Волков Александр, M3437

Домашнее задание

Структура базы данных «Университет»:

- *Students(StudentId, StudentName, GroupId)*
- *Groups(GroupId, GroupName)*
- *Courses(CourseId, CourseName)*
- *Lecturers(LecturerId, LecturerName)*
- *Plan(GroupId, CourseId, LecturerId)*
- *Marks(StudentId, CourseId, Mark)*

1. Информацию о студентах, с заданной оценкой по предмету «Базы данных».

$$\begin{array}{l}\text{select } \mathit{Students.StudentId}, \mathit{Students.StudentName}, \mathit{Students.GroupId}\\ \text{from } \mathit{Students}\\ \mathit{where} \, \exists \mathit{Cources} \, \exists \mathit{Marks} \, (\mathit{Courses.CourseName} = \text{'}\mathit{Базы\ данных}\text{'} \land\\ \mathit{Students.StudentId} = \mathit{Marks.StudentId} \land \mathit{Marks.CourseId} = \mathit{Courses.CourseId} \land \mathit{Marks.Mark} = ?)\end{array}$$

$$\begin{array}{l}\mathit{WithMakDatabases}(\mathit{StudentId}, \mathit{StudentName}, \mathit{GroupId}) :\text{-}\\ \quad \mathit{Students}(\mathit{StudentId}, \mathit{StudentName}, \mathit{GroupId}),\\ \qquad \mathit{Courses}(\mathit{CourseId}, \mathit{CourseName}),\\ \qquad \mathit{Marks}(\mathit{Mark}, \mathit{CourseId}, \mathit{StudentId}),\\ \qquad \mathit{CourseName} = \text{'}\mathit{Баззы\ данных}\text{'},\\ \qquad\qquad \mathit{Mark} = ?\end{array}$$

SELECT StudentId, StudentName, GroupId FROM Students

WHERE EXISTS (

SELECT * FROM Marks WHERE (

Students.StudentId = Marks.StudentId AND Marks.Mark = ? AND Marks.CourseId in

(SELECT Courses.CourseId FROM Courses WHERE Courses.CourseName = 'Базы данных')));

2. Информацию о студентах не имеющих оценки по предмету «Базы данных»:

   а) среди всех студентов

$$\begin{array}{l}\text{select } \mathit{Students.StudentId}, \mathit{Students.StudentName}, \mathit{Students.GroupId}\\ \text{from } \mathit{Students}\\ \mathit{where} \, \neg\exists \mathit{Marks} \, \exists \mathit{Courses} \, (\mathit{Students.StudentId} = \mathit{Marks.StudentId} \land\\ \mathit{Marks.CourseId} = \mathit{Courses.CourseId} \land \mathit{Courses.CourseName} = \text{Базы данных})\end{array}$$

$$\begin{aligned}
&NoMarkDatabase(StudentId, StudentName, GroupId) :- \\
&\quad Students(StudentId, StudentName, GroupId), \\
&\quad\quad not\ Marks(StudentId, CourseId, Mark), \\
&\quad\quad\quad Courses(CourseId, CourseName), \\
&\quad\quad\quad\quad CourseName = \text{Базы данных}
\end{aligned}$$

```
SELECT StudentId, StudentName, GroupId FROM students
WHERE NOT EXISTS (
SELECT * FROM Marks WHERE (
 Students.StudentId = Marks.StudentId AND Marks.CourseId in
( SELECT Courses.CourseId FROM Courses WHERE Courses.CourseName = 'Базы данных' ) ) );
```

б) среди студентов, у которых есть этот предмет

$$\begin{aligned}
&\text{select } Students.StudentId, Students.StudentName, Students.GroupId \\
&\text{from } Students \\
&where\ \exists Plan\ \exists Courses\ \neg\exists Marks(\ Plan.CourseId = Courses.CourseId \land \\
&Plan.GroupId = Students.GroupId \land Courses.CourseName = \text{Базы данных} \land \\
&\quad Mark.CourseId = Plan.CourseId \land Mark.StudentId = Students.StudentId\ )
\end{aligned}$$

$$\begin{aligned}
&NoMarkDatabase(StudentId, StudentName, GroupId) :- \\
&\quad Students(StudentId, StudentName, GroupId), \\
&\quad\quad Courses(CourseId, CourseName), \\
&\quad\quad\quad Plan(CourseId, GroupId, \_), \\
&\quad\quad not\ Marks(CourseId, StudentId, \_), \\
&\quad\quad\quad CourseName = '\ Базы\,данных\ '
\end{aligned}$$

```
SELECT StudentId, StudentName, GroupId FROM Students
WHERE StudentId.GroupId IN
( SELECT Plan.GroupId FROM Plan WHERE Students.GroupId = Plan.GroupId AND
Plan.CourseId in
( SELECT Courses.CourseId FROM Courses WHERE Courses.CourseName = 'Базы данных' ) )
AND NOT EXISTS
( SELECT * FROM Marks WHERE ( Students.StudentId = Marks. StudentId AND Marks.CourseId
in
( SELECT Courses.CourseId FROM cou Courses rses WHERE Courses.CourseName =  'Базы
данных' ) ) );
```

