Результаты выполнения задания №1.1

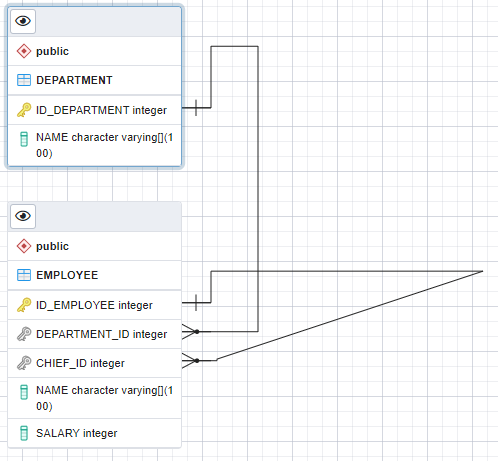


Рисунок 1 - Схема созданных таблиц

Обе таблицы были заполнены следующими данными.

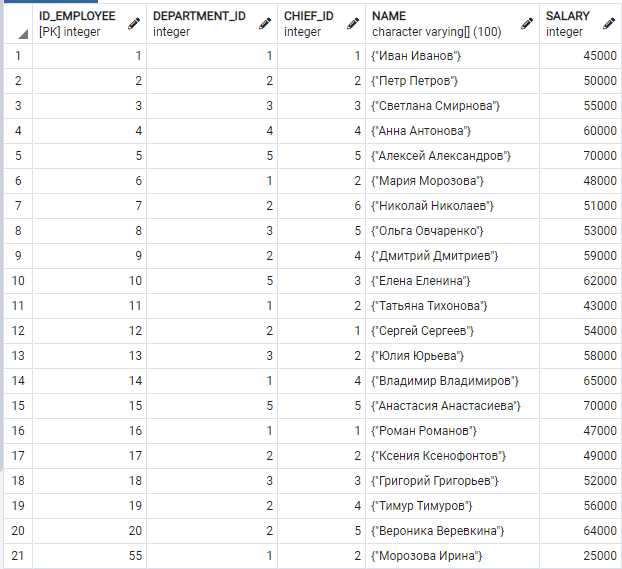


Рисунок 2 – Данные таблицы EMPLOYEE

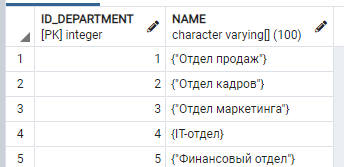


Рисунок 3 – Данные таблицы DEPARTMENT

Запросы:

1. Вывести список сотрудников, которые получают заработную плату ниже, чем у непосредственного руководителя.

SELECT e."ID\_EMPLOYEE",

e."DEPARTMENT\_ID",

e."CHIEF\_ID",

e."NAME",

e."SALARY",

c."SALARY" AS CHIEF\_SALARY

FROM public."EMPLOYEE" e

LEFT JOIN public."EMPLOYEE" c ON e."CHIEF\_ID" = c."ID\_EMPLOYEE"

WHERE e."SALARY" < COALESCE(c."SALARY", e."SALARY");



Рисунок 4 – Результат выполнение запроса 1

1. Вывести список сотрудников, которые получают в отделе минимальную заработную плату в своем отделе.

SELECT e."ID\_EMPLOYEE",

e."DEPARTMENT\_ID",

e."CHIEF\_ID",

e."NAME",

e."SALARY"

FROM public."EMPLOYEE" e

WHERE e."SALARY" = (Select min("SALARY") from public."EMPLOYEE" Where "DEPARTMENT\_ID" = e."DEPARTMENT\_ID");

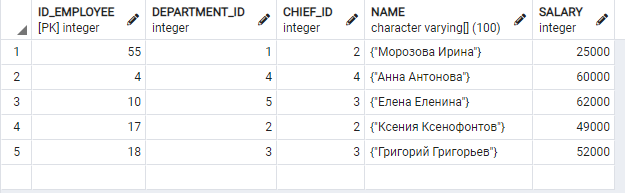


Рисунок 5 – Результат выполнение запроса 2

1. Вывести список ID отделов, количество сотрудников в которых не превышает трех человек.

SELECT "DEPARTMENT\_ID"

FROM public."EMPLOYEE"

GROUP BY "DEPARTMENT\_ID"

HAVING COUNT("ID\_EMPLOYEE") <= 3;

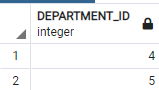


Рисунок 5 – Результат выполнение запроса 3

1. Вывести список сотрудников, не имеющих назначенного руководителя, который работал бы в том же отделе.

SELECT e."ID\_EMPLOYEE", e."DEPARTMENT\_ID", e."CHIEF\_ID", e."NAME", e."SALARY"

FROM "EMPLOYEE" e

LEFT JOIN "EMPLOYEE" c ON e."CHIEF\_ID" = c."ID\_EMPLOYEE"

WHERE e."CHIEF\_ID" IS NULL OR e."DEPARTMENT\_ID" != c."DEPARTMENT\_ID";



Рисунок 5 – Результат выполнение запроса 4

1. Найти список ID отделов с максимальной суммарной заработной платой сотрудников.

