**Лабораторная работа 5**

• Повторить теорию о транзакциях.

• Повторю теорию о триггере.

• Чтобы создать три триггера: один для вставки, обновления и удаления.

**1)Insert Trigger & Update trigger**

**CREATE** **TABLE** средние\_оценки (

ученик\_id **INT** **PRIMARY** **KEY**,

средняя\_оценка **NUMERIC**(4, 2)

);

Для работы с триггерами создам отдельную таблицу со средними оценками учеников.

Функция для триггера:

**CREATE** **OR** **REPLACE** **FUNCTION** обновить\_среднюю\_оценку()

**RETURNS** **TRIGGER** **AS** **$$**

**BEGIN**

--Update trigger (если уже есть такая запись)

**IF** **EXISTS** (**SELECT** 1 **FROM** средние\_оценки **WHERE** ученик\_id = **NEW**.ученик\_id) **THEN**

**UPDATE** средние\_оценки

**SET** средняя\_оценка = (

**SELECT** **AVG**(оценка)::**NUMERIC**(4, 2)

**FROM** оценки

**WHERE** ученик\_id = **NEW**.ученик\_id

)

**WHERE** ученик\_id = **NEW**.ученик\_id;

**ELSE**

-- Insert trigger( если такой записи еще нет)

**INSERT** **INTO** средние\_оценки (ученик\_id, средняя\_оценка)

**VALUES** (

**NEW**.ученик\_id,

(**SELECT** **AVG**(оценка)::**NUMERIC**(4, 2)

**FROM** оценки

**WHERE** ученик\_id = **NEW**.ученик\_id)

);

**END** **IF**;

**RETURN** **NEW**;

**END**;

**$$** **LANGUAGE** plpgsql;

Триггер:

**CREATE** **TRIGGER** рассчитать\_среднюю\_оценку

**AFTER** **INSERT** **ON** оценки

**FOR** **EACH** **ROW**

**EXECUTE** **FUNCTION** обновить\_среднюю\_оценку();

Данный триггер обновляет среднюю оценку ученика при добавлении новой оценки.   
Если средняя оценка уже существует (уже вычислялась), то ее значение обновляется (**update**).  
В противном случае такая запись создается(**insert**)

2) Delete Trigger

При удалении ученика, логичным будет удалять и связанные с ним данные (оценки, посещаемость)

Данный триггер будет срабатывать при удалении ученика (after delete on ученики), и будет удалять все связанные с ним данные

**CREATE** **OR** **REPLACE** **FUNCTION** удалить\_данные\_ученика()

**RETURNS** **TRIGGER** **AS** **$$**

**BEGIN**

-- Удаление оценок

**DELETE** **FROM** оценки

**WHERE** ученик\_id = **OLD**.ученик\_id;

-- Удаление данных о посещаемости

**DELETE** **FROM** посещаемость

**WHERE** ученик\_id = **OLD**.ученик\_id;

**RETURN** **OLD**;

**END**;

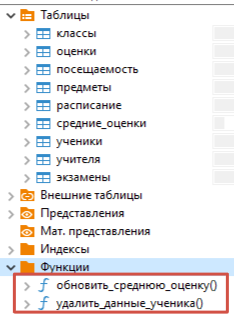
**$$** **LANGUAGE** plpgsql;

**CREATE** **TRIGGER** удалить\_связанные\_данные

**AFTER** **DELETE** **ON** ученики

**FOR** **EACH** **ROW**

**EXECUTE** **FUNCTION** удалить\_данные\_ученика();

\*Данные функции были созданы успешно