# Практическая работа №4 ПС

Студент: Болдинов Алексей ЭФМО-02-24

1. Используя таблицу РАБОЧИЙ ДЕНЬ, создайте функцию, которая на указанный день рассчитывает количества работающих сотрудников (отработавших ровно 60 минут). Результаты разбить по каждому часу.

CREATE TABLE ОПИСАНИЕ\_ОБЪЕКТА\_XML (

код\_объекта SERIAL PRIMARY KEY,

описание XML

);

CREATE TABLE Динамика\_цен (

код\_объекта INTEGER REFERENCES ОПИСАНИЕ\_ОБЪЕКТА\_XML(код\_объекта),

новая\_стоимость NUMERIC,

дата\_изменения DATE

);

1. Создание процедуры для заполнения таблицы ОПИСАНИЕ\_ОБЪЕКТА\_XML

CREATE OR REPLACE PROCEDURE заполнить\_описание\_объекта()

LANGUAGE plpgsql

AS $$

DECLARE

rec RECORD;

рейтинг INTEGER;

динамика XML;

BEGIN

FOR rec IN

SELECT

obj.код\_объекта,

obj.собственник,

obj.стоимость,

obj.этаж,

obj.общая\_площадь,

obj.жилая\_площадь,

addr.область,

addr.город,

addr.район,

addr.улица,

addr.дом,

addr.квартира

FROM объекты\_недвижимости obj

JOIN адреса addr ON obj.код\_объекта = addr.код\_объекта

LOOP

-- Определение рейтинга

рейтинг := (

SELECT RANK() OVER (PARTITION BY addr.район ORDER BY obj.стоимость DESC)

FROM объекты\_недвижимости obj

JOIN адреса addr ON obj.код\_объекта = addr.код\_объекта

WHERE obj.код\_объекта = rec.код\_объекта

);

-- Формирование XML динамики стоимости

динамика := (

SELECT XMLAGG(

XMLELEMENT(

NAME "Стоимость",

XMLATTRIBUTES(дата\_изменения AS "Дата\_установки\_стоимости"),

новая\_стоимость

)

)

FROM Динамика\_цен

WHERE код\_объекта = rec.код\_объекта

);

-- Вставка в таблицу ОПИСАНИЕ\_ОБЪЕКТА\_XML

INSERT INTO ОПИСАНИЕ\_ОБЪЕКТА\_XML (код\_объекта, описание)

VALUES (

rec.код\_объекта,

XMLELEMENT(

NAME "object",

XMLATTRIBUTES(

rec.код\_объекта AS "id",

rec.собственник AS "Собственник",

рейтинг AS "Рейтинг"

),

XMLELEMENT(NAME "Динамика\_стоимости", динамика),

XMLELEMENT(NAME "Этаж", rec.этаж),

XMLELEMENT(

NAME "Площадь",

XMLELEMENT(NAME "Общая\_площадь", rec.общая\_площадь),

XMLELEMENT(NAME "Жилая\_площадь", rec.жилая\_площадь)

),

XMLELEMENT(

NAME "Адрес",

XMLATTRIBUTES(rec.область AS "Область"),

XMLELEMENT(NAME "Город", rec.город),

XMLELEMENT(NAME "Название\_района", rec.район),

XMLELEMENT(NAME "Улица", rec.улица),

XMLELEMENT(NAME "Номер\_дома", rec.дом),

XMLELEMENT(NAME "Номер\_квартиры", rec.квартира)

)

)

);

END LOOP;

END;

$$;

1. Запросы
   1. Адреса объектов в указанном районе

SELECT улица, дом, квартира

FROM адреса

WHERE район = 'Указанный район'

ORDER BY район;

* 1. Адреса объектов в указанной области с жилой площадью ≥ 80% общей площади

SELECT город, улица, дом, квартира

FROM адреса addr

JOIN объекты\_недвижимости obj ON addr.код\_объекта = obj.код\_объекта

WHERE область = 'Указанная область' AND жилая\_площадь >= 0.8 \* общая\_площадь;

* 1. Названия районов, адреса объектов, рейтинг

SELECT район, CONCAT(город, ', ', улица, ', ', дом, ', ', квартира) AS адрес, рейтинг

FROM адреса addr

JOIN объекты\_недвижимости obj ON addr.код\_объекта = obj.код\_объекта

ORDER BY район;

* 1. Динамика изменения стоимости

SELECT код\_объекта, новая\_стоимость, дата\_изменения

FROM Динамика\_цен

WHERE код\_объекта = 1

1. Запросы, используя таблицу ОПИСАНИЕ\_ОБЪЕКТА\_XML
   1. Адреса объектов в указанном районе

SELECT

описание::XML->'Адрес'->>'Улица' AS улица,

описание::XML->'Адрес'->>'Номер дома' AS дом,

описание::XML->'Адрес'->>'Номер квартиры' AS квартира

FROM ОПИСАНИЕ\_ОБЪЕКТА\_XML

WHERE описание::XML->'Адрес'->>'Название района' = 'Указанный район';

* 1. Адреса объектов в указанной области с жилой площадью ≥ 80% общей площади

SELECT

описание::XML->'Адрес'->>'Город' AS город,

описание::XML->'Адрес'->>'Улица' AS улица,

описание::XML->'Адрес'->>'Номер дома' AS дом,

описание::XML->'Адрес'->>'Номер квартиры' AS квартира

FROM ОПИСАНИЕ\_ОБЪЕКТА\_XML

WHERE описание::XML->'Адрес'->>'Область' = 'Указанная область'

AND (CAST(описание::XML->'Площадь'->>'Жилая площадь' AS NUMERIC) >=

0.8 \* CAST(описание::XML->'Площадь'->>'Общая площадь' AS NUMERIC));

* 1. Названия районов, адреса, рейтинг

SELECT

описание::XML->'Адрес'->>'Название района' AS район,

описание::XML->'Адрес'->>'Улица' || ', ' || описание::XML->'Адрес'->>'Номер дома' AS адрес,

описание::XML->>'Рейтинг' AS рейтинг

FROM ОПИСАНИЕ\_ОБЪЕКТА\_XML;

* 1. Динамика изменения стоимости

SELECT

UNNEST(XPATH('//Стоимость', описание))::TEXT AS стоимость,

UNNEST(XPATH('//Стоимость/@Дата установки стоимости', описание))::TEXT AS дата

FROM ОПИСАНИЕ\_ОБЪЕКТА\_XML

WHERE код\_объекта = 1;

1. Процедура добавления ФИО собственника

CREATE OR REPLACE PROCEDURE добавить\_собственника(код INTEGER, фио TEXT)

LANGUAGE plpgsql

AS $$

BEGIN

UPDATE объекты\_недвижимости

SET собственник = фио

WHERE код\_объекта = код;

END;

$$;