**Cодержание урока**

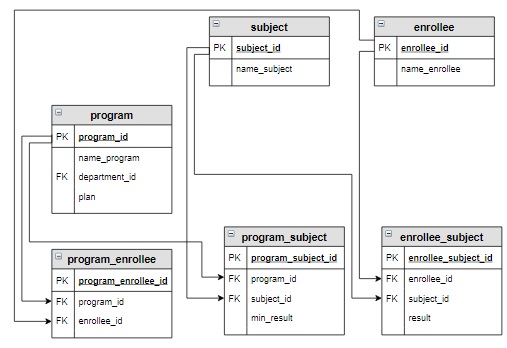
В данном уроке с помощью запросов корректировки данных для базы данных «Абитуриент» формируется список абитуриентов, рекомендованных к зачислению в университет:

* [создается таблица с суммой баллов абитуриентов по предметам ЕГЭ в соответствии с поданными заявлениями](https://stepik.org/lesson/310420/step/2?unit=292726);
* [из таблицы удаляются абитуриенты, если они не набрали минимального балла по предмету, необходимому для поступления на образовательную программу](https://stepik.org/lesson/310420/step/3?unit=292726);
* [абитуриентам, у которых есть медаль или значок ГТО, добавляются дополнительные баллы](https://stepik.org/lesson/310420/step/4?unit=292726);
* [абитуриенты сортируются в соответствии с набранными баллами по каждой образовательной программе](https://stepik.org/lesson/310420/step/5?unit=292726);
* формируется список абитуриентов, рекомендованных к зачислению ( [вставляется столбец для нумерации](https://stepik.org/lesson/310420/step/6?unit=292726), [осуществляется нумерация студентов по образовательной программе](https://stepik.org/lesson/310420/step/7?unit=292726), [выбираются абитуриенты с наибольшими баллами в соответствии с планом набора](https://stepik.org/lesson/310420/step/8?unit=292726)).

## Задание

Создать вспомогательную таблицу **applicant**,  куда включить **id** образовательной программы, **id** абитуриента, сумму баллов абитуриентов (столбец **itog**) в отсортированном сначала по **id** образовательной программы, а потом по убыванию суммы баллов виде (использовать [запрос](https://stepik.org/lesson/310418/step/10?unit=292724) из предыдущего урока).

**Фрагмент логической схемы базы данных:**



**Пояснение**

**Связанные шаги**

**Текст задания**(чтобы не прокручивать страницу) :

*Создать вспомогательную таблицу****applicant****,  куда включить****id****образовательной программы,****id****абитуриента, сумму баллов абитуриентов (столбец****itog****) в отсортированном сначала по****id****образовательной программы, а потом по убыванию суммы баллов виде (использовать [запрос](https://stepik.org/lesson/310418/step/10?unit=292724) из предыдущего урока).*

**Результат**

1

CREATE TABLE applicant AS

2

SELECT program\_enrollee.program\_id,

3

      program\_enrollee.enrollee\_id,

4

        SUM(result) AS 'itog'

5

FROM

6

  /\* program

7

    JOIN program\_enrollee

8

    ON program.program\_id = program\_enrollee.program\_id

9

      JOIN enrollee

10

      ON enrollee.enrollee\_id = program\_enrollee.enrollee\_id

11

        JOIN enrollee\_subject

12

        ON enrollee.enrollee\_id = enrollee\_subject.enrollee\_id

13

          JOIN subject

14

          ON subject.subject\_id = enrollee\_subject.subject\_id

15

            JOIN program\_subject

16

            ON subject.subject\_id = program\_subject.subject\_id

17

              AND program.program\_id = program\_subject.program\_id

18

              Option 2\*/

19

              program\_enrollee

20

               JOIN enrollee\_subject

21

                ON program\_enrollee.enrollee\_id = enrollee\_subject.enrollee\_id

22

                 JOIN program\_subject

23

                  ON program\_subject.program\_id = program\_enrollee.program\_id

24

                   AND program\_subject.subject\_id = enrollee\_subject.subject\_id

25

GROUP BY 1, 2

26

ORDER BY 1, 3 DESC;

27

​

28

SELECT \* FROM applicant;

## Задание

Из таблицы **applicant**, созданной на предыдущем шаге, удалить записи, если абитуриент на выбранную образовательную программу не набрал минимального балла хотя бы по одному предмету (использовать [запрос](https://stepik.org/lesson/310418/step/11?unit=292724) из предыдущего урока).

