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 №3

По дисциплине «Основы построения защищенных баз данных»

ВАРИАНТ 8

Выполнил: ст. гр. ТКИ-442

Зинченко Б.А.

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

Москва 2024

## Цель работы

Изучить операторы SQL диалекта PostgreSQL, необходимые для работы с данными, находящимися в разных таблицах. Научиться создавать запросы на соединение данных из множества таблиц. Отчет по выполненной работе должен быть выполнен с соблюдением ГОСТ по НИР 7.32 [31].

## РЕЗУЛЬТАТЫ ВЫПОЛНЕННОЙ РАБОТЫ

Задание

1. Формулировка задания:

Показать цену за штуку (UnitPrice)и поля Name, ProductNumber из таблиц Production.Product, Sales.SalesOrderDetail.

Текст скрипта:

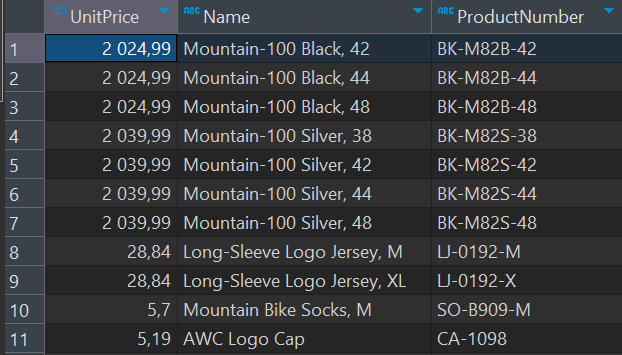
SELECT SOD."UnitPrice", P."Name", P."ProductNumber"

FROM "Production"."Product" P

INNER JOIN "Sales"."SalesOrderDetail" SOD ON P."ProductID" = SOD."ProductID";

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 1).



**Рисунок 1 − Результат выполнения первого задания**

Задание

1. Формулировка задания:

Показать список людей (поля FirstName, LastName), в котором указан уровень человека в организации (OrganizationLevel) или нет, из таблиц Person.Person, HumanResources.Employee, используя LEFT OUTER JOIN.

Текст скрипта:

SELECT p."FirstName", p."LastName", e."OrganizationLevel"

FROM "Person"."Person" p

LEFT OUTER JOIN "HumanResources"."Employee" e ON p."BusinessEntityID" = e."BusinessEntityID";

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 2).

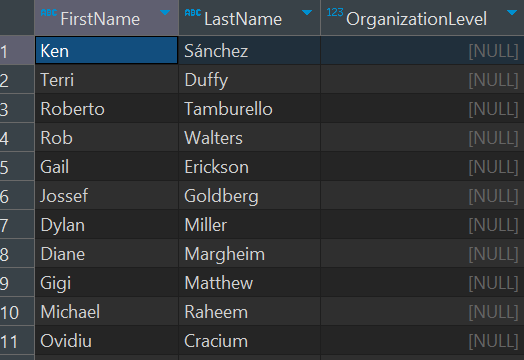


Рисунок  − Результат выполнения второго задания

Задание 3

1. Формулировка задания:

Показать список товаров (поле Name), в котором указан, есть ли у товара дата начала отслеживания цены (StartDate) или нет, из таблицы Production.ProductListPriceHistory, Production.Product, используя RIGHT OUTER JOIN.

Текст скрипта:

SELECT P."Name",

CASE WHEN PLPH."StartDate" IS NULL THEN 'No' ELSE 'Yes' END AS HasStartDate

FROM "Production"."Product" AS P

RIGHT OUTER JOIN "Production"."ProductListPriceHistory" AS PLPH

ON P."ProductID" = PLPH."ProductID";on p."BusinessEntityID" = e."BusinessEntit;

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 3).

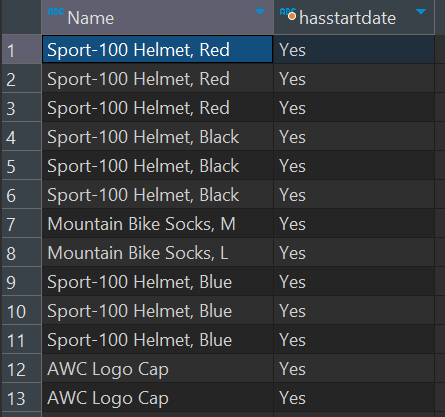


Рисунок 3 − Результат выполнения третьего задания

Задание 4

1. Формулировка задания:

Показать список продуктов (ProductID), которые имеют несколько различных остатков на складах (Quantity), из таблицы Production.ProductInventory, используя SELF JOIN.

1. Текст скрипта:

SELECT DISTINCT PI1."ProductID"

FROM "Production"."ProductInventory" AS PI1

INNER JOIN "Production"."ProductInventory" AS PI2 ON PI1."ProductID" = PI2."ProductID" and

PI1."Quantity" <> PI2."Quantity";

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 4).

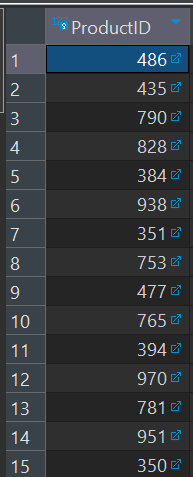


Рисунок 4 − Результат выполнения четвертого задания

Задание 5

1. Формулировка задания:

Показать список остатков на складах (Quantity), которые соответствуют нескольким продуктам (ProductID), из таблиц Production.ProductInventory, используя SELF JOIN.

1. Текст скрипта:

SELECT DISTINCT PI1."ProductID", PI1."Quantity"

FROM "Production"."ProductInventory" AS PI1

INNER JOIN "Production"."ProductInventory" AS PI2 ON PI1."ProductID" = PI2."ProductID"

WHERE PI1."Quantity" <> PI2."Quantity";

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 5).

