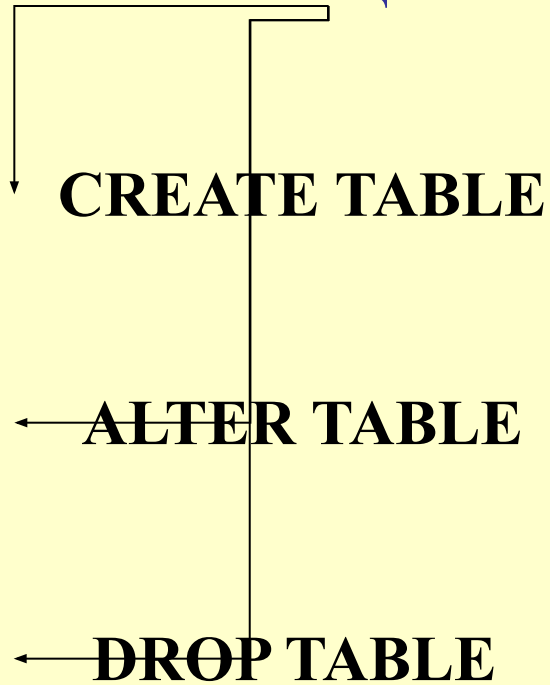


Лекция 17

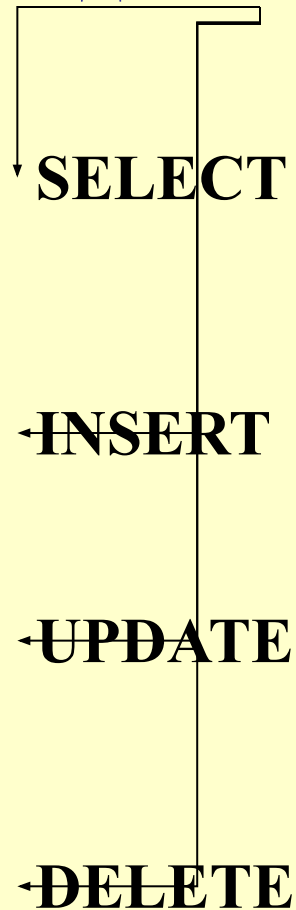
Построение баз данных с помощью SQL.

Манипулирование данными в SQL

Управление таблицами



Управление данными



Команда создания таблицы – CREATE TABLE

CREATE TABLE имя_таблицы
({<определение_столбца>|
<определение_ограничения_таблицы>}
[,...,{<определение_столбца>|
<определение_ограничения_таблицы >}])

Определение_столбца:

<Имя_столбца> <тип_данных>
[<ограничение_столбца>] [,...,<ограничение_столбца>]

Ограничение столбца

<ограничение_столбца> ::=

[CONSTRAINT <имя_ограничения >]

{ [DEFAULT <выражение>]

| [NULL | NOT NULL]

| [PRIMARY KEY | UNIQUE]

| [FOREIGN KEY

REFERENCES

 <имя_главной_таблицы>[(<имя_столбца> [, ..., n])]

[ON DELETE { CASCADE | NO ACTION }]

[ON UPDATE { CASCADE | NO ACTION }]

]

| [CHECK (<логическое_выражение>)]

}

Ограничения на уровне таблицы

```
<ограничение_таблицы> ::=  
[ CONSTRAINT <имя_ограничения>  
{ [ { PRIMARY KEY | UNIQUE }  
{(<имя_колонки> [ASC | DESC] [,...,n] )}  
| FOREIGN KEY  
[ ( <имя_колонки>[,..., n] ) ]  
REFERENCES <внешняя_таблица> [  
    (<имя_колонки_внешней_таблицы> [, ..., n] ) ]  
[ ON DELETE { CASCADE | NO ACTION } ]  
[ ON UPDATE { CASCADE | NO ACTION } ]  
| CHECK (<логическое_выражение> )  
}
```

Создание таблицы «Студенты» (БД «Сессия»)

```
CREATE TABLE Студенты  
(ID_Студент INTEGER NOT NULL,  
Фамилия CHAR(30) NOT NULL,  
Имя CHAR(15) NOT NULL,  
Отчество CHAR(20) NOT NULL,  
Номер_группы INTEGER NOT NULL,  
Адрес CHAR(30),  
Телефон CHAR(8),  
PRIMARY KEY (ID_Студент))
```

