

Лекция #6

# Базы данных

Михаил Моисеев

SQL-DML #2

# INSERT

Добавление одной записи.

**INSERT INTO** <имя\_таблицы> [ (<имя\_поля>,....) ]  
**VALUES** (<значение>,....)

- INSERT INTO students VALUES (1, 'Иван', 'Иванов', 2, 24)
- INSERT INTO students(id, fname, group\_id) VALUES (1, 'Иван', 2)
- INSERT INTO students VALUES (1, 'Иван', 'Иванов',  
(SELECT groups.id FROM groups WHERE number= '4081/1'), 24)
- INSERT INTO students VALUES (1, 'Иван', 'Иванов',  
(SELECT gid FROM (SELECT FIRST 1 groups.id AS gid,  
COUNT(students.id) AS s\_count FROM groups, students  
WHERE students.group\_id = groups.id GROUP BY groups.id ORDER BY  
s\_count)), 24);

# DELETE

Удаление одной или нескольких записей.

**DELETE FROM** <имя\_таблицы>  
[ **WHERE** <условие> ]

- DELETE FROM students;
- DELETE FROM groups WHERE number = '4081/1'
- DELETE FROM students WHERE age =  
(SELECT MAX(age) FROM students)
- DELETE FROM students WHERE  
(SELECT AVG(CAST(mark AS float)) FROM stud\_results  
WHERE stud\_results.student\_id = students.id ) < 3.5

# UPDATE

Изменение атрибутов одной или нескольких записей.

**UPDATE** <имя\_таблицы>

**SET** <имя\_столбца>=<значение>,...

[**WHERE** <условие>]

- UPDATE students SET group\_id = 1
- UPDATE students SET group\_id = 1 WHERE age > 20
- UPDATE stud\_results SET mark = 4 WHERE id IN  
(SELECT stud\_results.id FROM stud\_results, groups  
WHERE stud\_results.student\_id = groups.main\_student\_id AND  
stud\_results.semestr = 7)
- UPDATE stud\_results SET mark= (SELECT max(mark) FROM stud\_results)  
WHERE student\_id = 1

# UNION

Объединение результатов из двух выборок

<выборка1> **UNION** <выборка2>

- SELECT fname, lname FROM Students  
UNION  
SELECT fname, lname FROM Teachers

# OUTER JOIN

Внешнее соединение таблиц по заданному условию

<таблица1> [**FULL|LEFT|RIGHT**] **OUTER JOIN** <таблица2>  
**ON** <условие>

- SELECT distinct students.fname, students.lname, teachers.fname, teachers.lname FROM students **FULL OUTER JOIN** teachers **ON** students.fname = teachers.fname AND students.lname = teachers.lname
- SELECT ness\_subject\_id FROM subjects **RIGHT OUTER JOIN** subject\_rel **ON** subjects.id = subject\_rel.subject\_id WHERE subjects.name = 'Математика';

# OUTER JOIN #2

Может заменить соединение таблиц в конструкции WHERE:

- ❑ SELECT fname, lname, number FROM students, groups  
WHERE students.group\_id = groups.id
- ❑ SELECT number, fname, lname FROM  
students LEFT OUTER JOIN groups ON students.group\_id = groups.id
- ❑ SELECT fname, lname, specs.info FROM students, groups, specs  
WHERE (students.group\_id = groups.id) and (groups.spec\_id = specs.id)
- ❑ SELECT fname, lname, specs.info FROM groups  
LEFT OUTER JOIN specs ON groups.spec\_id = specs.id  
LEFT OUTER JOIN students ON students.group\_id = groups.id

