруу керек. Окуучулардын предметтер боюнча бааларын кошуп, баалардын санын бөлүш керек.

Foxbase нин каражаттары мына ушундай операцияларды өзү аткарат. Бул каражаттар маалыматтарды иштетүү каражаттары деп аталат. Аталган каражаттар менен кийинчерээк таанышабыз.

§7 Сорттоо

Кандайдыр бир талаанын маанилеринин өсүү же кемүү тартибинде иретке келтирүү процесси маалыматтардын базасын СОРТТОО деп аталат.

Сорттоо болуп жаткан талаа АЧКЫЧТЫК талаа деп аталат. Натыйжада маалыматтардын базалык жаңы файлы пайда болот. ИЗДӨӨ операциясы маалыматтардын базасына бир гана жазылышты таппастан, тандалыштын шарттарын канааттандырган жазылыштардын тобун ТАНДАЙТ. Маселен, студенттер жөнүндөгү төмөнкүдөй маалыматтарды тандоого болот:

- фамилияларды алфавиттик тартипте;
- жынысы жана туулган жылы боюнча;
- жетишүүнү өсүү жана кемүү тартибинде;

Маалыматтардын базасы ачык болгон учурда жазылыштарды сорттоо SORT (сорттоо) командасы менен ишке ашат. Бул команданын жалпы форматы төмөнкү көрүнүштө болот:

```
.SORT TO < маалыматтардын базасынын аты >  
ON <1-талаанын аты > [ Y D ] , <2-талаанын аты > [ Y D ]
```

Мында <маалыматтардын аты> — сорттолгондон кийин келип чыккан маалыматтардын базасынын аты; <1- талаанын аты > - сорттоонун жогорку ачкычы; <2- талаанын аты > , ... - сорттоонун кийинки ачкычтары; /D - тиешелүү ачкычтык талаанын маанилерин кемүү тартибинде сорттоо учурунда колдонулуучу которгуч. Эгер /D которгучу көрсөтүлбөсө, анда тиешелүү ачкычтык талаанын маанилерин өсүү тартибинде сорттойт. Төмөнкүдөй мисалдарды карайлык:

1-мисал.

```
. USE ANK  
. SORT TO BEN_FM ON FM
```

```
.USE BEN_FM
.LIST
```

Жогорку командалардын аткарылышы 1-сүрөттө көрсөтүлгөн:

```
.USE ANK
.SORT TO BEN_FM ON FM
100% sorted 6 records sorted
.USE BEN_FM
.LIST
Record # GR      FM      AT      AA      TJ      J
1 МОЭ-1-99 Аделов Дамир Муктаралиевич 17.07.81 Э
2 МОЭ-1-99 Айдаров Талант Ураимович 08.03.81 Э
3 МОЭ-1-99 Бакиров Алмаз Кадырович 12.05.82 Э
4 МОЭ-1-99 Исраилов Улан Бурканович 08.04.82 Э
5 МОЭ-1-99 Рысбаев Кубаныч Абсатарович 02.02.80 Э
6 МОЭ-1-99 Сартбаева Махарам Сагынбаевна 16.11.82 К
```

1-сүрөт.

BEN_FM маалыматтардын базасына студенттердин фамилиялары алфавиттик тартипте жайланышкан.

2-мисал

```
.USE ANKETA
.SORT TO BEN_JF ON J, FM
.USE BEN_JF
.LIST
```

Бул командалар аткарылып, алды менен кыздардын, андан кийин балдардын фамилиялары боюнча сорттоо ишке ашат. Демек мында эки ачкыч боюнча сорттоо уюштурулат (2-сүрөт).

```

USE OM
SORT TO BEN_JF ON J, FM
100% sorted 6 records sorted
USE ANC_JF
LIST

```

Record #	GR	FM	AT	AA	TJ	J
1	МОЭ-1-99	Сартбаева	Махарам	Сагынбаевна	16.11.82	К
2	МОЭ-1-99	Аделов	Дамир	Муктаралиевич	17.07.81	Э
3	МОЭ-1-99	Исраилов	Улан	Буркацович	08.04.82	Э
4	МОЭ-1-99	Айдаров	Талант	Ураинович	08.04.81	Э
5	МОЭ-1-99	Бакиров	Алмаз	Кадырович	12.05.82	Э
6	МОЭ-1-99	Рысбаев	Кубаныч	Абсатарович	02.02.80	Э

2-сүрөт.

3-мисал

```

USE OM
SORT TO JET_FIZ ON J/D, FIZ/D, FM
USE JET_FIZ
LIST FIELDS FM, J, FIZ

```

3-мисалдын аткарылышынын жыйынтыгы 3- сүрөттө келтирилген:

```

USE OM
SORT TO JET_FIZ ON J/D, FIV/D, FM
100% sorted
USE JET_FIZ
LIST FIELDS FM, J, FIZ

```

Record #	FM	J	FIZ
1	Аделов	Э	5
2	Бакиров	Э	5
3	Исраилов	Э	4
4	Айдаров	Э	4
5	Рысбаев	Э	4
6	Сартбаева	Э	5

3-сүрөт.

JET_FIV маалыматтардын базасынын жазылыштары төмөнкү тартипте сорттолгон: алды менен уул балдардын,

андан кийин кыздардын маалыматтары каралат. Мындай тартипте кароо J ачкычы жана /D которгучу аркылуу ишке ашат; мындай эки группанын ар биринде физика боюнча баалардын кемүү тартиби каралат; эгер физикадан баалары бирдей болуп калса, анда алфавиттик тартипте каралат.

Эки символдук салыштырууда төмөндөгүдөй эреже орун алары бизге белгилүү :

– каалаган цифра каалаган тамгадан кичине:

– «А»>>9»

– Эки цифранын «кичинеси» кичине:

«5»<<7»

Чоң тамга тиешелүү кичине тамгадан кичине:

«R»<<ч» , «T»<<т»

– Алфавиттин башынан алыс турган тамга, алфавиттин башына жакыныраак тамгадан чоң:

«Б» «Я»

Төмөнкү мисалдарды карап көрөлүк:

'Адълов' < «Айдаров»

'Эшбаева' > «ЭШБАЕВА»

'1917' < '2000'

'12' > '115'

Акыркы мисалдан символдорду жана сандарды салыштыруунун айырмачылыгын көрүүгө болот. ДАТА тибиндеги талаалардын «өсүшүү жана «кемиши» боюнча маалыматтардын базасын сорттоого болот. Мында эки датанын кийинкиси «чоң» болот. ANK маалыматтардын базасын студенттердин туулган даталарын өсүү тартибинде сорттойлук. Ал үчүн төмөнкү командаларды киргизүү керек:

. USE ANK

. SORT TO ZAM_TJ ON TJ

. USE ZAM_TJ

. LIST FM,AT,TJ

Экранда төмөнкүдөй информация пайда болот.

```

.USE ANK
.SORT TO ZAM_TJ ON TJ
  100% sorted   6 records sorted
.USE ZAM_TJ
.LIST GR, FM, AT, TJ
Record # GR      FM      AT      TJ
  1 MOЭ-1-99 Рысбаев Кубаныч 02.02.80
  2 MOЭ-1-99 Аделов Дамир 17.07.81
  3 MOЭ-1-99 Айдаров Талант 08.04.81
  4 MOЭ-1-99 Сартбаева Махарам 16.11.82
  5 MOЭ-1-99 Исраилов Улан 08.04.82
  6 MOЭ-1-99 Бакиров Алмаз 12.05.82

```

4-сүрөт.

§ 8. Маалыматтарды издөө

Маалыматтардын базасындагы информацияларды издөө төмөнкүдөй көрсөтмөлөрдү аткаруу менен ишке ашат:

-Адельев жөнүндөгү маалыматтардан турган жазылыштарды издөө:

-Математикадан беш алган окуучу жөнүндөгү маалыматтардын

жазылышын издөө:

ж.б

Издөөчү көрсөтмө төмөнкүдөй эки бөлүктөн турат: биринчиси эмне кылуу керек - «жазылышты табуу», ал эми экинчиси издөөнүн шартын билдирет - «фамилия-Адельев». Маалыматтардын базасынан издөө LOCATE (локалдоо) командасы аркылуу ишке ашат. LOCATE командасын жалпы түрдө төмөнкүчө жазууга болот:

.LOCATE FOR <шарт>

Мында <шарт> жөнөкөй учурда логикалык туюнтма болот. Анын, сол жагына талаанын аты, ал эми оң жагына издөөнү башкаруучунун мааниси көрсөтүлөт. Логикалык туюнтма сол

жана оң жактары логикалык катыштардын төмөнкүдөй белгилери болушу мүмкүн:

- = «барабар»
- < «кичине»
- > «чоң»
- <> «барабар эмес»
- <= «кичине же барабар»
- >= «чоң же барабар»

Маселе, эгер ОМ маалыматтардын базасындагы окуучу Абдукахаров жөнүндөгү билдирүүчү жазылышты табуу талап кылынса, анда издөөнүн шарты төмөнкүчө жазылмак:

FM= «Абдукахаров»

Эгер барабардыктын оң жагында символдук өзгөрүлмө болсо, анда ал сөзсүз апострофго алыныш керек.

Математикадан беш алган окуучу жөнүндөгү билдирүүчү жазылышты табуу үчүн издөө шартын төмөнкүчө жазабыз:

Мат=5

Мат=5тин оң жагы апострофго алынбады. Себеби, бул талаанын маанилери сан экендиги белгилүү.

LOCATE командасындагы жазылган <шарт> маалыматтарды издөө процессинде ар бир жазылыш үчүн эсептелиш керек. Логикалык туюнтманын мааниси төмөнкүдөй маанилердин бири болушу мүмкүн:

.Т. (англисче True – чын) же .F.англисче False-жалган). Эгерде кезектеги жазылыш үчүн бул маани .Т. га барабар болсо, анда жазылыш табылды дегенди билдирет. Андан кийин LOCATE командасын колдонууга мисал келтирели:

. USE OM

.LOCATE FOR FM= 'Рыспаев'

Натыйжада LOCATE командасындагы издөө шартын беш номерлүү (Record) жазылыш канааттандырат. LOCATE

командасы табылган жазылышты экранга чыгарбастан анын номерин гана чыгарат. Табылган жазылыш учурдагы жазылыш болуп калды. Аны экранга чыгарыш үчүн төмөнкү команданы берүү жетиштүү:

.DISPLAY

LOCATE командасы RECORD N, NEXT N жана ALL опциялары менен да колдонулушу мүмкүн. Бул учурда NEXT N форматы гана мааниге ээ. NEXT N опциясын издөөнү учурдагы жазылыштан баштап, N жазылышка чейин карайт. LOCATE командасына ALL опциясын жазса да, жазбаса да болот. Анткени LOCATE командасы бүт маалыматтардын базасынын башкы жазылыштарынан баштап акыркы жазылышка чейин издейт.