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

$Plan.LecturerId = Lecturers.LecturerId \wedge Students.StudentId = Marks.StudentId \wedge Marks.CourseId = Plan.CourseId)$

select $Students.StudentId, Students.StudentName, Students.GroupId$
from $Students$
$where \exists Lecturers \exists Plan \exists Marks (Lecturers.LecturerName = ? \wedge$

$StudentsAtLeastOneMark(StudentId, StudentName, GroupId)$:-
$Students(StudentId, StudentName, GroupId),$
$Lecturers(LecturerId, LecturerName),$
$Plan(LecturerId, CourseId, GroupId),$
$Marks(\_, CourseId, StudentId),$
$LecturerName = ?$

SELECT StudentId, StudentName, GroupId FROM Students

WHERE Students.StudentId IN

(SELECT Marks.StudentId FROM Marks WHERE Marks.CourseId IN

(SELECT Plan.CourseId FROM Plan WHERE Plan.LecturerId IN

(SELECT Lecturers.LecturerId FROM Lecturers WHERE Lecturers.LectureName = 'Георгий Корнеев'))) ;

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

select $Students.StudentId$
from $Students$
$where \neg\exists Marks \exists Lecturers \exists Plan (Lecturers.LecturerName = ? \wedge Plan.LecturerId = Lecturers.LecturerId$

$\wedge Marks.CourseId = Plan.CourseId \wedge Students.StudentId = Marks.StudentId)$

$StudentsHaveMark(StudentId)$:-$Students(StudentId, \_, GroupId),$
$Lecturers(LecturerId, LecturerName),$
$Plan(LecturerId, CourseId, \_), Marks(\_, CourseId, StudentId), LecturerName = ?$

$StudentsWithoutLecturerMark(StudentId)$:-$Students(StudentId, \_, \_), not\ StudentsHaveMark(StudentId)$

SELECT StudentId FROM Students EXCEPT

(SELECT StudentId FROM Students

WHERE Students.StudentId IN

```
 ( SELECT Marks.StudentId FROM Marks WHERE Marks.CourseId IN
( SELECT Plan.CourseId FROM Plan WHERE Plan.LecturerId IN
( SELECT Lecturers.LecturerId FROM Lecturers WHERE Lecturers.LectureName = 'Георгий
Корнеев' ) ) ) );
```

5. Студентов, имеющих оценки по всем предметам заданного лектора

$$\begin{aligned}&\text{select}\ Students.StudentId, Students.StudentName, Students.GroupId\\ &\text{from}\ Students\\ &where\ \exists Lecturers\ \forall\ Plan\ \exists Marks\,(Lecturers.LecturerName = ? \wedge Plan.LecturerId = Lecturers.LecturerId\\ &\wedge Marks.CourseId = Plan.CourseId \wedge Students.StudentId = Marks.StudentId\,)\end{aligned}$$

$$\begin{aligned}&LecturerCourses(CourseId) :- Lecturers(LecturerId, LecturerName),\\ &\quad Plan(LecturerId, CourseId, \_), LecturerName = ?\end{aligned}$$

$$\begin{aligned}&CoursesWithoutStudentMarks(StudentId, CourseId) :- Students(StudentId, \_, \_),\\ &\quad Courses(CourseId, \_), not\ Marks(\_, CourseId, StudentId)\end{aligned}$$

$$\begin{aligned}&StudentsWithAllLectureMarks(StudentId, StudentName, GroupId) \text{:-}\\ &\quad Students(StudentId, StudentName, GroupId),\\ &LecturerCourses(CourseId), not\ CoursesWithoutStudentMarks(StudentId, CourseId)\end{aligned}$$

```
SELECT StudentId, StudentName, GroupId FROM Students
WHERE EXISTS (
        SELECT * FROM Lecturers WHERE LecturerName = ‘Георгий Корнеев’ AND NOT
EXISTS (
        SELECT * FROM Plan, Courses WHERE Students.GroupId = Plan.GroupId AND
Lecturers.LecturerId  = Plan.LecturerId and Courses.CourseId = Plan.CourseId AND NOT
EXISTS
        (SELECT * FROM Marks WHERE Marks.StudentId = Students.StudentId AND
Marks.CourseId = Plan.CourseId)));
```