# INNER JOIN

Внутреннее соединение таблиц по заданному условию

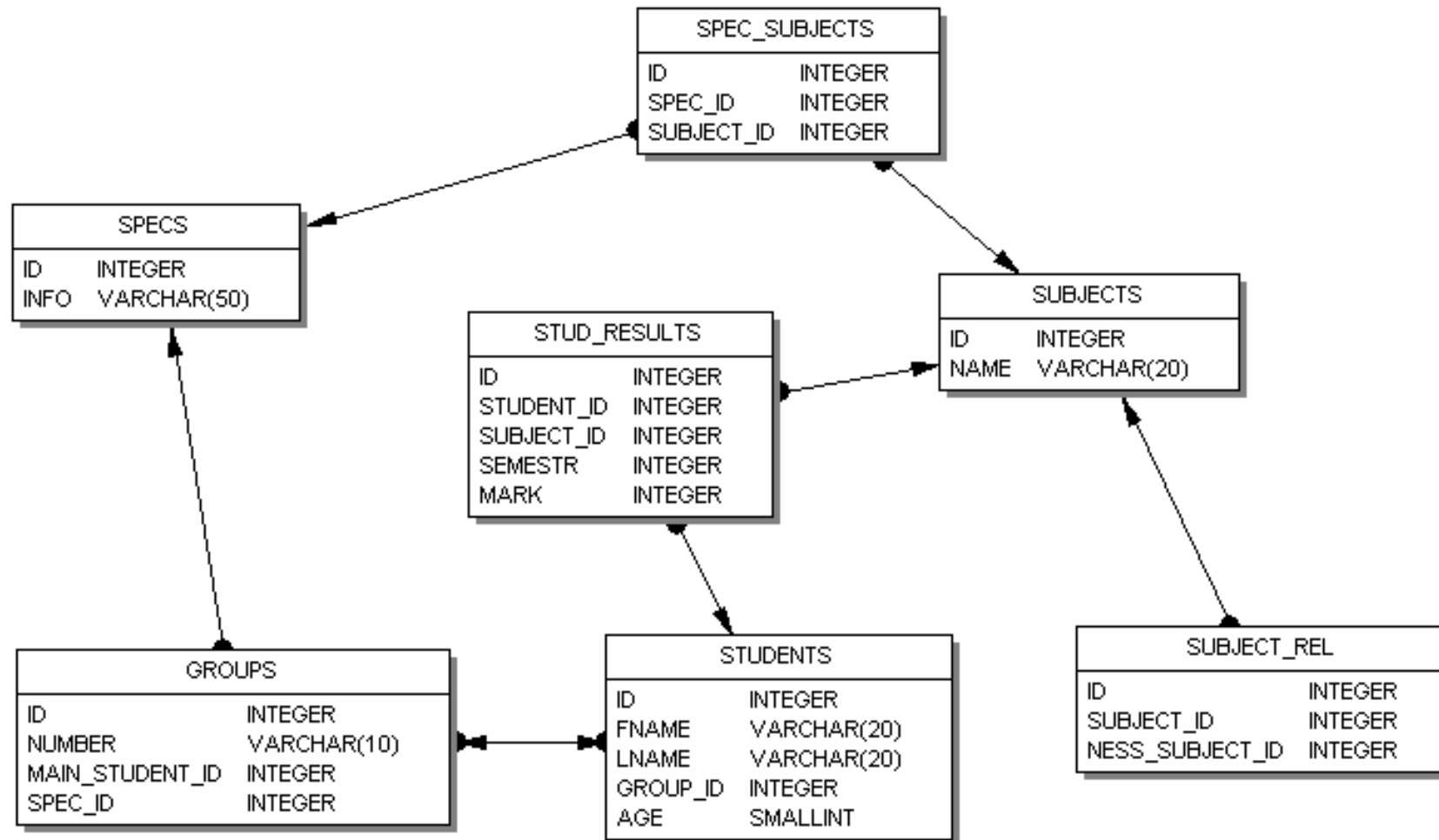
- ❑ SELECT number, fname, lname FROM  
students **FULL OUTER JOIN** groups ON students.group\_id = groups.id
- ❑ SELECT number, fname, lname FROM  
students **LEFT OUTER JOIN** groups ON students.group\_id = groups.id
- ❑ SELECT number, fname, lname FROM  
students **RIGHT OUTER JOIN** groups ON students.group\_id = groups.id
- ❑ SELECT number, fname, lname FROM  
students **INNER JOIN** groups ON students.group\_id = groups.id



# Вопросы

- Какие операторы языка SQL-DML вы знаете ?
- В каких частях этих операторов можно использовать вложенные запросы ?
- Для чего предназначен оператор IN, предложите пример использования.
- Что является результатом оператора UNION ?
- Чем отличаются разные типы соединений JOIN ?
- Какие операции можно выполнять с помощью JOIN ?
- Сколько таблиц может соединять JOIN ?
- Какие операторы используются для управления метаданными ?
- Какие типы ограничений на поля таблицы существуют ?
- Можно ли изменять структуру уже созданной таблицы ?

# Схема БД



# Контрольная работа #2

- Выполнить добавление нового студента в группу с наилучшей успеваемостью
- Выбрать в каждой группе старосту – студента с наибольшим числом оценок выше 3, использовать один из вариантов JOIN