**Лабораторная работа по теме "Язык PL/pgSQL" (ДЗ-10)**

*\* Не совсем разобрался где отслеживать стипендию студентов после выполнения функции (Задача 1), поэтому переделал немного таблицу students из задания*

Таблица *students*

*CREATE TABLE public.students (*

*id serial4 NOT NULL,*

*"name" text NOT NULL,*

*total\_score int4 NULL,*

*scholarship int4 NULL,*

*CONSTRAINT students\_pkey PRIMARY KEY (id)*

*);*

Таблица *activity\_scores*

*CREATE TABLE public.activity\_scores (*

*student\_id int4 NOT NULL,*

*activity\_type text NOT NULL,*

*score int4 NULL,*

*CONSTRAINT activity\_scores\_fk FOREIGN KEY (student\_id) REFERENCES public.students(id)*

*);*

**Задача 1: Расчет стипендии для студентов**

Функция *calculate\_scholarship*

*create or replace function calculate\_scholarship()*

*returns trigger as*

*$$*

*begin*

*update public.students set scholarship = (*

*select*

*case*

*when sum(as2.score) >= 90 then 1000*

*when sum(as2.score) >= 80 and sum(as2.score) < 90 then 500*

*else 0*

*end as scholarship*

*from activity\_scores as2*

*where as2.student\_id = new.student\_id)*

*where id = new.student\_id;*

*return null;*

*end;*

*$$*

*language 'plpgsql';*

Триггер *update\_scholarship\_trigger*

*create or replace trigger update\_scholarship\_trigger*

*after insert or update on public.activity\_scores*

*for each row*

*execute procedure public.calculate\_scholarship();*

**Задача 2: Учет баллов студентов**

Триггерная функция *update\_total\_score*

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

*returns trigger as*

*$$*

*begin*

*perform sum\_score(new.student\_id);*

*return null;*

*end;*

*$$*

*language 'plpgsql';*

Функция для расчета общего бала *sum\_score*

*create or replace function sum\_score(s\_id integer)*

*returns void as*

*$$*

*declare*

*s\_row record;*

*begin*

*for s\_row in*

*select*

*as2.student\_id*

*, sum(as2.score) as score*

*from public.activity\_scores as2*

*where*

*as2.student\_id = s\_id*

*group by as2.student\_id*

*loop*

*update public.students*

*set total\_score = s\_row.score*

*where id = s\_row.student\_id;*

*end loop;*

*end;*

*$$*

*language 'plpgsql';*

Триггер *update\_scholarship\_trigger*

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

*after insert on public.activity\_scores*

*for each row*

*execute procedure public.update\_total\_score();*

Примеры вывода результатов:

|  |  |
| --- | --- |
| Начальное состояние таблиц | |
| students | activity\_scores |
|  |  |
| Обновим баллы активности по студенту с id=3  *общее количество баллов не меняется т.к. за это отвечает другая функция* | |
| students | activity\_scores |
|  |  |
| Добавим новую активность студенту с id=3 | |
| students | activity\_scores |
|  |  |