SELECT "DEPARTMENT\_ID"

FROM "EMPLOYEE"

GROUP BY "DEPARTMENT\_ID"

HAVING SUM("SALARY") = (

SELECT MAX(TotalSalary)

FROM (

SELECT SUM("SALARY") AS TotalSalary

FROM "EMPLOYEE"

GROUP BY "DEPARTMENT\_ID"

) AS MaxTotalSalary

)



Рисунок 6 – Результат выполнение запроса 5

1. Составить SQL-запрос, вычисляющий сумму всех значений всех ЗП в конкретном столбце таблицы.

SELECT SUM("SALARY") AS TOTALSUM

FROM "EMPLOYEE"

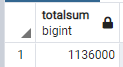


Рисунок 7 – Результат выполнение запроса 6

Реализации хранимой процедуры UPDATESALARYFORDEPARTMENT.

Входные данные: ID отдела, PERCENT процент повышения ЗП

Логика: данная процедура должна у всех сотрудников в рамках отдела с заданным ID (кроме начальника отдела) повышать ЗП на заданный процент (PERCENT). В случае, если после повышения ЗП у кого-либо из сотрудников ЗП оказалась выше, чем у начальника отдела, то повысить ЗП для начальника до аналогичной ЗП. На выходе вернуть перечень сотрудников (все данные из таблицы employee) с обновленной и старой ЗП.

Примем, что начальником отдела является сотрудник, имеющий CHIEF\_ID = ID\_Employee.

Код реализации:

DROP PROCEDURE IF EXISTS public."UPDATESALARYFORDEPARTMENT"(integer, numeric);

CREATE OR REPLACE PROCEDURE public."UPDATESALARYFORDEPARTMENT"(

IN "p\_id" integer DEFAULT 0,

IN "p\_percent" numeric DEFAULT 0)

LANGUAGE 'plpgsql'

AS $BODY$

BEGIN

DROP TABLE IF EXISTS "TEMPEMPLOYEE";

-- Создание временной таблицы

CREATE TEMPORARY TABLE "TEMPEMPLOYEE"

(

"ID\_EMPLOYEE" int,

"DEPARTMENT\_ID" int,

"CHIEF\_ID" int,

"NAME" varchar(100),

"NEW\_SALARY" int,

"OLD\_SALARY" int

);

INSERT INTO "TEMPEMPLOYEE" ("ID\_EMPLOYEE", "DEPARTMENT\_ID", "CHIEF\_ID", "NAME", "NEW\_SALARY", "OLD\_SALARY")

SELECT

"ID\_EMPLOYEE",

"DEPARTMENT\_ID",

"CHIEF\_ID",

"NAME",

"SALARY" AS "NEW\_SALARY",

"SALARY" AS "OLD\_SALARY"

FROM

"EMPLOYEE"

WHERE

"DEPARTMENT\_ID" = p\_id;

-- Обновление зарплат сотрудников

UPDATE "TEMPEMPLOYEE"

SET "NEW\_SALARY" = "NEW\_SALARY" \* (1 + p\_percent / 100)

WHERE "ID\_EMPLOYEE" != "CHIEF\_ID";

-- Зарплата начальника

DO $$

DECLARE

CHIEF\_Salary int;

MAX\_New\_Salary int;

BEGIN

-- Получение зарплаты начальника

SELECT "OLD\_SALARY" INTO CHIEF\_Salary

FROM "TEMPEMPLOYEE"

WHERE "ID\_EMPLOYEE" = (SELECT "CHIEF\_ID" FROM "TEMPEMPLOYEE" LIMIT 1);

-- Получение наибольшей зарплаты среди сотрудников

SELECT MAX("NEW\_SALARY") INTO MAX\_New\_Salary

FROM "TEMPEMPLOYEE"

WHERE "ID\_EMPLOYEE" != (SELECT "CHIEF\_ID" FROM "TEMPEMPLOYEE" LIMIT 1);

-- Обновление зарплаты начальника, если это необходимо

IF CHIEF\_Salary < MAX\_New\_Salary THEN

UPDATE "TEMPEMPLOYEE"

SET "NEW\_SALARY" = MAX\_New\_Salary

WHERE "ID\_EMPLOYEE" = (SELECT "CHIEF\_ID" FROM "TEMPEMPLOYEE" LIMIT 1);

END IF;

END $$;

-- Обновление зарплат сотрудников в основной таблице

UPDATE "EMPLOYEE"

SET "SALARY" = "TEMPEMPLOYEE"."NEW\_SALARY"

FROM "TEMPEMPLOYEE"

WHERE "EMPLOYEE"."ID\_EMPLOYEE" = "TEMPEMPLOYEE"."ID\_EMPLOYEE";

END;

$BODY$;

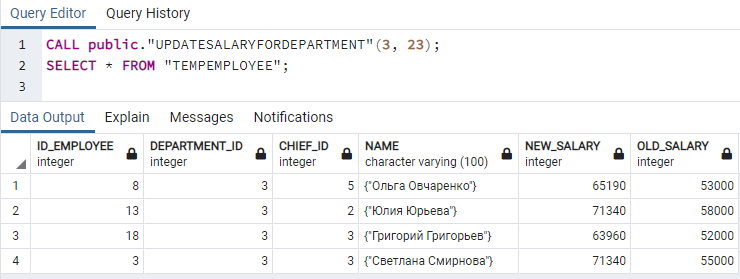


Рисунок 8 – Пример выполнения процедуры



Рисунок 9 –Созданная процедура