Создание таблицы «Дисциплины» (БД «Сессия»)

```
CREATE TABLE Дисциплины  
(ID_Дисциплина INTEGER NOT NULL,  
Наименование VARCHAR(40) NOT  
NULL,  
PRIMARY KEY (ID_Дисциплина),  
UNIQUE (Наименование))
```

Создание таблицы «Учебный план» (БД «Сессия»)

```
CREATE TABLE    Учебный_план
(ID_План          INTEGER NOT NULL,
ID_Дисциплина    INTEGER NOT NULL,
Семестр          INTEGER NOT NULL,
Количество_часов INTEGER,
ID_Преподаватель INTEGER,
PRIMARY KEY     (ID_План),
CHECK          ((Семестр >= 1) OR
(Семестр <= 10)))
```


Создание таблицы «Сводная ведомость» (БД «Сессия»)

```
CREATE TABLE Сводная_ведомость  
(ID_Студент  INTEGER NOT NULL,  
ID_План      INTEGER NOT NULL,  
Оценка      INTEGER NOT NULL,  
Дата_сдачи  DATETIME NOT NULL,  
PRIMARY KEY (ID_Студент, ID_Дисциплина),  
CHECK      ((Оценка >= 0) OR  
              (Оценка <= 5)))
```

Создание таблицы «Сводная ведомость» (БД «Сессия»)

```
CREATE TABLE Кадровый_состав  
(ID_Преподаватель INTEGER NOT NULL,  
Фамилия CHAR(30) NOT NULL,  
Имя CHAR(15) NOT NULL,  
Отчество CHAR(20) NOT NULL,  
Должность CHAR(20) NOT NULL,  
Кафедра CHAR(3) NOT NULL,  
Адрес CHAR(30),  
Телефон CHAR(8),  
PRIMARY KEY (ID_Преподаватель))
```

Команда ALTER TABLE

- добавить в таблицу определение нового столбца;
- удалить столбец из таблицы;
- изменить значение по умолчанию для какого-либо столбца;
- добавить или удалить первичный ключ таблицы;
- добавить или удалить внешний ключ таблицы;
- добавить или удалить условие уникальности;
- добавить или удалить условие на значение.

ALTER TABLE <имя_таблицы>

[**ALTER COLUMN** <имя_столбца> [**SET DEFAULT** <выражение>]]
[**DROP DEFAULT**]]

[[**ADD** <определение_столбца>]

[[**DROP COLUMN** <имя_столбца> [**CASCADE**]][**RESTRICT**]]

[[**ADD**

[<определение_первичного_ключа>]][<определение_внешнего_ключа>]]

[<условие_уникальности>]][<условие_на_значение>]]

[[**DROP CONSTRAINT** <имя_ограничения> [**CASCADE**]][**RESTRICT**]]

Добавление столбца

```
ALTER TABLE Студенты  
ADD Год_поступления INTEGER NOT NULL  
DEFAULT YEAR(GETDATE())
```

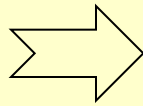
Модификация столбца

```
ALTER TABLE Студенты  
ALTER COLUMN Номер_группы CHAR(6) NOT  
NULL
```

Удаление столбца

```
ALTER TABLE Студенты  
DROP COLUMN  
Год_поступления
```

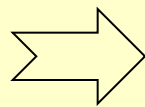
**Добавление
внешних ключей в
таблицу
«Учебный_план»**



```
ALTER TABLE Учебный_план  
ADD CONSTRAINT  
FK_Дисциплина  
FOREIGN KEY (ID_Дисциплина)  
REFERENCES Дисциплины
```

```
ALTER TABLE Учебный_план  
ADD CONSTRAINT  
FK_Кадровый_состав  
FOREIGN KEY  
(ID_Преподаватель)  
REFERENCES Кадровый_состав  
ALTER TABLE
```

**Добавление
внешних ключей в
таблицу
«Сводная_ведомость»**



```
Сводная_ведомость  
ADD CONSTRAINT FK_Студент  
FOREIGN KEY (ID_Студент)  
REFERENCES Студенты
```

```
ALTER TABLE  
Сводная_ведомость  
ADD CONSTRAINT FK_План  
FOREIGN KEY (ID_План)  
REFERENCES Учебный_план
```

Удаление ограничений

```
ALTER TABLE Учебный_план  
DROP CONSTRAINT
```

FK Дисциплина

Удаление внешнего ключа

```
ALTER TABLE Студент  
ADD CONSTRAINT DEF_Номер_группы  
DEFAULT 1
```

FOR Номер_группы

```
ALTER TABLE Студент
```

```
DROP CONSTRAINT
```

```
DEF_Номер_группы
```

