

Филиппов Дмитрий, М3439

## Домашнее задание 5.

**Схема БД:** Courses(courseId, courseName), Lecturers(lecturerId, lecturerName), Marks(studentId, courseId, mark), Groups(groupId, groupName), Students(studentId, studentName, groupId), Plan.

Запросы:

### 1. Информация о студентах с заданной оценкой по предмету «Базы данных»

- Реляционная алгебра:  $\pi_{studentId, studentName, groupId}(\sigma_{mark=fixMark, courseName='Базы данных'}(Students \bowtie Marks \bowtie Groups \bowtie Courses))$
- SQL: SELECT S.studentId, S.studentName, G.groupId FROM (Students S NATURAL JOIN Marks M NATURAL JOIN Groups G NATURAL JOIN Courses C) where (M.mark=fixMark AND C.courseName = 'Базы данных');

### 2. Информация о студентах, не имеющих оценки по предмету «Базы данных»

- Среди всех студентов:

- Реляционная алгебра:  $Students - \pi_{studentId, studentName, groupId}(\sigma_{mark=fixMark, courseName='Базы данных'}(Students \bowtie Marks \bowtie Groups \bowtie Courses))$
- SQL: SELECT \* FROM Students EXCEPT ALL SELECT S.studentId, S.studentName, G.groupId FROM (Students S NATURAL JOIN Marks NATURAL JOIN Groups G NATURAL JOIN Courses C) where (M.mark=fixMark AND C.courseName = 'Базы данных');

- Среди всех студентов, у которых есть предмет:

- Реляционная алгебра:  $Students - \pi_{studentId, studentName, groupId}(\sigma_{mark=fixMark, courseName='Базы данных'}(Students \bowtie Marks \bowtie Groups \bowtie Courses)) \wedge (\pi_{studentId, studentName, groupId}(\sigma_{courseName='Базы данных'}(Students \bowtie Courses \bowtie Groups \bowtie Plan)))$
- SQL: SELECT \* FROM Students EXCEPT ALL SELECT S.studentId, S.studentName, G.groupId FROM (Students S NATURAL JOIN Marks M NATURAL JOIN Groups g NATURAL JOIN Courses C) where (M.mark=fixMark AND C.courseName = 'Базы данных') INTERSECT SELECT (S.studentId, S.studentName, G.groupId) FROM (Students S NATURAL JOIN Courses C NATURAL JOIN Groups g NATURAL JOIN Plan P) where C.courseName = 'Базы данных';

### 3. Информацию о студентах, имеющих хотя бы одну оценку у заданного лектора

- Реляционная алгебра:  $\pi_{studentId, studentName, groupId}(\sigma_{lecturerName=fixLecturerName}(Lecturers \bowtie Marks \bowtie Groups \bowtie Plan))$
- SQL: SELECT S.studentId, S.studentName, G.groupId FROM (Lecturers L NATURAL JOIN Marks M NATURAL JOIN Groups G NATURAL JOIN Plan P) where L.lecturerName = fixLecturerName;

### 4. Идентификаторы студентов, не имеющих ни одной оценки у заданного лектора

- Реляционная алгебра:  $\pi_{studentId}(Students - \pi_{studentId, studentName, groupId}(\sigma_{lecturerName=fixLecturerName}(Lecturers \bowtie Marks \bowtie Groups \bowtie Plan)))$

- SQL: ☺

5. Студентов, имеющих оценки по всем предметам заданного лектора

- Реляционная алгебра:  $Students \bowtie_{\sigma_{lecturerName=fixLecturerName}} (Marks \bowtie Courses \bowtie Lecturers \bowtie Plan)$

- SQL: ☺

6. Для каждого студента имя и предметы, которые он должен посещать

- Реляционная алгебра:  $\pi_{studentName, courseName}(Students \bowtie Courses \bowtie Plan)$

- SQL: ☺

7. По лектору всех студентов, у которых он хоть что-нибудь преподавал

- Реляционная алгебра:  $\pi_{studentId, studentName}(\sigma_{lecturerName=fixLecturerName}(Lecturers \bowtie Students \bowtie Plan))$

- SQL: ☺

8. Пары студентов, такие, что все сданные первым студентом предметы сдал и второй студент

- Реляционная алгебра:  $\rho_{studentName=fstudentName, studentId=fstudentId}(\pi_{studentId, studentName, courseId, courseName}(\sigma_{mark \geq 60}(Students \bowtie Courses \bowtie Marks))) * \rho_{studentName=fstudentName, studentId=fstudentId}(\pi_{studentId, studentName, courseId, courseName}(\sigma_{mark \geq 60}(Students \bowtie Courses \bowtie Marks)))$

- SQL: ☺

9. Такие группы и предметы, что все студенты группы сдали предмет

- Реляционная алгебра:  $(\pi_{groupName, studentName} Students) * (\pi_{studentName, courseName}(\sigma_{mark \geq 60} Marks))$

- SQL: ☺

10. Средний балл студента

- По идентификатору:

- Реляционная алгебра:  $avg_{mark, \emptyset}(\pi_{mark}(\sigma_{studentId=fixStudentId}(Students \bowtie Marks)))$
- SQL: ☺

- Для каждого студента:

- Реляционная алгебра:  $avg_{mark, \{studentId, studentName\}}(\pi_{studentId, studentName, mark}(Students \bowtie Marks))$
- SQL: ☺

11. Средний балл средних баллов студентов каждой группы

- Реляционная алгебра: ☺

- SQL: ☺

12. Для каждого студента число предметов, которые у него были, число сданных предметов и число не сданных предметов

- Реляционная алгебра: ☺

- SQL: ☺