**Фрагмент логической схемы базы данных:**

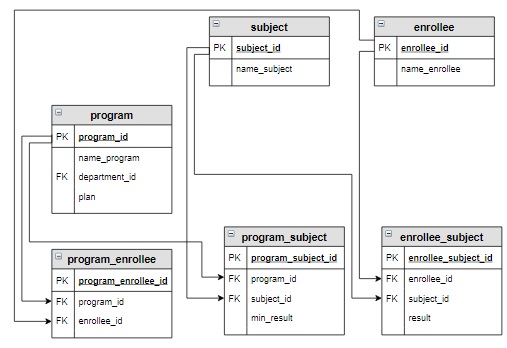
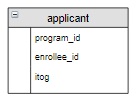


Таблица **applicant**:



**Пояснение**

**Связанные шаги**

**Текст задания**(чтобы не прокручивать страницу) :

*Из таблицы****applicant****,  созданной на предыдущем шаге, удалить записи, если абитуриент на выбранную образовательную программу не набрал минимального балла хотя бы по одному предмету (использовать [запрос](https://stepik.org/lesson/310418/step/11?unit=292724) из предыдущего урока).*

1

DELETE FROM applicant

2

USING applicant JOIN program\_subject USING(program\_id)

3

               JOIN enrollee\_subject USING(enrollee\_id, subject\_id)

4

WHERE result < min\_result;

5

​

6

DELETE FROM applicant

7

WHERE (program\_id, enrollee\_id) IN

8

(

9

   SELECT program.program\_id, enrollee.enrollee\_id

10

   FROM program

11

    JOIN program\_enrollee

12

     ON program.program\_id = program\_enrollee.program\_id

13

      JOIN enrollee

14

       ON enrollee.enrollee\_id = program\_enrollee.enrollee\_id

15

        JOIN enrollee\_subject

16

         ON enrollee.enrollee\_id = enrollee\_subject.enrollee\_id

17

          JOIN subject

18

           ON subject.subject\_id = enrollee\_subject.subject\_id

19

            JOIN program\_subject

20

             ON subject.subject\_id = program\_subject.subject\_id

21

              AND program.program\_id = program\_subject.program\_id

22

   WHERE result < min\_result

23

);

24

​

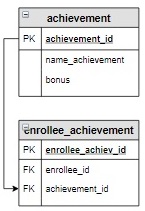
25

SELECT \* FROM applicant;

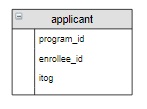
## Задание

Повысить итоговые баллы абитуриентов в таблице **applicant** на значения дополнительных баллов (использовать [запрос](https://stepik.org/lesson/310418/step/7?unit=292724) из предыдущего урока).

**Фрагмент логической схемы базы данных:**



**Структура корректируемой таблицы:**



**Пояснение**

**Связанные шаги**

**Текст задания**(чтобы не прокручивать страницу) :

*Повысить итоговые баллы абитуриентов в таблице****applicant****на значения дополнительных баллов (использовать [запрос](https://stepik.org/lesson/310418/step/7?unit=292724) из предыдущего урока).*

**Результат**

1

UPDATE

2

  applicant, (SELECT enrollee.enrollee\_id, SUM(bonus) AS 'Бонус'

3

   FROM

4

        enrollee

5

         JOIN enrollee\_achievement

6

          ON enrollee.enrollee\_id = enrollee\_achievement.enrollee\_id

7

           JOIN achievement

8

            ON achievement.achievement\_id = enrollee\_achievement.achievement\_id

9

             GROUP BY enrollee.enrollee\_id) AS wert

10

SET applicant.itog = applicant.itog + wert.Бонус

11

WHERE wert.enrollee\_id=applicant.enrollee\_id;

12

​

13

SELECT \* FROM applicant;