Маалыматтардын базасында LOCATE командасынын издөө шарттын канааттандырган бир нече жазылыштар болушу мүмкүн. Биринчи табылган жазылыштын номери экранга чыккандан кийин, издөөнү улантыш үчүн CONTINUE (улант) командасын берүү керек:

.CONTINUE

Натыйжада команданын издөө шарттын канааттандырган кийинки жазылыштын номери чыгат. Мына ушундай жол менен командадан издөө шарттын канааттандырган бардык жазылыштарды карап чыгууга болот. 1-сүрөттө математикадан беш алган окуучуларды табуу командалары жана издөөнүн жыйынтыктары келтирилген:

```
.USE OM
.LOCATE FOR MAT=5
  Record= 1
.DISP FM, AT
  Record   FM   AT
           Адельев Дамир
```

```
.CONTINUE
  Record= 2
.DISP FM, AT
  Record =   FM   AT
           Исраилов Улан
```

```
.CONTINUE
  Record= 4
.DISP FM, AT
  Record   FM   AT
           Бакиров Алмаз
```

```
.CONT
  Record= 6
.DISP FM, AT
  Record   FM   AT
           Сартбаева Махарам
```

```
.CONTINUE
  End of locate scope
```

1-сүрөт.

CONTINUE командасынан кийин киргизилген DISPLAY командасы табылган окуучунун фамилиясы жана аты экранга чыгат. Эгер LOCATE командасынын издөө шарттын бир дагы жазылыш канааттандырбаса, анда 1- чийменин эң акырындагы билдирүү чыгат (маалыматтардын базасынын акырына жетти).

LOCATE командасынын FOR опциясы татаал логикалык туюнтма болушу мүмкүн. Бул туюнтма логикалык операциялардын белгилери менен байланышкан. Негизги логикалык амалдар төмөндөгүлөр:

- . AND. (жана) - логикалык көбөйтүү;
- . OR. (же) логикалык кошуу;
- . NOT. (эмес) - тануу.

Мында AND. опциясы төмөндөгүдөй көрсөтмөлөрдү издөө шарттарын түзүү үчүн колдонулат: «Кыргыз тил жана математикадан беш алган окуучулардын маалыматтары жөнүндөгү жазылышты тапкыла же информатикадан төрт жана беш алган кыздардын маалыматтары жөнүндөгү жазылышты тапкыла». OM маалыматтардын базасынан мындай жазылыштарды издөө шарттары төмөнкүдөй болот.

KT=5.AND.MAT=5
J=>K».AND.INF>=4

Экинчи татаал издөө шарттын канааттандырган OM маалыматтардын базасындагы жазылыш төмөндөгүдөй болот:

```

. USE OM
. LOCATE FOR J=>k».AND.INF>=4
  record = 6
. DISP FM,INF
  record#   FM      INF
           6  Сартбаева  5
. CONT
  End of llocate scope

```

2-сүрөт

Мында логикалык операция .OR. татаал логикалык шартты жазыш үчүн колдонулат. Татаал логикалык шарт жөнөкөй логикалык шарттардан турат. Татаал шарттын мааниси .T. болуш үчүн жөнөкөй логикалык шарттардын жок дегенде бири чын болуш керек. Мисалы, OM маалыматтардын базасындагы математика жана информатика предметтеринин жок дегенде биринен үч деген баа алган окуучуну табуунун издөө шарттын төмөнкүчө жазууга болот:

MAT=3 .OR .INF =3.

Анда издөөнүн жыйынтыгы төмөндөгүдөй болот (3-сүрөт):

```
. USE OM
. LOCATE FOR MAT=3.AND.INF=3
  record = 5
. DISP FM , MAT,INF
  record#  FM    MAT  INF
          5  Рысбаев  4    3
. CONT
  End of lacope scope
```

3-сүрөт.

Логикалык операция .NOT. физикадан «эки» алган

```
. NOT. FIZ = 2
```

математикадан төрттөн кем эмес баа алган

```
. NOT.MAT.<=3
```

«студенттердин жазылыштарын тапкыла» деген сыяктуу көрсөтмөлөр боюнча издөөнү жүргүзөт. Жогорудагы логикалык үч операциянын жардамы менен татаал логикалык туюнтманы да түзүүгө болот. Бул учурда төмөнкүдөй программанын фрагментин келтирүүгө болот:

```
. USE OM
. LOCATE FOR .NOT. FM= 'Сартбаева'.AND.;
(FIZ=5.OR.MAT=5)
```

Бул командалар боюнча издөөнүн жыйынтыгы төмөндөгү таблицанда келтирилген (4-сүрөт):

```

USE OM
LOCATE FOR .NOT. FM= 'Сартбаева'.AND.;
(FIZ=5.OR:MAT=5)
Record= 1
.DISP FM, FIZ, MAT
Record #      FM      FIZ      MAT
      1      Адельев      5      5

.CONT
Record =      4
.DISP FM, FIZ, MAT
Record #      FM      FIZ      MAT
      4      Бакиров      5      5

.CONT
End of locate scope

```

4-сүрөт.

Бул издөөдө Сартбаевадан башка физика же математикадан беш алган окуучулардын жазылыштары табылат.

§ 9. Маалыматтарды тандоо

«Издөө» жана «тандоо» деген түшүнүктөрдү айрып билүү зарыл. Издөөнүн максаты - LOCATE командасындагы шартты канааттандырган маалыматтардын базасындагы жазылыштардын ордун аныктоо болуп эсептелет. Издөөнүн натыйжасында табылган жазылыш учурдагы жазылыш болот. Эгер LOCATE командасын кошо колдонгон болсок, анда колдонуучуга табылган жазылыштын номери да баяндалат.

Ал эми тандоо өзүнчө издөөнү камтып турат. Бирок тандоонун максаты жазылыштардын ордун аныкташ эмес, анын максаты маалыматтардын жаңы базасын түзүү болуп эсептелет. Бул жаңы түзүлүүчү маалыматтардын базасы берилген шартты канааттандырган жазылыштардан туруш керек. Мындай шарт тандоонун шартты деп аталат.

Маалыматтарды тандоо опциясы FOR жана COPY командасы пайдаланылат. Бул команданын форматы төмөнкү көрүнүштө болот:

. COPY TO <файлдын аты> FOR <шарт>

Бул команда боюнча <шартты> канааттандырган жазылыштар жаңы түзүлүп жаткан маалыматтардын базасына көчүрүлөт. Ал эми <файлдын аты> жаңы түзүлгөн файлдын аты болот. FOR до көрсөтүлгөн тандоо шартты LOCATE командасынын FOR опциясындагы логикалык шарттардай эле аныкталышат.

1-сүрөттө OM маалыматтардын базасын тандоо үчүн COPY командасынын колдонушу предметтер боюнча жок дегенде бир үчү бар студенттердин жазылыштарын, жаңы маалыматтардын базасын ORTO базасына көчүрүү керектелген.

USE OM

COPY TO ORTO FOR FIZ=3.OR.MAT=3.OR.INF=3.OR.KT

2 records copied

USE ORTO

.LIST FIEL FM, AT, AA, FIZ, MAT, INF, KT

Record #	FM	AT	AA	FIZ	MAT	INF
----------	----	----	----	-----	-----	-----

1	Айдаров	Талант	Ураимович	4	3	4
---	---------	--------	-----------	---	---	---

2	Рысбаев	Кубаныч	Абсатарович	3	4	4
---	---------	---------	-------------	---	---	---

1-сүрөт.

Жаңы түзүлгөн маалыматтардын базасынын үстүнөн төмөнкү операцияларды аткарууга болот: структураны модификациялоо; отчетторду иштеп чыгуу жана кагазга чыгаруу. (печаттоо) жана башкалар.

Маалыматтарды иштеп чыгуу

§ 10. Турактуулар өзгөрүлмөлөр жана туюнтмалар

Маалыматтарды иштеп чыгуу деп, алардын үстүнөн жүрүзүлгөн ар түрдүү амалдарды түшүнөбүз. Маалыматтар базасынын учурундагы жазуусунун талааларын маанилерин иштеп чыгууга дуушар да болобуз. Маалыматтардын үстүнөн амалдарды жүргүзүүдө турактуулар жана өзгөрүлмөлөр колдонулат.

Турактуулар программада өзгөрүлбөгөн алдын-ала аныкталган мааниге ээ болушу мүмкүн.

Мисалы: «Каалаган клавишаны бас»; 3; 7; 5; Т., { 1,3,5}.

Өзгөрүлмөлөр программанын иштөө мезгилинде же кандайдыр амалдын аткарылышынын негизинде жаңы мааниге ээ болот. Алар оперативдик эстин областтарында адрестөө каражаты болуп саналат. Ар бир өзгөрүлмөнүн аты жана мааниси бар. Программанын тиги же бул ордунда өзгөрүлмөнүн атын колдонгондо, анда анын маанисин же ал маанини өзгөртүү деп түшүнүү керек.

Мисалы, $a = a + 2$

Демек, бул амалдын негизинде a өзгөрүлмөсүнүн маанисин 2 ге чоңойду.

Өзгөрүлмөнүн негизги мүнөздөөчүсү болуп анын классы эсептелет.

Foxbase системасында өзгөрүлмөлөр эки класска тиешелүү.

