Орищенко Виталий

@Vitosh992

Группа 1

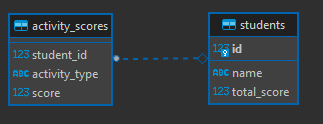
**Архитектор ИИ**

**Задание 10**

1.

**create** **table** students (id SERIAL **primary** **key**, **name** **text**, total\_score **integer**)

**create** **table** activity\_scores (student\_id **integer**, **foreign** **key** (student\_id) **references** students(id), activity\_type **text**, score **integer**)



Для упрощения работы создадим сначала триггер на просчет суммы баллов для каждого студента при изменении, добавлении или удалении данных в таблицы total\_score

**create** **or** **replace** **function** update\_total\_score()

**returns** **trigger** **as** **$$**

**begin**

**update** students

**set** total\_score = (

**select** **COALESCE**(**SUM**(score), 0)

**from** activity\_scores

**where** activity\_scores.student\_id = **new**.student\_id

)

**where** students.id = **new**.student\_id;

**return** **new**;

**end**;

**$$** **language** plpgsql;

**create** **trigger** update\_total\_score\_trigger

**after** **insert** **or** **update** **or** **delete**

**on** activity\_scores

**for** **each** **row**

**execute** **function** update\_total\_score();

**insert** **into** activity\_scores (student\_id, activity\_type, score) **values**

(1,'Test', 15 ),

(1,'Test', 30 ),

(2,'Test', 20 ),

(2,'Test',40 ),

(2,'Test', 11 ),

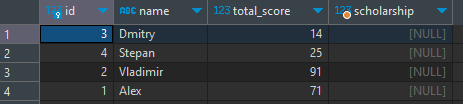
(3,'Test', 14 ),

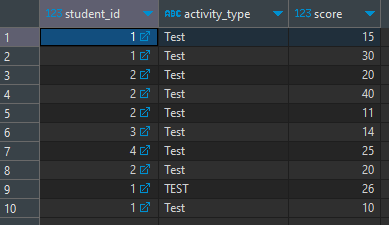
(4,'Test', 25 ),

(2,'Test', 20 ),

(1,'TEST', 26),

(1,'Test', 10 )





Создаем функцию, которая будет подсчитывать размер стипендии по условиям:

**create** **or** **replace** **function** calculate\_scholarship(iden **int**)

**returns** **numeric** **as** **$$**

**declare**

scholarship **numeric**;

scores **numeric**;

**begin**

**select** total\_score **into** scores **from** students **where** id = iden;

**if** scores >= 90 **then**

scholarship := 1000;

**elsif** scores < 90 **and** scores >= 80 **then**

scholarship := 500;

**else**

scholarship := 0;

**end** **if**;

**return** scholarship;

**end**;

**$$** **language** plpgsql;

**select** (calculate\_scholarship(1))

**select** (calculate\_scholarship(2))

**select** (calculate\_scholarship(3))

**select** (calculate\_scholarship(4))









Создадим новую функцию со вложенной функцией, которая будет использовать подсчитанный размер стипендии, чтобы занести его в новую ячейку таблицы students.

**create** **or** **replace** **function** calculate\_and\_update\_scholarship()

**returns** **trigger** **as** **$$**

**begin**

**update** students

**set** scholarship = calculate\_scholarship(**new**.student\_id)

**where** students.id = **new**.student\_id;

**return** **new**;

**end**;

**$$** **language** plpgsql;

Создадим на эту функцию триггер, который будет срабатывать при любом изменении данных в таблице activity\_scores.

**create** **trigger** calculate\_scholarship\_trigger

**after** **insert** **or** **update** **or** **delete**

**on** activity\_scores

**for** **each** **row**

**execute** **function** calculate\_and\_update\_scholarship();

Загрузим в activity\_scores новые данные для каждого студента, чтобы задействовать новый тригер:

**insert** **into** activity\_scores (student\_id, activity\_type, score) **values**

(1,'Test', 1 ),

(2,'Test', 1 ),

(3,'Test', 1 ),

(4,'Test', 1 )

Проверим таблицу students