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

$$\begin{aligned}&\text{select}\ Students.StudentName, Courses.CourseName\\ &\text{from}\ Students, Courses\\ &where\ \exists Plan(Students.GroupId = Plan.GroupId \wedge Plan.CourseId = Courses.CourseId)\end{aligned}$$

$$StudentCourses(StudentId, StudentName, CourseName) :- Students(StudentId, StudentName, GroupId),$$
$$Courses(CourseId, CourseName), Plan(\_, CourseId, GroupId)$$

```
SELECT StudentId, StudentName, CourseName FROM Students, Courses WHERE
Students.GroupId IN (

 SELECT Plan.GroupId FROM Plan WHERE Courses.CourseId = Plan.CourseId);
```

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

$$\begin{array}{l}\text{select } Students.StudentId, Students.StudentName \\ \text{from } Students \\ where\ \exists Lecturers\ \exists Plan(Lecturers.LecturerName = ? \wedge \\ Plan.LecturerId = Lecturers.LecturerId \wedge Studetns.GroupId = Plan.GroupId)\end{array}$$

$$StudentsLecturer(StudentId, StudentName) :- Students(StudentId, StudentName, GroupId),$$
$$Lectureres(LecturerId, LecturerName), Plan(LecturerId, \_, GroupId), LecturerName = ?$$

```
SELECT StudentId, StudentName FROM Students

WHERE Students.GroupId IN

( SELECT Plan.GroupId FROM Plan WHERE Plan.LecturerId IN

( SELECT Lecturers.LecturerId FROM Lecturers WHERE Lecturers.LecturerName = ‘Георгий
Корнеев’ ) );
```

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

$$\begin{array}{l}\text{select } Students1.StudentName, Students2.StudentName, S\ as\ \text{Student1}, S\ as\ \text{Student2} \\ \text{from Students1, Students2} \\ \text{where } \forall \text{Marks1}(\exists \text{Marks2}(\text{Marks1}.StudentId <> \text{Students1}.StudentId \vee \text{Marks1}.Mark < 60 \vee \\ \text{Marks2}.StudentId = \text{Students2}.StudentId \wedge \text{Marks2}.Mark \geq 60 \wedge Marks1.CourseId = Marks2.CourseId)\end{array}$$

$$\begin{gathered}FriendLosser(StudentId1, StudentId2, CourseId) :\text{-}\\ Students(StudentId1, StudentName, \_), Students(StudentId2, StudentName, \_)\\ Marks(Mark1, CourseId, StudentId1), Mark1 \geq 60,\\ \neg Marks(\_, CourseId, StudentId2)\end{gathered}$$

$$\begin{gathered}FriendLosser(StudentId1, StudentId2, CourseId) :\text{-}\\ Students(StudentId1, StudentName, \_), Students(StudentId2, StudentName, \_)\\ Marks(Mark1, CourseId, StudentId1), Mark1 \geq 60,\\ Marks(\_, CourseId, StudentId2), Mark2 < 60\end{gathered}$$

$$\begin{gathered}SuccessfulFriends(StudentId1, StudentId2) :\text{-}\\ Students(StudentId1, StudentName2, \_),\\ Students(StudentId2, StudentName2, \_),\\ \neg FriendLosser(StudentId1, Student2, \_)\end{gathered}$$

____

```
SELECT S1.StudentName, S2.StudentName
FROM Students AS S1, Students AS s S2 WHERE NOT EXISTS (
SELECT * FROM Marks AS M1 WHERE
        M1.StudentId = S1.StudentId AND M1.Mark >= 60 AND
        NOT EXISTS(
SELECT * FROM Marks  AS M2 WHERE
        M2.StudentId = S2.StudentId  AND  M1.CourseId = M2.CourseId  AND  M2.Mark < 60 ));
```

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

select $Courses.CourseId, Groups.GroupId$
from $Courses, Groups$
$where\ \forall Students (\exists Marks (Students.GroupId <> Groups.GroupId \vee Students.StudentId = Marks.StudentId$
$\wedge Marks.CourseId = Courses.CourseId$
$\wedge Marks.Mark \geq 60))$

$$\begin{gathered}Failed(StudentId, CourseId, GroupId) :\text{-} Students(StudentId, \_, GroupId),\\ \neg Marks(StudentId, CourseId, \_)\\ Failed(StudentId, CourseId, GroupId) :\text{-} Students(StudentId, \_, GroupId),\\ Marks(StudentId, CourseId, \_), Mark < 60\end{gathered}$$

$$\begin{gathered}SuccessfulGroups(GroupId, CourseId) :\text{-}\\ \neg Failed(\_, CouresId, GroupId), Groups(GroupId, \_), Courses(CourseId, \_)\end{gathered}$$

```
SELECT CouseId, GroupId from Gourses, Groups WHERE NOT EXISTS(
        SELECT * FROM Students WHERE Students.GroupId = Groups.GroupId
        AND NOT EXISTS (
        SELECT  * FROM Marks WHERE Marks.StudentId = Students.StudentId AND
        Marks.CourseId = Courses.CourseId AND Mark >= 60)))
```