Алар:

- 1) глобалдык
- 2) локалдык

Глобалдык өзгөрүлмө программанын каалаган бөлүгүнүн аткарылуу процессинде түзүлөт жана ал программанын

жойгонго чейин жашайт. Бул класстагы өзгөрүлмөлөрдү программанын каалаган участогунда колдонууга болот. Глобалдык өзгөрүлмөлөр **Public** командасы менен баяндалат. Форматы

Public <өзгөрүлмөнүн аттары>

Логикалык өзгөрүлмө өзү баяндалган программалык блокто түзүлөт жана **PRIVATE** командасынын жардамы аркылуу баяндалат, бул класстагы өзгөрүлмөлөр программа иштелип бүткөндө өчүрүлөт. Команданын форматы:

PRIVATE <өзгөрүлмөнөн аттары>

Эки класстагы өзгөрүлмөлөр 4 типте болушу мүнкүн:

- Сандык N;
- Символдук C;
- Логикалык L;
- Календардык дата D.

Өзгөрүлмөлөрдүн тиби ага берилген маанисине карата аныкталат. Эгерде талаанын аты менен өзгөрүлмөнүн аты бирдей болсо, анда аткарылуучу амал талаанын үстүнөн жүргүзүлөт. Өзгөрүлмөлөргө кайрылуу үчүн өзгөрүлмөнүн атынын алдына «M→» үч символду жазуу керек.

Бир же бир нече операнддарды амалдардын белгилери менен бириктирилип жазылышы туютма деп аталат. Туонтмалар баштапкы маалыматтардын үстүнөн аныкталган амалдарды аткаруу менен жаңы маалыматтарды түзүү үчүн кызмат кылат. Туонтмаларда амалдар солдон оңго карай аткарылат. Эгерде амалдардын аткарылуу тартибин өзгөртүүгө туура келсе, анда кашааларды колдонобуз. Амалдар бир типтеги маалыматтардын үстүнөн жүргүзүлөт.

Ар түрдүү типтеги маалыматтардын үстүнөн жүргүзүлгөн амалдарды карап көрөлү. Сандык амалдар сандык типтеги операнддардын үстүнөн жүргүзүлөт. Амалдын жыйынтыгы сан болот. Алар стандарттык арифметикалык амалдарга негизделген:

оператор	амал
+	кошуу;
-	кемитүү;
*	көбөйтүү;
/	бөлүү;
** же ^	даража көтөрүү;
%	бүтүнгө бөлүүдөн калган калдык.

Логикалык амалдар логикалык типтеги маалыматтардын үстүнөн жүргүзүлөт жана амалдын жыйынтыгы логикалык мааниге ээ.

Оператор * амал

.AND. логикалык «жана» -эгерде эки операнддык мааниси *t* болсо, анда туют манын мааниси да *t*. болот. Ал эми жок дегенде бири эле *f*. болсо, анда мааниси *t*.болот.

.OR. логикалык «же» - эгерде эки операнддардык тын бири *t*. болсо, туютманын мааниси *t*. болот, ал эми экөө тең *f*. болсо *T*. мааниси *f*. блот.

Not. же! логикалык «жок» - эгер оператив, *t*. болсо, *T*. мааниси *f*. ал эми операнддык *f*. болсо *T*. мааниси *t*. болот.

Катыш амалдарды бинардык болот. Алар эки операцияны бир типте болушун талап кылат. Эгерде текшерилүүчүнүн шартты чын болсо, амалдын жыйынтыгы *T*. болот, ал эми текшерилүүчүнүн шартты жалган болсо *f*. болот.

Оператор.	Амал.
=	барабар
!=, ?	барабар емс
<	кичине
>	чоң

<=

кичине же барабар

>=

чоң же барабар

Катыш амалы бир жолчонун экинчи жолчодо камтылуусун аныктоо үчүн да колдонулат. Бул амал \$ белгиси менен белгиленет.

Өзгөрүлмөлөргө маани берүү үчүн эки ыкма колдонулат. Алардын бири STORE командасы. Команданын форматы төмөнкүчө:

.STORE <туюнтма> TO <өзгөрүлмөнүн командасы>

Бул команда <өзгөрүлмөнүн командасы> көрсөтүлгөн өзгөрүлмөлөргө <туюнтманын> маанисин ыйгарат.

Мисалы: STORE 5.5 to x
STORE x^2 to y,z

Биринчи команда аткарылганда x – өзгөрүлмөсү - 5.5 маанисине ээ болот. Экинчи команда y, z өзгөрүлмөлөрү, 30,25 маанисине ээ болушат. Маани берүүчү командалардын экинчиси (=) командасы. Анын форматы <өзгөрүлмө>=<туюнтма>. Бул команда <туюнтманын> маанисин <өзгөрүлмөгө> ыйгарат.

Мисалы: x=3.2
y=5*x+6

Өзгөрүлмөлөрдүн маанисин көрүү үчүн DISPLAY MEMORY же LIST MEMORY командасын колдонсо болот. Өзгөрүлмөлөрдү эстен жоюу RELEASE командасы менен аткарылат.

Мисалы: RELE ALL – бар өзгөрүлмөлөрдү жоюу
RELE A2, A2- A1,A2 өзгөрүлмөлөрдү жоюу
RELE ALL LIKE AT- 'AT' менен башталган өзгөрүлмөлөрдүн бардыгын жоюу
RELE ALL EXCERT AT- 'AT' менен башталган өзгөрүлмөлөрдүн бардыгын жоюу.

Зарыл учурда өзгөрүлмөлөрдүн мааниси менен бирдикте «Мем кеңейтилишине» ээ болгон файлга сактоого болот. Ал SAVE TO командасы менен ишке ашат, ал эми команданын форматы төмөнкүчө:

.SAVE TO <файлдын аты>

Сакталган өзгөрүлмөлөрдү калыбына келтирүү RESTORE FROM <файлдын аты> командасы менен аткарылат. Жаткан прикладдык программаны иштөө процессинде өзгөрүлмөлөрдүн маанилерин клавишадан киргизүү үчүн ACCEPT жана INPUT командасы пайдаланылат. ACCEPT командасы символдук маанилерди, INPUT командасы сандык жана логикалык маанилерди киргизүү үчүн колдонулат. Эки команда жол көрсөтүүчү чыгарууга мүмкүндүк түзөт.

Мисал: ACCEPT «сөз киргиз» TO x

Бул команда x өзгөрмө аргументи символдук маани берүү үчүн колдонулат.

INPUT «сан киргиз» TO y

Команда y өзгөрүлмөсүнө сандык маани берет. Эгерде INPUT командасы менен өзгөрүлмөгө символдук маани бере турган болсо, анда ал маанини тырмакча га же апострофага алып берүү керек.

Туюнтмалардын, өзгөрүлмөлөрдүн же жазуулардын тамгаларынын маанилеринин экранга чыгуу үчүн ? комдасы же ?? командасы пайдаланылат. ?? командасы маанилерди ошол жолчого, ал эми ? командасынын өзгөрүлмөнүн маанисин жаңы жолчого чыгарат.

Команданын форматы төмөндөгүдөй:

? <туюнтма>

Мисалы,

? a1 – a1 өзгөрүлмөлөрүнүн маанисин чыгарат.

? x,y – x,y өзгөрүлмөлөрүнүн маанисин чыгарат.

§ 11. Кээ бир SET командалар

Колдонмо программаларды түзүү үчүн системалык иштөө параметрлерин жана режимдерин тактоо керек. Ал үчүн SET командасы колдонулат. Анын форматын эки түрдө жазууга болот.

1. SET < режимдин аты > on / off

2. SET < параметрдин аты > to < параметрдин мааниси >

1- форматтагы командасы on жана off кызматчы сөздөрүнө байланыштуу режимди тургузат же жокко чыгарат.

Программаны жөнгө салуу үчүн төмөнкүдөй SET командаларын колдонобуз:

Set TALK ON/off –ар бир команданын аракетинин жыйынтыгын берүү орнотот;

SET ECHO on/ OFF – аткарылуучу команданы чыгарууну орнотот /жоет;

SET STEP on/ OFF – ар бир командадан кийин программаны токтотууну орнотот /жоет;

SET DEBUS on/ off – SET ECHO командадагы информацияларды принтерге чыгарууну орнотот / жоет;

SET BELL on/off - маалыматтарды киргизүүдө талаанын аягына жетсе же ката кетирсе, үн сигналын берүүнү орнотот / жоет;

SET CONSOLE on/off – командалар аткарылгандагы жыйынтыктарды экранга чыгарууну орнотот / жоет;

SET EXACT on/off - символдук жолчолорду салыштырууда так дал келүүнүн зарылчылыгын орнотот / жоет;

SET PRINT on /off – принтерди чычканды орнот / жоет;

SET ESCUPE on/off- <ESC> клавишасын басканда процессти токтотууну орнотот / жоет.

SET MENU on/off –толук экрандын командалары үчүн курсорду башкаруунун сурап-билүү информациясын чыгарууну орнотот / жоет;

SET STATUS on/off – экрандын 22- жолчосунда статус жолчону чыгарууну орнотот;

SET HEADING on/off –DISPLAY, LIST, SUM, AVERAGE

командаларында талаанын аттары жана

Record сөзү чыгат, орнотот / жоет;

§ 12. Негизги стандарттык функциялар

Математикалык амалдардын жардамы менен учурдагы жазуунун сандык маанилүү талаасы, турактуулар жана өзгөрүлмөлөр менен амалдарды аткарууга болот. Ошондой эле математикалык функциялар да колдонулат. Алар функциялар сандык аргументтер менен амалдарды жүргүзөт жана жыйынтыгы сан болот. Функциянын аргументти болуп турактуу туюнтмалардын формулалары боло алат. Ал формулалар төмөндөгүлөр:

№	Функция	Кызматы
1	ABS(X)	X тин абсолюттук чоңдугу
2	EXP(X)	Натуралдык логарифмдин негизин даражага көтөрүү
3	LOG(X)	X тин натуралдык логарифми
4	Sqrt(X)	Xтин квадраттык тамыры
5	Round(x,n)	X анык санынын бөлчөк бөлүгүнүн чекиттен кийин N ге чейин тегеректөө
6	Mod(X1,X2)	X1ди X2 ге болгондогу калдык
7	Max(X1,X2)	X1 жана X2 нин чоңу
8	Min(X1,X2)	X1 жана X2 нин кичинеси
9	INT(X)	X тин бүтүн бөлүгү

Мисалы: ABS (-5.75)=5.75

INT (6.2)=6

Round (-5.6234,3)=-5.623

Mod (8,3)=2

Min (3.7,2.5)=2.5

Символдук маалыматтарды иштеп чыгууда ар түрдүү амалдарды аткарууга туура келет. Мындай маалыматтарды иштетүү үчүн атайын функциялар бар.