Удаление значения по умолчанию

Удаление таблицы

```
DROP TABLE <имя_таблицы>
```

Извлечение данных – команда SELECT

```
SELECT <Список_выбора>  
[ INTO <Новая_таблица> ]  
FROM <Исходная_таблица>  
[ WHERE <Условие_отбора> ]  
[ GROUP BY <Ключи_группировки> ]  
[ HAVING <Условие_отбора> ]  
[ ORDER BY <Ключи_сортировки> [ ASC | DESC ] ]
```

SELECT	INTO
FROM	WHERE
GROUP BY	HAVING
UNION	ORDER BY
COMPUTE	FOR
OPTION	

Раздел SELECT -

обязательный раздел, который должен указываться в любом запросе: задает набор столбцов, возвращаемых после выполнения запроса, т.е. внешний вид результата

```
SELECT [ ALL | DISTINCT ]  
[ TOP n [ PERCENT ] [ WITH TIES ] ]  
<Список_выбора>
```


Ключевые слова ALL и DISTINCT

SELECT ALL Семестр,
Отчетность
FROM Учебный_план

	Семестр	Отчетность
▶	1	э
	2	э
	3	э
	4	э
	8	э
	1	э
	4	э
	1	э
	2	э
	3	э
	2	э
	3	э
	2	э
	1	э
	2	э
	1	э
	1	э
	1	э
	1	э

SELECT DISTINCT Семестр,
Отчетность
FROM Учебный_план

	Семестр	Отчетность
▶	1	э
	1	э
	2	э
	2	э
	3	э
	3	э
	4	э
	4	э
	5	э
	5	э
	6	э
	6	э
	7	э
	7	э
	8	э
	8	э
	9	э
	9	э
*		

Использование ключевого слова TOP

SELECT TOP 5 * FROM Студенты

ID_Студент	Фамилия	Имя	Отчество	Номер_Группы	Год_поступления
1	Агапов	Иван	Иванович	2002\1	2002
2	Агупов	Петр	Александрович	2000\1	2000
3	Агуреев	Дмитрий	Александрович	2002\1	2002
4	Акатьева	Мария	Алексеевна	2000\1	1999
5	Акулов	Алексей	Юрьевич	2000\2	2000
*					

SELECT TOP 10 PERCENT * FROM Студенты

ID_Студент	Фамилия	Имя	Отчество	Номер_Группы	Год_поступления
1	Агапов	Иван	Иванович	2002\1	2002
2	Агупов	Петр	Александрович	2000\1	2000
3	Агуреев	Дмитрий	Александрович	2002\1	2002
4	Акатьева	Мария	Алексеевна	2000\1	1999
5	Акулов	Алексей	Юрьевич	2000\2	2000
6	Алексеев	Иван	Александрович	2002\2	2002
7	Амаев	Тамерлан	Джабраилович	2001\2	2000
8	Аюбова	Оксана	Аюбовна	2001\2	2001
9	Барыкин	Юрий	Владимирович	2000\2	2000
10	Басов	Константин	Викторович	2000\2	2000
11	Бибчук	Мария	Борисовна	2000\1	2000
12	Белова	Ирина	Владимировна	2002\2	2002
*					

Влияние порядка сортировки на возвращаемый набор данных

```
SELECT TOP 10 PERCENT *  
FROM Студенты  
ORDER BY Номер_Группы
```

	ID_Студент	Фамилия	Имя	Отчество	Номер_Группы	Год_поступления
▶	2	Агупов	Петр	Александрович	2000\1	2000
	4	Акатьева	Мария	Алексеевна	2000\1	1999
	11	Бebчук	Мария	Борисовна	2000\1	2000
	26	Голдобин	Михаил	Александрович	2000\1	2000
	30	Гулько	Александр	Юрьевич	2000\1	2000
	31	Гунько	Екатерина	Сергеевна	2000\1	2000
	55	Кривченков	Михаил	Юрьевич	2000\1	2000
	58	Лазаренко	Екатерина	Владимировна	2000\1	2000
	61	Лебедев	Андрей	Евгеньевич	2000\1	2000
	64	Лейпунская	Анна	Михайловна	2000\1	2000
	73	Маслова	Анна	Владимировна	2000\1	2000
	76	Митина	Светлана	Олеговна	2000\1	2000
*						

