ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»  
(РУТ (МИИТ))

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

ОТЧЁТ  
О ЛАБОРАТОРНОЙ РАБОТЕ №4

По дисциплине «Основы построения защищенных баз данных»

ВАРИАНТ 19

Выполнил: ст. гр. ТКИ-441

Чекан Фёдор Сергеевич

Проверил: к.т.н., доц. Васильева М. А.

Москва 2024

## Цель работы

## Изучить операторы sql на примере диалекта СУБД Postgres Pro, необходимые для работы с подзапросами. Научиться создавать вложенные и коррелированные запросы. Подготавливать отчет, выполненный согласно требованиям гост 7.32.

## РЕЗУЛЬТАТЫ ВЫПОЛНЕННОЙ РАБОТЫ

Задание 1

1. Формулировка задания:

Показать товары, которые продались быстрее всех (время продажи – минимальное). (Таблица Production.Product). Вывести наименование и время продажи.

1. Текст скрипта:

SELECT "Name", "SellEndDate" - "SellStartDate" as "SellTime"

FROM "Production"."Product"

WHERE "SellEndDate" = (SELECT MIN("SellEndDate") from "Production"."Product" where "SellEndDate" is not null);

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 1).

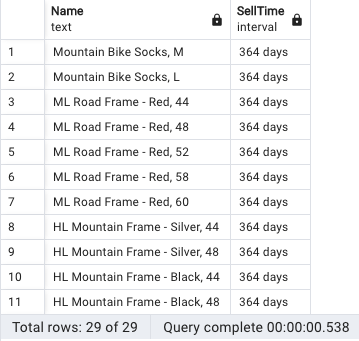


Рисунок 1 − Результат выполнения первого задания

Задание 2

1. Формулировка задания:

Ранжирование товаров в порядке возрастания номера продукта. ("Production"."Product"). Примечание к ранжированию – ранг может начинаться с произвольного числа (например, 2 или 5), главное – ранг не должен уменьшаться.

1. Текст скрипта:

SELECT ep."BusinessEntityID"

, ep."Gender"

, (SELECT COUNT(\*)

FROM "HumanResources"."Employee" ep2

WHERE ep2."Gender" <= ep."Gender") AS "Range"

FROM "HumanResources"."Employee" ep

ORDER BY "Range";

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 2).

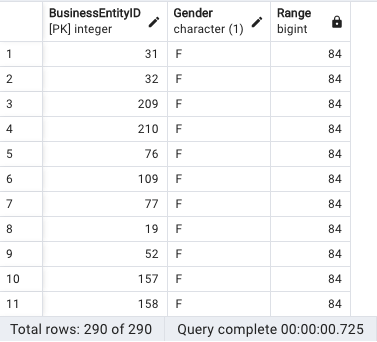


Рисунок 2 − Результат выполнения второго задания

Задание 3

1. Формулировка задания:

Получите список всех ProductId, AverageLeadTime которых меньше, чем максимальный AverageLeadTim чем у продукта с ProducID = 709. Таблица ProductVendor.

1. Текст скрипта:

select pv."ProductID", pv."AverageLeadTime"

from "Purchasing"."ProductVendor" pv

where pv."AverageLeadTime" in

(select max(pv1."AverageLeadTime") from "Purchasing"."ProductVendor" pv1 where pv1."ProductID" = 709);

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 3).

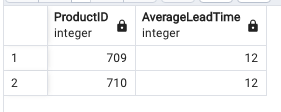


Рисунок 3 − Результат выполнения третьего задания

Задание 4

1. Формулировка задания:

Для каждого из поставщиков выведите его название, а также название следующего по убыванию кредитного рейтинга поставщика ("Purchasing"."Vendor").

1. Текст скрипта:

select v1."Name",

(select v2."Name"

from "Purchasing"."Vendor" v2

where v2."CreditRating" < v1."CreditRating"

order by v2."CreditRating" desc limit 1) as "PreviousName"

from "Purchasing"."Vendor" v1

order by v1."CreditRating" desc;

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 4).

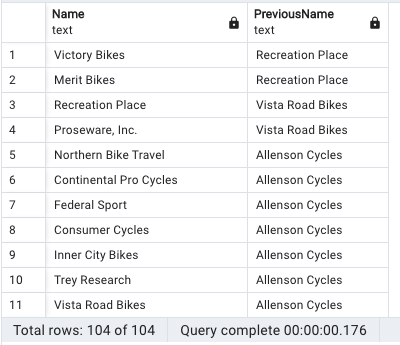


Рисунок 4 − Результат выполнения четвертого задания

Задание 5

1. Формулировка задания:

Показать ProductID товаров, StandardPrice которых меньше средней товаров того же BusinessEntityID (Таблица Purchasing.ProductVendor). Показать поля ProductID, AverageLeadTime и StandardPrice.

1. Текст скрипта:

SELECT p1."ProductID", p1."AverageLeadTime", p1."StandardPrice"

FROM "Purchasing"."ProductVendor" p1

WHERE p1."StandardPrice" IS NOT NULL

AND p1."StandardPrice" <

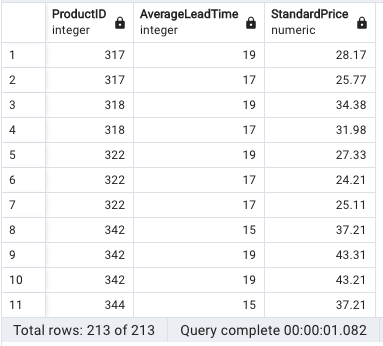
(SELECT AVG(p2."StandardPrice") as "Average"

FROM "Purchasing"."ProductVendor" p2

where p1."BusinessEntityID" = p2."BusinessEntityID");

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 5).



**Рисунок 5 − Результат выполнения пятого задания**

## Отметка о выполнении лабораторной работы в веб-хостинге GitHub

Подтверждение о правильно выполненной работе представлено ниже (Рисунок 6).

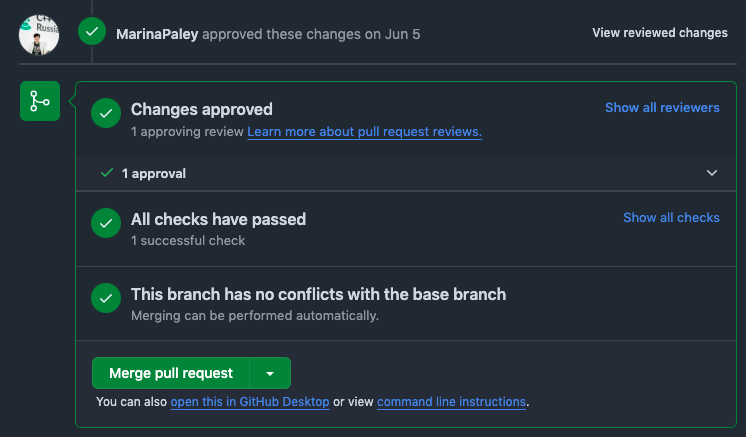


Рисунок 6 − Подтверждение о правильно выполненной работе