№	Функция	Кызматы
1	CHR(X)	X санын символго өзгөрөтүү (x=1,2,...,255)
2	LEN©	С символдук жолчосунун узундугун аныктайт
3	AT(C1,C2)	C1 жолчосун C2 жолчосунан издейт; эгерде C1 жолчосу C2 жолчосуна камтылса, анда C1 жолчосу баштапкы позициянын номерин берет, ал эми камтылбаса, Одү берет

Сандык константтардан, өзгөрүлмөлөрдөн жана математикалык функциялардан арифметикалык туюнтмаларды түзүүгө болот.

$$2.5+x**2*SQRT(x);$$

$$A*EXP(x)Y(C+LOG(x))+2.506$$

Математикалык тилинде булар төмөндөгүдөй жазылат:

$$2,5+x^2+2\sqrt{x};$$

$$\frac{A \cdot e^x}{C + LOG(x)} + 2.506;$$

Төмөнкү арифметикалык туюнтманын маанисин эсептөө программасын түзгүлө.

$$(1.4x^2 + \sqrt{x}) = \sqrt{(e - \ln x)}$$

$$.X=2.126$$

$$2.126$$

$$.A=x**2$$

$$4.526$$

$$.B=SQRT(x)$$

$$1.458$$

$$C=EXP(x)$$

$$8.381$$

$$.D=LOG(x)$$

$$0.754$$

$$.Y=(1.4*A+B)/SQRT(C-D)$$

$$2.8192$$

$$.? ROUND (Y.2)$$

$$2.8200$$

$$.? ROUND (Y.0)$$

$$3000$$

$$.INT(Y)$$

$$2$$

Ү өзгөрүлмөсүнө туюнтмасынын мааниси ыйгарылат да ROUND функциясы аркылуу ар түрдүүчө тегеректелип, Ү тин маанилери экранга чыккандыгы көрүнүп турат. Мындагы суроо(?) белгиси PRINT командасы экендиги бизге белгилүү.

Константалардын, өзгөрүлмөлөрдүн жана маалыматтардын базасынын жазылыштарынын маанилерин STR(x) функциясынын жардамы менен символдук түргө өзгөртүүгө болот. STR функциясынын аргументи X, учурунда арифметикалык туюнтма болушу мүмкүн. Сандык маанилер арифметикалык амалдардын операнддалары болот. Ал эми ага тиешелүү символдор жана тамгалар литералар чынжырчасы түзүлөт биз билебиз. Бул чынжырчаларга математикалык функциялар колдонулбайт. Бирок STR функциясы үчүн dbase ге белгилүү амалдар аткарылат.

STR функциясынын жалпы форматы төмөнкү көрүнүштө болот:

STR (<арифметикалык туюнтма>,[<узундугу>],[<dec>])

Бул функция символдордун жыйынтыктоочу узундугун чектөөгө мүмкүндүк берет. Бул болсо милдеттүү эмес <узундук> параметри аркылуу болот. Ошондой эле сандык маанилердин бөлчөктүү разряддарынын санын символдорго айландырып чектөөгө болот (<dec> маанилүү эмес параметри). STR функциясынын колдонулушу көрсөтүлгөн:

```
.. ? (-1424.148)
-1424
.. ? STR (-1424.148,9,3)
-1424.148
.. ? STR (-1424.148,2)
-1424.15
.. ? STR (-1424.148,8)
-1424
.. ? STR (-1424.148,7,3)
*****
.. ? 15+17
32
```

.? STR (15)+17
(DATA TYPE MISMATCH)

(команда не опознана)

.? STR (15)+17

(DO YOU WANT SOME NHELP ? (Y/N)

OPERATOR / OPERAND TYPE MISTACH

.? STR (15)+'17'
1517

Жогорку мисалдардан төмөнкүлөрдү байкоого болот. Эгер <узундук> параметри жетишээрлик көрсөтүлбөсө, анда сандык маани, STR функциясында көрсөтүлгөн жылдызчалардын жолчосуна өзгөрүлөт. Акыркы жолчодон мурунку жолчодо STR(15) ти 17 ге кошууга аракет кылынган Бул туюнтма туура эмес түзүлгөн. Себеби STR(15) символдук жолчо, ал эми 17 сан болуп эсептелет. Ошондуктан «биригишпеген маалыматтардын тиби» деген информация пайда болду. Эң акыркы туюнтма туура. Анткени STR(15) жана '17' лер символдук жолчолор. Бул туюнтмадагы «+» белгиси конкатенация амалы экендиги 1-главадан бизге белгилүү. Ошондуктан STR(15)+'17'ге барабар болду.

§ 13. Программа түзүүгө мисалдар

Foxbase – программа сүйлөмдөн турат. Ар бир сүйлөм бир жолчого жазылат. Эгерде сүйлөм бир жолчого сыйбаса, анда аны экинчи жолчого улантуу үчүн; символ колдонулат. Программада командаларды жазууда төрт тамгасын жазуу жетиштүү. Ошондуктан, биз төрт тамганы гана жазалы. Каалагандай программа RETURN командасы менен аяктайт

1-мисал. үч бурчтуктун үч жагы берилген. a=5, b=7, c=6. Аянтын тапкыла.

Бул мисалга программа түзөлү:

```
set talk off  
a=5  
stop 7 to b  
stor 6 to c
```

```

P=(a+b+c)|2
S= sgrt (p*(p-a)*(p-b)*(p-c))
?s
set talk on
retu
экранга 14.6969 саны чыгат.

```

2-мисал. Тик бурчтуу үч бурчтуктун катеттери берилген, гипотенузасын тапкыла. Бул мисалга программа түзөлү

```

set talk off
input «1-катетин киргиз « to a
input «2-катетин киргиз « to b
c=sgrt (a**2+b**2)
?c
retu

```

экранга «1- катетин киргиз » информациясы чыгат, ага 7 маанисин берели, «2-катетин киргиз» информациясы чыгат. Ага 9 маанисин берели. Натыйжада экранга 11.4 саны чыгат.

3-мисал. $\Delta S=2,9$ м өтүү. Телонун ылдамдануусун аныктагыла. 6 секундадагы тело кыймылдагандагы өтүлгөн жол

$$\Delta S = S_6 - S_5 = (V_0 t_6 + at_6^2 | 2) - (V_0 t_5 + at_5^2 | 2)$$

$$\Delta S = V_0 (t_6 - t_5) + a (t_6^2 - t_5^2) | 2, \text{ мындан а ны аныктасак,}$$

$$a = 2(\Delta S - V_0 (t_6 - t_5)) / (t_6^2 - t_5^2)$$

```

set talk off
S1=2.9
V0=4
T6=6
T5=5
A=2*(S1-V0*(t6-t5))/(t6**2-t5**2)
?a
set talk on
retu
экранга -0.2 чыгат.

```

FOX BASE системасында сызыктуу тармактануучу,
циклдик программаларды түзүү

§ 14 Башкаруучу структуралар

Башкаруучу структуралар программанын аткарылуу удаалаштыгын өзгөртүү үчүн колдонулат. Башкаруучу структуралардын башталышынын жана аягынын аныкталышы ошондой эле бир нече сүйлөмдүн түзүлүшү программанын башка элементтеринен айырмаланышы болуп саналат. Башкаруучу структуралар камтылган болушу да мүмкүн. Бул учурда бир структура экинчи структурада толугу менен камтылган болушу керек.

Программанынын аткарылышынын эки тармагынын бирин тандоо үчүн кандайдыр шартты текшерүүгө дуушар болгон учурлар кездешет. Мындай учурда IF командасын пайдаланылат. Бул команданын толук жана толук эмес форматтары бар. Толук эмес форматы төмөндөгүчө:

```
.IF<шарт>
<команданын группасы>
ENDIF
```

Команданын негизги кызматы: эгер <шарт> чын мааниге ээ болсо, анда <командалардын группасы> аткарылат, ал эми <шарт> жалган мааниге ээ болсо, анда башкаруу ENDIF кызматчы сөзүнөн кийинки командага берилет. Команданын толук форматы төмөндөгүчө:

```
.IF <шарт>
<команданын 1-группасы>
ELSE
<команданын 2-группасы>
ENDIF
```

Команданын негизги кызматы: эгер <шарт> чын мааниге ээ болсо, анда (команданын 1-группасы) аткарылып, башкаруу ENDIF кызматчы сөзүнөн кийинки команда берилет; эгер <шарт> жалган мааниге ээ болсо, анда <команданын 2-

группасы> аткарылат, башкарууну ENDF кызматчы сөзүнөн кийинки командага берилет.

IF командасы камтылган учуру төмөнкү форматка ээ:

```
.IF <шарт>
<команданын 1-группасы>
ELSE IF <2-шарт>
<команданын 2-группасы>
...
ELSE IF <п-шарт>
<команданын п-группасы>
ELSE
<команданын (п+1) группасы>
ENDIF
...
ENDIF
```

Мында <1-шарт>, <2-шарт>, ..., <п-шарт> <команданын 1-группасы>, <команданын 2 группасы>, ..., <команданын п-группасы> аткарышын аныктоочу логикалык туюнтмалар, <команданын п+1 группасы> - бардык шарттар аткарылбаган учурда аткарылуучу программанын бөлүгү.

1-мисал: $ax^2+bx+c=0$ квадраттык теңдеменин тамырын табуу талап кылынсын.

```
set talk off
input «a ны киргиз» to a
input «b ны киргиз» to b
input «c ны киргиз» to c
d=b**2-4*a*c
y=2*a
if d>=0
x1=(-b+sgrt(d))/y
x2=(-b-sgrt(d))/y
? x1,x2
else
? «теңдеменин тамыры жок»
endif
```

```
set talk on
retu
```

Программа аткарылып, экранга «аны киргиз» информациясы чыккандан кийин клавиатурадан 5ти киргизели. Андан кийин «b ны киргиз», ага бны берели жана «с ны киргиз», ага 3тү берсек экранга теңдеменин тамыры жок деген информация чыгарылат. Эгерде жогорку эле суроолорго тиешелүү түрдө 1, 5, -6 сандарын ыйгарсак анда 1, -6 жооптору чыгат.