Использование ключевого слова WITH TIES

SELECT TOP 10 PERCENT WITH TIES * FROM Студенты

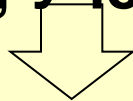
	ID_Студент	Фамилия	Имя	Отчество	Номер_Группы	Год_поступления
▶	2	Агупов	Петр	Александрович	2000\1	2000
	4	Акатьева	Мария	Алексеевна	2000\1	1999
	11	Бebчук	Мария	Борисовна	2000\1	2000
	26	Голдобин	Михаил	Александрович	2000\1	2000
	30	Гулько	Александр	Юрьевич	2000\1	2000
	31	Гунько	Екатерина	Сергеевна	2000\1	2000
	55	Кривченков	Михаил	Юрьевич	2000\1	2000
	58	Лазаренко	Екатерина	Владимировна	2000\1	2000
	61	Лебедев	Андрей	Евгеньевич	2000\1	2000
	64	Лейпунская	Анна	Михайловна	2000\1	2000
	73	Маслова	Анна	Владимировна	2000\1	2000
	76	Митина	Светлана	Олеговна	2000\1	2000
	79	Никольская	Анастасия	Александровна	2000\1	2000
	81	Панков	Дмитрий	Геннадьевич	2000\1	2000
	83	Пасенова	Медея	Гераклитовна	2000\1	2000
	84	Петрова	Алина	Николаевна	2000\1	2000
	92	Сибгатуллина	Таисия	Насимовна	2000\1	2000
	95	Соловьев	Антон	Алексеевич	2000\1	2000
	99	Стыценко	Илья	Владимирович	2000\1	2000
	104	Франтова	Елена	Владимировна	2000\1	2000
	111	Шваб	Кирилл	Юрьевич	2000\1	2000
*						

Предложение <Список_выбора>

```
<Список_выбора> ::=  
{ *  
| { <Имя_таблицы> | <Псевдоним_таблицы> }.*  
| { <Имя_столбца> | <Выражение> }  
[ [ AS ] <Псевдоним_столбца>]  
| <Псевдоним_столбца> = <Выражение>  
} [ ,...,n ]
```

```
SELECT ID_Дисциплина, Наименование,  
Семестр
```

```
FROM Дисциплина, Учебный_план
```



```
SELECT Дисциплина.ID_Дисциплина,  
Наименование,  
Семестр  
FROM Дисциплина, Учебный_план
```

Запрос для таблицы «Студенты», представляющий фамилию, имя и отчество в одной колонке

Используем операцию конкатенации (сложения) символьных строк и значение ФИО в качестве псевдонима столбца:

```
SELECT TOP 10 Фамилия + ' ' + Имя + ' ' + Отчество as ФИО,  
Номер_Группы
```

```
FROM Студенты
```

	ФИО	Группа
▶	Агапов Иван Иванович	2002\1
	Агулов Петр Александрович	2000\1
	Агуреев Дмитрий Александрович	2002\1
	Акатьева Мария Алексеевна	2000\1
	Акулов Алексей Юрьевич	2000\2
	Алексеев Иван Александрович	2002\2
	Амаев Тамерлан Джабраилович	2001\2
	Аюбова Оксана Аюбовна	2001\2
	Барыкин Юрий Владимирович	2000\2
	Басов Константин Викторович	2000\2
*		

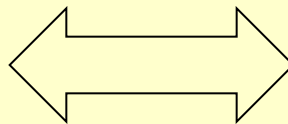
Раздел FROM

FROM { <Источник_данных> } [,...,n]

<Источник_данных> ::=

<имя_таблицы> [[AS] <псевдоним_таблицы>] |
<связка_таблиц>

Имя таблицы
«Учебный_план»



Псевдоним
tpl

Раздел WHERE

WHERE <условие_отбора>|
<имя_столбца> {= | *= | =*} <имя_столбца>

SELECT Фамилия, Имя, Отчество, Номер_Группы,
Год_поступления
FROM Студенты
WHERE Год_поступления < 2000

	Фамилия	Имя	Отчество	Номер_Группы	Год_поступления
▶	Акатьева	Мария	Алексеевна	2000\1	1999
	Козлова	Нина	Сергеевна	2000\2	1999
*					

Оператор BETWEEN

<выражение> [NOT] **BETWEEN** <начало_диапазона>
AND <конец_диапазона>

(<выражение> >=<начало_диапазона>) **AND** (<выражение> <=<конец_диапазона>)