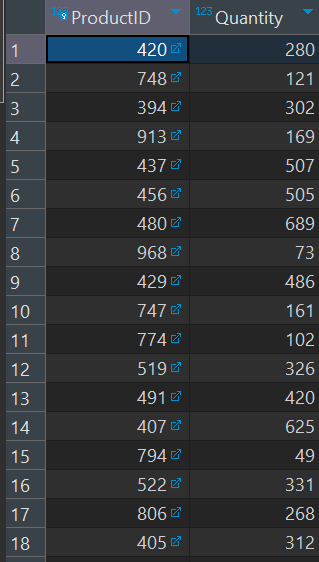


Рисунок 5 − Результат выполнения пятого задания

Задание 6

1. Формулировка задания:

Показать список покупателей (CustomerID), которые имеют несколько

--сумм налога (TaxAmt), из таблицы Sales.SalesOrderHeader,

--используя SELF JOIN. Сумма налога меньше 100

1. Текст скрипта:

SELECT DISTINCT *SOH1*."CustomerID"

FROM "Sales"."SalesOrderHeader" AS *SOH1*

INNER JOIN "Sales"."SalesOrderHeader" AS *SOH2* ON *SOH1*."CustomerID" = *SOH2*."CustomerID"

WHERE *SOH1*."TaxAmt" <> *SOH2*."TaxAmt" AND *SOH1*."TaxAmt" < 100;

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 6).

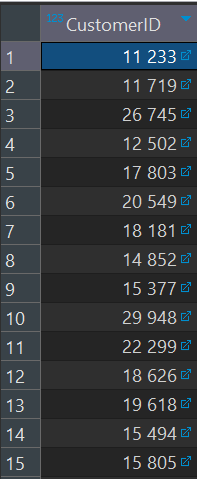


Рисунок 6 − Результат выполнения пятого задания

Задание 7

1. Формулировка задания:

-- Показать комбинированный список таблиц Person.Address, Person.BusinessEntityAddress по

-- полям AddressID, ModifiedDate, используя UNION.

1. Текст скрипта:

-- Выборка из таблицы Person.Address

**SELECT** "AddressID", "ModifiedDate"

**FROM** "Person"."Address"

**UNION**

-- Выборка из таблицы Person.BusinessEntityAddress

**SELECT** "AddressID", "ModifiedDate"

**FROM** "Person"."BusinessEntityAddress";

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 7).

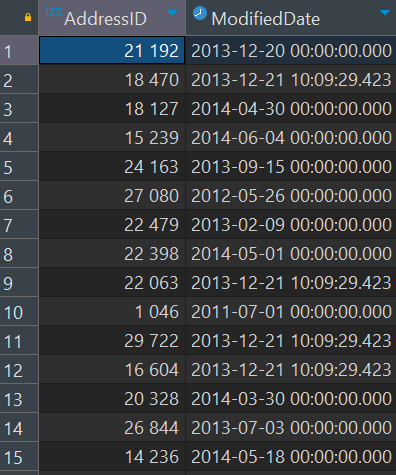


Рисунок 7 − Результат выполнения пятого задания

Задание 8

1. Формулировка задания:

Показать список ProductID, которые содержатся в таблице Production.Product, но не содержатся в таблице Production.WorkOrder.

1. Текст скрипта:

**SELECT** "ProductID"

**FROM** "Production"."Product"

**EXCEPT**

**SELECT** "ProductID"

**FROM** "Production"."WorkOrder";

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 8).

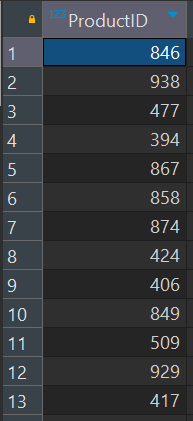


Рисунок 8 − Результат выполнения пятого задания

Задание 9

1. Формулировка задания:

Ограничить результирующий набор, полученный в п.8.Текст скрипта:

SELECT "ProductID"

FROM "Production"."Product"

EXCEPT

SELECT "ProductID"

FROM "Production"."WorkOrder"

LIMIT 10; -- Здесь выбираются только первые 10 записей

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 9).

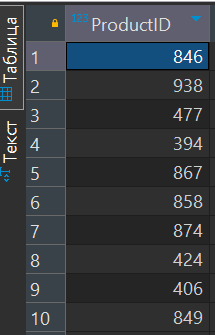


Рисунок 9 − Результат выполнения пятого задания

Задание 10

1. Формулировка задания:

Пропустить 15 строк из результирующего набора, полученного в п.9.

1. Текст скрипта:

**SELECT** "ProductID"

**FROM** "Production"."Product"

**EXCEPT**

**SELECT** "ProductID"

**FROM** "Production"."WorkOrder"

**LIMIT** 10 -- Здесь выбираются только первые 10 записей

OFFSET 15; -- Здесь пропускаются первые 15 записей

1. Результат выполнения скрипта:

Результат выполнения скрипта представлен ниже (Рисунок 10).

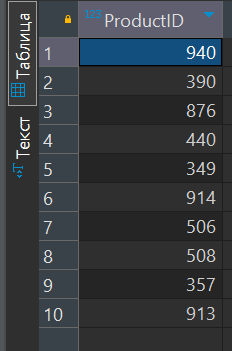


Рисунок 10 − Результат выполнения пятого задания

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

Подтверждение о правильно выполненной работе представлено ниже (Рисунок 11).



Рисунок 11 − Подтверждение о правильно выполненной работе