IF командасын камтыган учуруна мисал келтирели.
2-мисал: x_1, x_2, x_3 сандарынын максимумун тапкыла

```
set talk off
input '1- санды киргиз' fo  $x_1$ 
input '2- санды киргиз' to  $x_2$ 
input '3 - санды киргиз' to  $x_3$ 
if  $x_1 > x_2$ 
if  $x_1 > x_3$ 
MAX=  $x_1$ 
Else
Max= $x_3$ 
Endi
Else if  $x_2 > x_3$ 
Max=  $x_2$ 
Else
Max= $x_3$ 
Endif
? max
set talk on
retu
```

OM маалыматтар базасынан Абдыкахаровдун алган бааларын издөөнун эки вариантын көрсөтөлү. Биринчи вариантында LOCATE командасын, экинчи вариантында SEEK командасын колдонобуз.

3-мисал: *LOCATE командасынын жардамы менен

```

Set talk of
set exas on:
@ 0,0 clear
acce «фамилияны киргиз» to a
?
USE OM
loca for fm=a
if eof ()
? a « OM файлында жок»
else
? fm, mat,fiz,inf,kt
endif
set talk on
retu

```

Экранга «фамилияны киргиз» информациясы чыгат. Эгерде Адельев деп киргизсек, анда анын математикадан, физикадан, информатикадан, кыргыз тилинен жетишүү баалары чыгарылат. Ал эми OM файлында жок фамилия киргизилсе, экинчи вариантын түзүүдө OM файлы FM талаасы боюнча индекстелген болуу керек. Индекстелген файлдын аты FMOM болсун. Программа төмөнкүдөй болот:

```

4-мисал. * SEEK командасынын жардамы менен
Set talk off
Set exas on
@ 0,0 clea
acce «фамилия киргиз» to a
?
use OM inde fmom
seek a && мында FIND командасын колдонсок да болот
if eof ()
? a +' фамилиясы жок'
else
? fm, mat, fz, in, kt
endi
set talk on
retu

```

Эгерде файлдын көлөмү чоң болсо, анда экинчи вариантты пайдалануу ыңгайлуу.

Экранга «фамилия киргиз» информациясы чыккандан кийин Абдыкахаров деп берсек, анда 3-мисалдын жыйынтыгын алабыз.

IF командасы программанын аткарылышынын эки тармагынын бирин тандаарын жогоруда карап көрдүк. Программада экиден көп тармак болгон учурда камтылган IF командасын колдонсо болоору 2-мисалда келтирилди. Эми экиден көп тармак болгон учурда атайын DO CASE командасын колдонуу ыңгайлуу. Бул команда да эки форматка ээ. Биринчи формат төмөндөгүдөй:

```
.DO CASE  
CASE <1-шарт>  
< командалардын 1 – группасы>  
CASE <2-шарт>  
< командалардын 2-группасы>  
.....  
CASE <n-шарт. >  
< командалардын n-группасы>  
END CASE
```

Команданын негизги кызматы: Эгерде <1-шарт> чын мааниге ээ болсо, анда <командалардын 1-группасы> аткарылып, башкаруу END CASE кызматчы сөзүнөн кийинки командага ыйгарылат, эгер <1- шарт> жалган мааниге ээ болсо, анда <2-шарт> текшерилет <2-шарттын > мааниси чын болсо, анда <командалардын 2-группасы> аткарылат, башкаруу END CASE командасынан кийинки командага берилет; эгер <2-шарт.> жалган мааниге ээ болсо, анда <3-шарт> текшерилет ж.б.у.с .

Эми команданын. 2-форматын келтирели;

```
.DO CASE  
CASE <1-шарт >  
< командалардын . 1-группасы >  
CASE <2-шарт >  
< командалардын 2-группасы >
```

...

```

CASE <.n-шарт >
<командалардын n-группасы >
OTHERWISE
< командалардын(n+1)-группасы >
END CASE

```

Командасынын аткарылышы 1- форматтагы команда сыяктуу эле. Бирок бардык шарттар жалган мааниге ээ болсо, анда < командалардын (n+1)-группасы > аткарылып, башкаруу ENDC командасынан кийинки командага ыйгарылат.

5-мисал. Экинчи мисалды DO CASE командасынын жардамы менен түзөлү.

```

* DO CASE командасынын жардамы менен
SET talk off

```

```

Inpu '1-санды киргиз ' to x1
Inpn ' 2-санды киргиз ' to x2
Inpu ' 3- санды киргиз ' to x3
Do CASE
CASE x1>x2
Do case
Case x1>x3
Max=x1
Othe
Max=x3
Endc
Case x2>x3
Max=x2
othe
Max=x3
Endc
? max
set talk on
retu

```

Циклдик программаларды түзүү үчүн DO WHILE командасын колдобуз. Команданын форматы төмөндөгүдөй.

```

.DO WHILE < шарт >
<циклдин телосу>

```

ENDDO

-мында <шарт> циклдин аткарылуу шарттын аныктоочу логикалык туюнтма. Командасынын негизги кызматы: <шарт> жалган мааниге ээ болгонго чейин циклдин телосу аткарыла берет. Жалган мааниге ээ болгондо башкаруу ENDDO кызматчы сөзүнөн кийинки командага берилет. Эгерде <шарт> биринчи текшерилгенде эле жалган мааниге ээ болсо, анда цикл бир да жолу аткарылбайт. Башкаруу ENDDO кызматчы сөзүнөн кийинки командага ыйгарылат.

6-мисал. 1 ден n ге чейинки сандардын суммасын тапкыла.

Бул мисал үчүн программанын эки вариантын көрсөтөлү.

1-вариантта IF командасын, экинчи вариантта DO While командасын пайдаланабыз.

1- вариант.

IF командасын пайдаланып эсептөө

set talk off

Stor o to s

Stor 1 to k

Inpu 'n ge маани бер ' to n

If k<=n

S=s+k

K=k+1

Endi

? s

set talk on

retu

2-вариант.

* DO WHILE командасынын жардамы менен эсептөө варианты

set talk off

stor o to s

stor 1 to k

inpu 'n ge маани бер ' to n

DO WHIL k<=n

S=s+n

```
k=k+1
endd
? s
set talk on
retu
n=10 s=55
```

7- мисал. OM файлындагы отличниктердин фамилияларын издөө.

```
* отличниктердин тизмеси:
Set talk off
Use OM
If eof ()
Do case
Case mat=5 .and. fiz=5 .and. inf=5 .and.kt=5
Do otl
Endc
Loop
Endif
Set talk on
Retu
Эми otl программалык файлын түзөлү.
Set talk off
? fm
set talk on
retu
```

8-мисал. OM файлындагы бир сабактан жетишпеген окуучунун тизмеси.

бир сабактан жетишпеген тизме.

```
* жетишпегендердин тизмеси.
Set talk off
Use om
If eof ()
Do case
Case mt=2 .or. fiz=2 .or. inf=2 .or.kt=2
Do jt
Endc
Loop
```

```

Endif
Set talk on
Retu
Эми jt программалык файлын түзөлү.
Set talk off
? fm
set talk on
retu

```

9-мисал. Сөз берилген «а» тамгасы катышкандыгын аныктоо

```

set talk off
@ 0,0 clea
acce ' сөз киргиз' to x
y=len (x)
k=1
DO whil k<=y
  B=sulstr (x,k,1)
  If b='a'
    ? x+ 'сөзүндө а тамгасы бар'
  exit
endi
k=k+1
enddo
set talk on
retu

```

Циклдин телосунда LOOP жана EXIT командаларын пайдаланууга болот. LOOP командасы башкаруучу циклдин башталышына ыйгарат, б.а. кайрадан <шарт> текшерилет. Ал эми EXIT командасы циклден чыгууга аргасыз кылат.

§ 15. Бир нече маалыматтар базасы менен бир учурда иштөө

Чыныгы маалыматтар базасы бир нече маалыматтар файлдарынан турат. Көпчүлүк учурда ал файлдарды бир убакта иштетүүгө туура келет. Маалыматтардын базасынын башкаруу системасынын каражаттары менен бир убакта 10 го

чейинки ар түрдүү маалыматтар файлдарын иштетүүгө болот. Ар бир маалыматтар файлы өз алдынча жумушчу областтарда ачылышы керек. Бир файл бир гана жумушчу областта ачылат. – файлы менен бирдикте жумушчу областта индекстелген файлдар, форматтык файлдары ачыла алат. Жумушчу областта тандоо SELECT командасы менен аткарылат. Команданын форматы төмөндөгүчө:

.SELECT <область>

Мында область – жумушчу областтынын аты, же анын идентификациялаган номери. Бул команда аткарылгандан кийин көрсөтүлгөн область учурдагы область болуп калат. Жумушчу областтарды 1-10го чейинки цифралар, A-J тамгалары, файлдын аты менен көрсөтүүгө болот. Ошондой эле жумушчу областка псевдоним – ALIAS ат ыйгарып алабыз. Жумушчу областка ALIAS атты ыйгаруу USE командасы менен файлды ачып жатканда берилет.

Мисалы: SELE 3

USE ANK alias f1 файлды ачуу жана f1 псевдонимин ыйгаруу. Активдүү эмес жумушчу областтын жазууларынын талааларына кайрылуу үчүн талаанын атынын алдына ALIAS – атты жана '->' символдорун көрсөтүү керек.

f1> fm; f1> tj

Маалыматтар файлдарын эки же андан көп жумушчу областтарда ачкандан кийин, алардын арасында байланыш тургузууга болот. Ал үчүн SET RELATION командасын колдонобуз. Анын форматы төмөндөгүчө: SET RELATION <жол көрсөтүүчү> INTO <псевдоним>.