SELECT Наименование, Семестр, Количество_часов

FROM Учебный_план INNER JOIN Дисциплины ON

Учебный_план.ID_Дисциплина =

Дисциплины.ID_Дисциплина

WHERE Количество_часов

BETWEEN 50 AND 100

	Наименование	Семестр	Количество_часов
▶	Английский язык	1	90
	Английский язык	3	90
	Английский язык	4	100
	Английский язык	8	80
	История России	1	74
	Концепции современного естествознания	3	56
	Математический анализ	2	56
	Дискретная математика	4	54
	Основы алгебры и геометрии	1	52
	Основы алгебры и геометрии	2	52
	Информатика	1	92
	Основы программирования	1	52
	Теория вероятности и математическая статистика	3	54
	Операционная система UNIX	7	56
	Финансы и кредит	5	50
	Менеджмент	5	50
	Статистика	8	50
	Предметно-ориентированные информационные системы	8	50
*			

Оператор IN

<выражение> [NOT] **IN** (<выражение1>, ..., <выражениеN>)

```
SELECT Наименование, Семестр, Количество_часов
FROM Учебный_план INNER JOIN Дисциплины ON
Учебный_план.ID_Дисциплина = Дисциплины.ID_Дисциплина
WHERE Наименование IN ('Английский язык', 'Физическая
культура')
```

	Наименование	Семестр	Количество_часов
▶	Английский язык	1	90
	Английский язык	2	110
	Английский язык	3	90
	Английский язык	4	100
	Английский язык	8	80
	Физическая культура	1	138
	Физическая культура	2	130
	Физическая культура	3	140
*			

Оператор LIKE

<Символьное_выражение> [NOT] LIKE <образец>

Символы-разделители	Значение
%	'%кош%' → 'кошка', 'окошко', 'лукошко', 'кошмар' и т.п.
_	'программ_' → 'программа', 'программ', 'программы', но не 'программист' или 'программой'
[ABC0-9]	'[A-Z]%' - образец любой последовательности символов, начинающейся с буквы латинского алфавита
[^ABC0-9]	'%[^0-9]' - образец любой последовательности символов, которая не должна заканчиваться цифрой

```
SELECT Фамилия, Имя, Отчество,  
Должность  
FROM Кадровый_состав  
WHERE Должность LIKE '%пр%'
```

	Фамилия	Имя	Отчество	Должность
▶	Сидоров	Самуил	Савельевич	Проф.
	Гиацинтова	Галина	Григорьевна	Проф.
	Петров	Павел	Петрович	Ст.преп.
	Китов	Кирилл	Кириллович	Проф.
	Воробьева	Вера	Васильевна	Ст.преп.
*				

Раздел ORDER BY (сортировка)

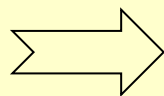
ORDER BY {<условие_сортировки>[ASC | DESC] }[,...,n]

SELECT TOP 20 Наименование, Семестр, Количество_часов
FROM Учебный_план, Дисциплины
WHERE (Учебный_план.ID_Дисциплина =
Дисциплины.ID_Дисциплина)

ORDER BY

	Наименование	Семестр	Количество_часов
▶	Физическая культура	1	138
	Английский язык	1	90
	Информатика	1	90
	История России	1	72
	Основы алгебры и геометрии	1	52
	Основы программирования	1	52
	Культурология	1	32
	Социология	1	32
	Политология	1	32
	Психология и педагогика	1	32
	Математический анализ	1	26
	Математический анализ	1	26
	История мировых цивилизаций	1	20
	Введение в специальность	1	12
	Физическая культура	2	130
	Английский язык	2	110
	Математический анализ	2	56
	Основы алгебры и геометрии	2	52
	Экономическая теория	2	34
	Философия	2	32
*			

Раздел UNION



1>
UNION [ALL]
<Спецификация_Запроса_
2>
...
[UNION [ALL]]
<Спецификация_Запроса_
n>

```
SELECT Фамилия, Имя, Отчество, Должность, Телефон  
FROM Кадровый_состав  
WHERE Телефон LIKE '120%'
```

UNION

```
SELECT Фамилия, Имя, Отчество, Новый_столбец = 'Студент',  
Телефон  
FROM Студенты  
WHERE Телефон LIKE '120%'  
ORDER BY Фамилия
```

	Фамилия	Имя	Отчество	Должность	Телефон
▶	Барыкин	Юрий	Владимирович	Студент	120-5448
	Гулько	Александр	Юрьевич	Студент	120-7787
	Кессель	Глеб	Юрьевич	Студент	120-1112
	Китов	Кирилл	Кириллович	Проф.	120-4668
	Козлова	Нина	Сергеевна	Студент	120-3335
	Цветкова	Светлана	Сидоровна	Доцент	120-1716
*					