Бул команда активдүү жумушчу областтагы файлынын аты менен ALIAS аныкталган, активдүү эмес жумушчу областтагы файлдардын ортосунда жол көрсөтүүсү боюнча байланыш орнотот. Байланыш үчүн <жол көрсөтүүчү> төмөнкү түрдө боло алат:

1. талаанын аты, талаалардын аттарынын группасынын жолчосу;
2. RECNO () функциясы;

3. арифметикалык туюнтма

Файлдарды талаа боюнча байланыш түзгөндө активдүү эмес файл менен ушул талаа боюнча индекстелген болушу керек. RECNO 0 функциясы жана арифметикалык туюнтмалар активдүү жана активдүү эмес файлдардагы жазууларды номерлери боюнча байланыштырат. Бул учурларда индекстөө талап кылынбайт.

Кээ бир учурларда жумушчу областтарда ачылган файлдарды бириктирип, бир убактылуу файл катары кароого туура келет. Ал убактылуу файлдарды иштетип бүткөндөн кийин ERASE же DELETE FILE командалары менен өчүрүп койсо болот.

Маалыматтардын эки файлынын жазууларын жана талааларын бириктирип, жаңы файлды түзүү JOIN командасы менен аткарылат

Команданын форматы төмөндөгүчө:

JOIN WITH <псевдоним> TO <файл> FOR <шарт> FIELDS <талаалардын тизмеси>.

Команда аткарылганда жаңы түзүлүүчү файл <талаалардын тизмесинде> көрсөтүлгөн бириктирилүүчү файлдардын талааларын камтыйт. Эгерде FIELDS параметри көрсөтүлбөсө, анда жаңы түзүлгөн файлдын талаалары активдүү файлдын, андан кийин активдүү эмес файлдын талааларынан түзүлөт. Агтары дал келген талаалар үчүн активдүү файлдын талаалары камтылат.

Бул команда төмөндөгүчө иштейт:

1. жазуунун көрсөткүчү активдүү файлдын 1-жазуусуна тургузулат;

2. 2-файлдын бардык жазуулары каралат.

Эгерде кезектеги жазуу FOR <шарттын> канааттандырса, анда ал жаңы файлга киргизилет:

3. жазуунун көрсөткүчү активдүү файлдын кийинки жазуусуна тургузулат;

4. активдүү файлдын аягына чыкканга чейин 2- жана 3- пункттар кайталанат.

Файлдарды бириктирүүгө мисал кароо үчүн жаңы маселелерди коелу. Бизге окутуучулар жана кызматчылар жөнүндөгү маалыматтар берилсин.

Биринчиси үчүн OKUT .dbf, экинчиси учун KYZ .dbf файлдарын түзөлү. OKUT .dbf файлы төмөндөгү талаалардан турсун:

FAA KYZ KAFEGRA

Ал эми KYZM .dbf файлы төмөндөгү талаалардан турсун:

KYZ OKLAD

Бул эки файлды KYZ талаасы боюнча бириктирели. Программа төмөнкүдөй болот.

*Файлдарды бириктирүү

Set talk off

Sele 1

Use kyz alia f2

Sele 2

Use okut

join with f2 tj anketa FOR KYZ \$ f2 >KYZ FIEL FAA;

KYZ, f2>oklad, kafedra

Close DATA

Set talk on

Retu

Жаңы түзүлгөн ANKETA .dbg файлынын структурасы төмөндөгүчө: FIO KYZ OKLAD KAFEDRA

§ 16 LOWER, UPPER, LIRIM, RTRIM, REPLICATEL,
STIFF

Foxbase тилинде символдук маанилерди өзгөртүп түзүү амалдардын жана сандардык функциялардын жардамы менен иш жүзүнө ашырат. Эки символдук жолчолор менен амалдарды аткаруунун негизгиси ? болуп саналат. Бул амалдын эки жолчо үчүн жыйынтыгы болуп, ал жолчолор бириктирилип жазылган жаңы жолчо болуп эсептелет. Ал амал (+) операторунун жардамы менен аткарылат. Мисалы, '125'+ '569' жыйынтыгы '125569'.

Бул амалдын даагы бир варианты бар. Ал (-) оператору менен аткарылат. Мында биринчи операнддын аягындагы пробелдер, ал эми экинчи операнддын башталышындагы пробелдер алынып ташталат. Мисалы, '628'+ ' 561' амалынын жыйынтыгы '628 561', ал эми '628'- ' 561' түрүндөгү амалдын жыйынтыгы '628561' болот.

Катыш амалдарын аткаруу үчүн (\$) оператору колдонулат. Эгерде экинчи операнд биринчи операнддын бөлүгү болсо, анда бул амалдын жыйынтыгы логикалык мааниси .T., ал эми бөлүгү болсо, анда жыйынтыгы логикалык, мисалы .F. болот. Мисалы, амал аткарылгандан кийин

X= 'Алымкулов' \$ 'M'

X өзгөрүлмөсүнүн мааниси .T. болот, анкени, 'M' символу биринчи операнддын бөлүгү .

Эки жолчонун бири-бири менен (=) оператору менен аткарылат. Бул амал SET EXACT ON/OFF .

Так салыштыруу режиминде биринчи жолчонун экинчи жолчо менен толук дал келиши текшерилет. Мында экинчи жолчонун аягындагы пробелдер эсине алынбайт.

Эгерде 1-? баштап <1-жолчо> <2-жолчо> толугу менен дал келсе, анда салыштыруунун жыйынтыгы .T. болот. <2-жолчонун> узундугу <1-жолчонун> узундугунан ашпоосу керек.

Мисалы: «ABC»+»ABC» жыйынтыгы .F.

«ABC»=»ABC» «ABCDE» =»ABC» .T.

«ABC»=»ABC».F.

Жолчолорду чоң-кичине салыштыруу да ушундай эле болот.

Символдук жолчо менен иштөө үчүн колдонуучунун карамагында структуралык файлдардын группасы бар. Алардын жардамы менен жолчону өзгөртүп түзүүгө, камтылуучу жолчону бөлүп алууга, башкы типке өзгөртүп түзүүгө болот .

Жолчолду өзгөртүп түзүүгө. Бул группага LOWER (), UPPER (), LTRIM (), RTRIM (), REPLICATE (), STIFF (), [DESCEND (), ALLTRIM], LOWER () функциясы символдук жолчодо баш тамгалар менен жазылган тамгаларды кичине тамгаларга өтүп түзөт. Файлга кайрылуу форматы:

LOWER(<жолчо>)

Жолчодогу башка символдор өзгөрбөйт. Мисалы, LOWER(МОЭ-1-99- группа) жыйынтыгы «МОЭ-1-99-группа» болот.

UPPER() функциясы берилген жолчодогу кичине тамгаларды баш тамгаларга өзгөртөт. Кайрылуу форматы:

UPPER(<жолчо>)

Жолчо башка символго өзгөрүп калат.

LTRIM() функциясы жолчонун башындагы пробелдерди четтетет. Жолчо бош болсо же жалаң гана пробелден турса, анда бул функция бош жолчону ыйгарат. Кайрылуу форматы:

LTRIM (<жолчо>)

Бул функция менен сандарды өзгөртүп түзүүнүн жыйынтыгында алынган пробелден четтетүү үчүн колдонулат.

Мисалы: STR функциясы менен 20 санынын узундугун көрсөтпөй өзгөртүп түзсөк, анда жыйынтыгында «20» жолчосу болот. Ал эми LTRM (STR(20)) нын жыйынтыгы «20» болот.

RTRIM() функциясы маалыматтар базасы фиксирленген узундугун талааларды пробел четтетүүдө колдонууга ыңгайлуу.

STIFF() функциясы баштапкы жолчонун бөлүгүн четтетүү же алмаштыруу үчүн кызмат кылат. Функциянын форматы:

STIFF (<жолчо>, <баштапкы>, <узундук>, <коюу>).

Мында <жолчо>-баштапкы жолчо

<баштапкы>- баштапкы жолчодон четтетилүүчү фрагменттин башталышы же жаңы фрагмент коюлуучу орунду аныктаган позиция.

<узундук>-четтетүүчү фрагменттин узундугу

<коюу>-коюлуучу символдук жолчо.

Функция башталган жолчодон <баштапкы> позициядан баштап жана <узундук> узундугуна ээ болгон фрагменттин ордуна <коюу> жолчосун алмаштырып, жаны жолчого түзөт. Эгерде <узундук> параметринин мааниси 0 гө барабар болсо, анда <коюу> жолчосун көрсөтүлгөн позициядан баштап коет. Мисалы, STIFF («Асанов» 5,0, »бек») жыйынтыгы Асанбеков

Эгерде узундук параметри баштапкы жолчонун символдорунун санынан ашык болсо, анда жолчодогу бар символдор гана өчүрүлөт.

Мисалы, STIFF («Группа боюнча тизме», 8,20, «А 5-курс») жыйынтыгы «Группа А 5-курс» болот.

REPLICATEL() функциясы баштапкы жолчону кайталоо менен жаңы жолчону түзөт. Функциянын форматы:

REPLICATEL (<жолчо>, <кайталоо>)

Мында <жолчо>-баштапкы жолчо

<кайталоо>-баштапкы жолчодогу кайталоонун саны.

Жыйынтык жолчонун узундугу 65535 символдон ашпоосу керек. Эгер <кайталоо> параметри 0го барабар болсо, анда функция бош жолчону берет.

Мисалы, REPLICATEL («!», 6) жыйынтыгы !!!!! болот

§ 17. REPLACE командасы

Маалыматтарды иштетүүдө жыйынтыктар өзгөрүлмөлөрдүн маанилери катарында сакталып, аларды ар кандай жолдомолорду жана отчетторду чыгарыш үчүн колдонууга болот. Практикалык маселелерде, көпчүлүк учурларда маалыматтардын базасынын жазылыштарынын талааларынын маанилерин өзгөртүүгө туура келет. Төмөнкүдөй амалдарды аткаруу менен өзгөртүүгө болот: эсептөөлөрдү жүргүзүп, мурда бош турган талааларга ал маанилерди ыйгаруу ж.б.

Маалыматтардын базасынын жазылыштарынын талааларын иштеп чыгуу менен талааларга жаңы маанилерди ыйгарууну REPLACE (алмаштыруу) командасы аткара алат.

REPLACE командасынын жалпы форматы төмөнкү көрүнүштө болот:

REPLACE [<чектегич>] <талаанын аты> WITH<туюнтма> [<талаанын аты> WITH <туюнтма>...].

Мында <чектегич> өзгөртүү киргизиле турган жазылышты көрсөтө турган опция: бул N- номерлүү жазылыш (RECORD), учурдагы жазылыштан баштап кийинки жазылыш, N жазылыш (NEXT N) же бардык жазылыштар (ALL); < талаанын аты > – талаанын мааниси

өзгөрүлө турган талаа; <туянтма> – алдын ала эсептелип, ал маани өзгөрүлүүчү талаага берилет: туянтманын тиби жана анын маанисинин узундугу маалыматтардын базасынын структурасында көрсөтүлгөн өзгөрүлүүчү талаалардын типтерине жана узундуктарына тиешелеш болуш керек.

REPLACE командасы FOR < шарт > опциясы менен толукталышы да мүмкүн.

Эгерде FOR опциясында көрсөтүлгөн логикалык шарттын мааниси .T. (чын) болсо, анда учурда REPLACE командасы менен өзгөртүү бардык жазылыштарга колдонулат.

Эгер FOR опциясы да, <чектелиш> опциясы да командада көрсөтүлбөсө, анда REPLACE командасы учурдагы жазылышка гана өзгөртүү киргизе алат. REPLACE командасында бир нече <талаанын аты> < туянтма > түгөйлөрү көрсөтүлүшү мүмкүн. Бул болсо бир нече талаалардын маанилерин өзгөртүүгө мүмкүнчүлүк түзөт. Анда булардын команда менен кошо жалпы саны чектелген болушу керек (254 символдон ашык болбошу керек).

REPLACE командасынын колдонушунун ААЕВ-«Айлык акыны эсептөө ведомосту» маалыматтардын базасын иштетүү үчүн демонстрациялайбыз. Маалыматтардын базасы ААЕВ нын структурасы 1- сүрөттө келтирилген (структура түзүлгөн деп эсептейбиз):

```
.DISP STRU
structure fir database: AAЕV.DBF
number of data records:
Data of last update          01/01/93
```

Field	Field name	TYPE	Width	Dec
1	TN	Numeric	6	
2	FA	Character	12	
3	BS	Numeric	6	2
4	EA	Numeric	6	2
5	SI	Numeric	6	2
6	BK	Numeric	6	2
7	AV	Numeric	6	2
8	KN	Numeric	6	2
9	AS	Numeric	6	2
** Total **			60	

1-сүрөт.

Сүрөттөгү талаалардын аталыштары төмөнкүчө: TN-«табелдик номери», FA-«фамилиясы, аты, атасынын аты», BS-«бардык сумма», EA-«Эмгек акы», SI-«сыйлык», BK-«бардык кармалганы», AV-«аванс», KN-«киреше налогу», AS-«алына турган сумма» төмөнкү формулалар менен эсептелет:

$$BS=EA+SI$$

$$KN=100*0.08+(BS-100)*0.13$$

$$BK=AV+KN$$

$$AS=BS-BK$$

Киреше налогун эсептөө формуласын 1993-жылга ~~чейинки~~ мезгилдер үчүн түшүндүрөбүз. Ал 100 сомго чейинкиси 8%, 100 сомдон ашканы 13%ти түзгөн.

Мурда түзүлгөн AAEV маалыматтар базасын ачып, LIST командасы менен экранга чыгаралы. (2-сүрөт):

```
USE AAEV
LIST
```

Records	TN	FA	BS	EA	SI	BK	AV	KN	AS
1	11	Арзиев ы.	0.00	200.00	30.00	0.00	30.00	0,00	0.00
2	12	Карыбеков М	0.00	187.00	40.00	0.00	75.00	0,00	0.00
3	13	Кенжебаев К.	0.00	245.00	75.00	0.00	120.00	0.00	0.00
4	14	Айдаров А.	0.00	145.00	50.00	0.00	60.00	0.00	0.00
5	15	Рысбаев К.	0.00	280.00	50.00	0.00	120.00	0.00	0.00

2-сүрөт.

3-сүрөттө BC "бардык сумма" талаасынын маанилерин эсептөөнү камсыз кылуучу REPLACE командасы киргизилген:

```
USE AARV
REPLACE ALL BS WITH EA+SI
5 Records replased
LIST FA, BS, EA, SI
```

Record #	FA	BS	EA	SI
1	Арзиев Б.	250.00	200.00	50.00
2	Карыбеков М.	227.00	187.00	40.00
3	Кенжебаев К.	320.00	245.00	75.00
4	Айдаров А.	195.00	145.00	50.00
5	Рысбаев К.	330.00	280.00	50.00

3-сүрөт..

4-сүрөттө толбогон бардык талаалардын маанилерин эсептөөнү камсыз кылуучу REPLACE командасы киргизилген:

```

.REPLACE ALL KN WITH (100*0.08)+(BS-100)*0.13
    5 Records replaced
.REPLACE ALL BK WITH KN+AV
    5 Records replaced
.REPLACED FLL AS WITH BS-BK
    5 Records replaced
.LIST
Record # FA      BS      EA      SI      BK      AV      KN      AS
  1  Арзиев Ы.  250.00  200.00  50.00  107.00  80.00  27.50  142.00
  2  Карыбеков М. 227.00  187.00  40.00  99.51  75.00  24.51  127.40
  3  Кенжебаев К. 320.00  245.00  75.00  56.60  120.00  36.60  163.40
  4  Айдаров А. 1195.00 145.00  50.00  80.35  60.00  20.35  114.65
  5  Рысбаев А.  330.00  80.00  50.00  57.90  120.00  37.90  172.10
    
```

4-сүрөт.

Келтирилген мисалдарда REPLACE командасынын ар бири AAEV маалыматтардын базасынын жазылыштарынын бир гана талаасынын маанисин эсептейт жана аны алмаштырат. Бир эле REPLACE командасы менен бардык эсептөөлөрдү жазууга болот (3-сүрөт). Бул учурда иштеп чыгуу эффективдүү болот. Себеби, мурунку мисалдарда REPLACE командасы канча болсо, ошончо жолу AAEV маалыматтардын базасын карап чыгат. Ал эми 3-сүрөттө REPLACE командасы маалыматтардын базасын бир гана карап чыгуу менен ишти бүтүрөт (5-сүрөт).

```

.REPLACE ALL BS WITH EA+SI, ;
    KN WITH (100*0.00)+(BS-100)*0.13, ;
    BK WITH AV+KN, ;
    AS WITH BS-BK
5 records reolaced
.LIST
Rec. # TN FA      BS      EA      SI      BK      AV      KN      AS
  1 11 Арзиев Ы.  250.00  200.00  50.00  107.00  80.00  127.50  142.50
  2 12 Карыбеков М. 227.00  187.00  40.00  99.51  75.00  24.51  127.40
  3 13 Кенжебаев К. 320.00  245.00  75.00  156.60  120.00  36.60  263.40
  4 14 Айдаров А.  195.00  145.00  50.00  80.35  60.00  20.35  114.65
  5 15 Рысбаев К.  330.00  280.00  50.00  157.90  120.00  37.90  172.10
    
```

5-сүрөт.

§ 18. COUNT, SUM, AVERAGE КОМАНДАЛАРЫ

Математикалык эсептөөлөрдүн каражаттары катарында AVERAGE (орто), SUM (сумма), жана COUNT (саноо) командалары да эсептелишет. Бул командалар FOR жана <чектегич> операцияларына ээ. Бул чектелиштер менен бирдикте жогорку командалар маалыматтардын базасынын жазылыштарынын төмөнкүлөрүн эсептей алышат:

- сандык талаалардын орточо маанисин (AVERAGE);
- сандык талаалардын маанилеринин суммасын (SUM);
- берилген шартты канааттандырган жазылыштардын саны (COUNT).

AVERAGE командасынын жалпы форматы төмөнкүчө болот;

~~AVERAGE~~ [<талаалардын аттарынын тизмеси >]
[<чектегичи >]

[FOR<шарт>]

[TO<өзгөрүлмөлөрдүн тизмеси>]

Мында, <талаалардын аттарынын тизмеси>, орточо маанилери эсептеле турган талаалардын тизмеси; <өзгөрүлмөлөрдүн тизмеси>-эсептелген орточо маанидерди ыйгара турган өзгөрүлмөлөрдүн аттарынын тизмеси. Тизмедеги өзөрүлмөлөрдүн саны, <талаалардын санына тиешелеш болуусу зарыл >. AVERAGE командасынын бардык опцияларынын жазылышы шарт эмес экендиги команданын жалпы форматынан көрүнүп турат.

AVERAGE

Командасы бардык сандык талаалардын орточо маанилерин эсептеп экранга чыгарат. Бул командага чейин AAЕV файлы ачылган болуш керек (1-сүрөт)

```
.USE AAЕV
```

```
.AVERAGE
```

```
5 records averaged
```

TN	BS	EA	SI	BK	AV	KN	AB
13	264.00	211.40	53.00	120.37	91.00	29.37	144.03

1-сүрөт.

2-сүрөттөгү командалар боюнча физика предмети жана группадагы жетишүүнүн бардык баалары боюнча орточо маанилери эсептелип, экранга чыгарылган (учурдагы маалыматтар JET_OR базасынан алынган . айрым предметтер боюнча OR талаасында жазылган):

```
.USE JET_OR
.LIST FM, FIZ, OR
```

Record#	FM	FIZ	OR
1.	Адельев	5	4.5
2.	Исраилов	4	4.66
3.	Айдаров	4	3.38
4.	Бакиров	5	4.66
5.	Рысбаев	4	4.16
6.	Сартбаева	5	5.00

```
. AVERAGE FIZ, OR

6 Records averaged
  FIZ      OR
  4        4.31
```

2-сүрөт.

2-сүрөттө физика предметинин орточо баллы 4,32 ге барабар болуп калды.

Мындай жыйынтык окуу жайынын жетекчисин кызыктырбасы белгилүү. Ошондуктан, маанини үтүрдөн кийинки эки белгиге чейинки тактыкта алууга болот. (3-сүрөт):

```
.USE JET_OR
.AVERAGE FIZ TO OBF
  6 Records averaged
  FIZ
  4
. ? STR(OBF,4,2)
  4.05
```

3-сүрөт.

SUM командасы <чектегич> жана FOR опциялары менен бирге жазылыштардын сандык талааларынын орточо маанисин эсептөө үчүн колдонулат.

SUM командасынын жалпы форматы төмөнкө көрүнүштө болот:

```
.SUM [<чектегич>] [<талаалардын аттарынын тизмеси>]  
[TO<өзгөрүлмөлүрдүн тизмеси>]  
[FOR<шарт>]
```

4-сүрөттө AAEV маалыматтардын базасындагы сандык талаалардын суммасын SUM командасынын жардамы менен эсептегендигин көрүүгө болот.

```
.USE AAEV  
.SUM  
5 records summed  
TN    BS    EA    SI    BK    AV    KN    AS  
65    1322.00  1057.00  265.00  601.07  455.00  146.86  720.14  
.SUM BS, BK,AS  
5 records summed  
BS    BK    AS  
1322.00  601.86  720.14  
.SUM SI TO BSI  
5 records summed  
SI  
265.00  
.?BSI  
265.00
```

4-сүрөт.

COUNT-командасы маалыматтардын базасындагы берилген шартты канааттандырган жазылыштардын санын аныктайт. Ал шарт FOR опциясында аныкталган. Зарылчылык болгондо алынган маанини кандайдыр бир өзгөрүлмөгө ыйгарып коюу керек. Бул команда төмөнкүдөй форматка ээ:

```
.COUNT[<чектегич>] [FOR<шарт>] [TO<өзгөрүлмө>]  
.COUNT
```

командасы аткарылганда жыйынтык маалыматтардын базасындагы жазылыштардын саны болот.

5-сүрөттө балдары 4.3төн жогору болгон, андан кийин балдары 4.0гө барабар же андан чоң болгон окуучулардын санын COUNT командасы менен аткарылгандыгын байкоого болот:

```
.USE JET_OR  
.COUNT FOR OR>4.3  
3 Records  
.COUNT FOR OR>=4.0 TO NOR  
5 records  
.? NOR  
5
```

5-сүрөт.

6-сүрөттө COUNT жана SUM командаларынын жардамы менен окуучулардын санын (бул маани OSA өзгөрүлмөсүнө берилсин) жана алардын математикадан алган бааларынын суммасын (SMAT) экранга чыгаруу каралган:

```
.USE JET_OR  
.COUNT TO OSA  
6 records  
.SUM MAT TO SMAT  
6 records  
MAT  
28  
.?SMATYOSA  
4.66
```

6-сүрөт.

Бул сүрөттөгү SMATYOSA окуучулардын орточо баасын берерин көрүүгө болот.

Тигил же бул окуучунун бардык предметтер боюнча орточо баллын JET маалыматтар базасын ачуу менен төмөнкү команда боюнча да табууга болот:

```
.AVERAGE (FIZ+MAT+INF+KT) Y 5 TO OM
```

7- сүрөттө бул команданын аткарылышынын жыйынтыгы келтирилген:

```
.USE JET  
.AVERAGE (FIZ+MAT+INF+KT) Y 5 TO OM  
6 records averaged  
(FIZ+MAT+INF+KT) Y 5 TO OM  
.?OM  
4.31
```

6-сүрөт.

§ 19. Маалыматтарды печаттоо

Маалыматтардын базасын эксплуатациялоодо издөө жана иштетүү операциялары аткарыларын көрдүк. Аларды жолдомо жана отчет же башка документтер катары түзүп, печатка берүү максатка ылайыктуу. Экранга даярдалган информацияларды (маалыматтардын базасына тиешелүү) экранга чыгаруу үчүн Print Screen командасын клавиатурадан алуу керек. Натыйжада экрандагы жолчолор ирети менен принтерге салынган кагазга түшө берет. Печатка чыгарылуучу жолчолорду кагаздын сол жактан оң жагын көздөй каалаган позицияга жылдырып алууга болот (бир позиция -1 символ). Ал үчүн төмөнкү команданы кийирүү зарыл.

.SET MARGIN TO < n >

Мында <n> - жылдырылуучу позициялардын саны. Бул команда кийинки печатка чыгаруучу командалар үчүн сакталат. Бул режимди өзгөртүү үчүн кайрадан SET MARGIN TO командасын кийирүү керек.

Печатты уюштуруудан мурда дисплейдин экранын тазалоо керек. Ал үчүн CLEAR () командасын берүү керек:

.CLEAR

Экрандагы информацияны кагазга түшүрүүнүн дагы бир жолу бар.

Ал төмөнкү команда боюнча ишке ашат.

.SET PRINT ON

Бул команданы киргизгенден кийин колдонуучу аркылуу берилген командалар, алардын аткарган жыйынтыктары, системанын билдирүүлөрү ж.б. жалаң гана экранга чыкпастан печаттоочу түзүлүшкө да чыгат. Печатка чыгарууну кайталоону токтотуу үчүн төмөнкү команданы киргизүү керек:

.SET PRINT OFF

PRINT ON режимин FoxBase менен иштөөнү бардык сеансында же колдонуучу ишин бүтүргөнгө чейин колдонушу мүмкүн.

Жогоруда көрсөтүлгөн жөнөкөй жол менен алынган документтер жумушчу гана мүнөзгө ээ жана алар чектелген гана колдонуштарга ээ.

Официалдуу документтерди печатка чыгаруу [8] эмгекте толугу менен каралган.

Көнүгүүлөр.

1. «Студенттердин» маалыматтардын базасынын структурасын түзгүлө жана экранга чыгаргыла.
2. Кафедранын мугалимдери жөнүндөгү маалыматтардын структурасын түзгүлө.
3. «Контракттык сумма» маалыматтардын базалык файлын түзгүлө.
4. Кафедранын мугалимдери жөнүндөгү маалыматтардын базасын түзгүлө;
5. EDIT режиминде OM файлын пайдаланып, 7 жана 8-каалаган жазылыштарды жазып, жаңы файлды OMM деп атагыла. Жыйынтыгын экранга чыгаргыла.
6. CHANGE командасын пайдаланып, JET маалыматтардын базасындагы бардык окуучулардын математикадан жаңы бааларын киргизгиле.
7. APPEND командасын пайдаланып, маалыматтарды OMM маалыматтардын базасына ANK жана JET базаларына которгула.
8. LOCATE командасын пайдаланып OM маалыматтардын базасынан Сартбаева жөнүндөгү маалыматы бар жазылышты экранга чыгаргыла.
9. AAEV маалыматтардын базасын пайдаланып, айлыкты 590 сомдон кем албаган кызматчылардын фамилияларын экранга чыгаргыла.
10. LOCATE командасын пайдаланып, JET маалыматтардын базасына эң жакшы окуган окуучулар жөнүндөгү билдинүүнү экранга чыгаруу менен бирге аны печатка чыгаргыла.
11. LIST, DISP командаларын FOR опциясы менен пайдаланып, төмөнкү фамилияларды печатка чыгаргыла:
 - а) жакшы окугандар;
 - б) үчкө окугандар.
12. JET_OR маалыматтардын базасынын бардык жазылыштарындагы OR талаасынын маанилерин толтургула.
13. ANK маалыматтардын базасы үчүн окуучулардын бардык предметтер боюнча орточо маанилерин аныктагыла.

1. Кергаль И. Методы программирования на Бейсике / Перевод с французского Мир. 1991
2. Есяян А.Р. и др. Информатика. –М:Просвещение. 1991
3. Бленд Г. Основы программирования на языке Бейсик в стандарте MSX / Перевод с английского-М:Финансы и статистика. 1989
4. Кетков Н.А. Диалог на языке Бейсик для мини и макро-ЭВМ.-М. Наука, 1988
5. Макдона Р. Основы микрокомпьютерных вычислений./ Перевод с английского.-М. : 1989,
6. Радер Дж. ,Милсап К. Бейсик для персонального компьютера фирмы IBM / Перевод с английского.-М :Радио и связь, 1991.
7. Куправа Т.А Создание программирование баз данных средствами СУБД.-М.: Мир. 1991
8. Савельев А.Я. и др. Создание и использование баз данных. Т.3.-М.: Высшая школа, 1991.

Мазмуну

Кириш сөз	3
I Глава	
Маалыматтар базасын түзүү	
§ 1. Негизги түшүнүктөр	4
§ 2. Маалыматтар базасын түзүү	5
§ 3. Маалыматтар .dbf-файлынын алгачкы ирет толтуруу	10
§ 4. Маалыматтарды окуу	11
§ 5. Жазылыштарды киргизүү жана чыгаруу	16
§ 6. Структураны модификациялоо	22
§ 7. Сорттоо	31
§ 8. Маалыматтарды издөө	35
§ 9. Маалыматтарды тандоо	41
II Глава	
Маалыматтарды иштеп чыгуу	
§ 10. Турактуулар өзгөрүлмөлөр жана туюнтмалар	43
§ 11. Кээ бир SET командалар	47
§ 12. Негизги стандарттык функциялар	49
§ 13. Программа түзүүгө мисалдар	52
III Глава	
FOX BASE системасында сызыктуу тармактануучу, циклдик программаларды түзүү	54
§ 14. Башкаруучу структуралар	
§ 15. Бир нече маалыматтар базасы менен бир учурда иштөө	62
§ 16. LOWER, UPPER, LTRIM, RTRIM, REPLICATE, STIFF	65
§ 17. REPLACE командасы	68
§ 18. COUNT, SUM, AVERAGE КОМАНДАЛАРЫ	72
§ 19. Маалыматтарды печаттоо	76
Көнүгүүлөр.	77
Адабияттар	78

ул. Перисомтова, 12-107.

Тел: 23-77-62. Факс: 219, 238, 152, 172, 137, 208,

Н.А.



АБЖАПАРОВА Ү.А., РАЕВ К.Т., РАЕВ А.К.

FoxBase III Plus системасында программалоо

Редактор - Абжапарова Ү.А.
Тех. редактор - Маматалиев М.О.
Басмага даярдаган - Тойчубаев А.А.

Басууга кол коюлду 16.05.2000-ж. Форматы 60x84^{1/16}. Көлөмү 5
Тиражы 500. Буюртма № 2262. Офсеттик ыкма менен басылды.

Ош облустук басмаканасы. Ош ш. Курманжан датка — 209.

БИБЛИОТЕКА
Одесского государственного
университета
ИЗДАНИЕ: 30=00

30c



932819