|  |
| --- |
| МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ  ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  «РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА» (РУТ (МИИТ))  ИНСТИТУТ ТРАНСПОРТНОЙ ТЕХНИКИ И СИСТЕМ УПРАВЛЕНИЯ   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Кафедра «Управление и защита информации» |
|  |
| **ЛАБОРАТОРНАЯ РАБОТА №3**  **по дисциплине «Основы построения защищенных баз данных»**  Вариант №2 |
|  |
|  |
|  |
| Выполнил: ст. гр. ТКИ – 441  Галыба Л.Д. |
| Проверил: к.т.н., доц.,  Васильева М.А. |
|  |
|  |
|  |
| Москва 2024 |

**Оглавление**

[Цель работы 3](#_Toc161776117)

[Выполнение работы 4](#_Toc161776118)

[Задание 1 4](#_Toc161776119)

[Задание 2 4](#_Toc161776120)

[Задание 3 5](#_Toc161776121)

[Задание 4 6](#_Toc161776122)

[Задание 5 7](#_Toc161776123)

[Задание 6 8](#_Toc161776124)

[Задание 7 9](#_Toc161776125)

[Задание 8 10](#_Toc161776126)

[Задание 9 11](#_Toc161776127)

[Задание 10 12](#_Toc161776128)

[Результат проверки преподавателем 14](#_Toc161776129)

[Вывод 15](#_Toc161776130)

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.

Выполнение работы

Задание 1

Показать номера телефонов (PhoneNumber) и поля PersonType, FirstName, LastName из таблиц Person.Person, Person.PersonPhone.

**SELECT**

*p*."PersonType"

, *p*."FirstName"

, *p*."LastName"

, *pp*."PhoneNumber"

**from**

"Person"."Person" *p*

**inner** **join** "Person"."PersonPhone" *pp*

**on** *p*."BusinessEntityID" = *pp*."BusinessEntityID"

|  |
| --- |
|  |
| 1. – Выполнение скрипта №1 |

Задание 2

Показать список продуктов (поле Name), в котором указано, есть ли у продукта название модели или нет, из Production.ProductionModel, Production.Product, используя LEFT OUTER JOIN.

**select**

*p*."Name"

, *pm*."Name"

**from**

"Production"."Product" *p*

**left** **outer** **join** "Production"."ProductModel" *pm*

**on** *p*."ProductModelID" = *pm*."ProductModelID"

|  |
| --- |
|  |
| 1. – Выполнение скрипта №2 |

Задание 3

Показать список ID людей (поле BusinessEntityId), в котором указано, работает ли человек в магазине или нет, из таблиц Sales.Store, Person.BusinessEntity, используя RIGHT OUTER JOIN.

**select**

*be*."BusinessEntityID"

, *s*."Name"

**from**

"Sales"."Store" *s*

**right** **outer** **join** "Person"."BusinessEntity" *be*

**on** *s*."BusinessEntityID" = *be*."BusinessEntityID"

|  |
| --- |
|  |
| 1. – Выполнение скрипта №3 |

Задание 4

Показать список продуктов (ProductID), которые содержатся на нескольких складах (LocationID), из таблицыProduction.ProductInventory, используя SELF JOIN

**select** **distinct** *pi1*."ProductID"

**from**

"Production"."ProductInventory" *pi1*

**inner** **join** "Production"."ProductInventory" *pi2*

**on** *pi1*."ProductID" = *pi2*."ProductID"

**where** *pi1*."LocationID" <> *pi2*."LocationID"

|  |
| --- |
|  |
| 1. – Выполнение скрипта №4 |

Задание 5

Показать список складов (LocationID), на которых содержатся несколько продуктов (ProductID), из таблицы Production.ProductInventory, используя SELF JOIN.

**select** **distinct** *pi1*."LocationID"

**from**

"Production"."ProductInventory" *pi1*

**inner** **join** "Production"."ProductInventory" *pi2*

**on** *pi1*."LocationID" = *pi2*."LocationID"

**where** *pi1*."ProductID" <>*pi2*."ProductID"

|  |
| --- |
|  |
| 1. – Выполнение скрипта №5 |

Задание 6

Показать список продуктов из таблицы Production.Product с таким же цветом как у продукта «ML Road Frame–W – Yellow, 38», используя SELF JOIN.

**select** **distinct** *p*."Name"

,*p*."Color"

**from**

"Production"."Product" *p*

**inner** **join** "Production"."Product" *p2*

**on** *p*."Color" = *p2*."Color"

**where** *p2*."Name" **like** **'ML Road Frame-W - Yellow, 38%'**

|  |
| --- |
|  |
| 1. – Выполнение скрипта №6 |

Задание 7

Показать комбинированный список таблиц Person.AddressType, Person.ContactType по полям ID, Name, ModifiedDate, используя UNION.

**select**

*at2*."AddressTypeID"

, *at2*."Name"

, *at2*."ModifiedDate"

**from**

"Person"."AddressType" *at2*

**union**

**select**

*ct*."ContactTypeID"

, *ct*."Name"

, *ct*."ModifiedDate"

**from**

"Person"."ContactType" *ct*

|  |
| --- |
|  |
| 1. – Выполнение скрипта №7 |

Задание 8

Показать список BusinessEntityID, которые содержатся в таблице Sales.SalesPerson, но не содержатся в таблице Sales.Store.

**select**

*sp*."BusinessEntityID"

**from**

"Sales"."SalesPerson" *sp*

**except**

**select**

*s*."BusinessEntityID"

**from** "Sales"."Store" *s*

|  |
| --- |
|  |
| 1. – Выполнение скрипта №8 |

Задание 9

Ограничить результирующий набор, полученный в п.8.

**select**

*sp*."BusinessEntityID"

**from**

"Sales"."SalesPerson" *sp*

**except**

**select**

*s*."BusinessEntityID"

**from** "Sales"."Store" *s*

**order** **by** "BusinessEntityID"

**limit** 16

|  |
| --- |
|  |
| 1. – Выполнение скрипта №9 |

Задание 10

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

**select**

*sp*."BusinessEntityID"

**from**

"Sales"."SalesPerson" *sp*

**except**

**select**

*s*."BusinessEntityID"

**from** "Sales"."Store" *s*

**order** **by** "BusinessEntityID"

**limit** 16

**offset** 15

|  |
| --- |
|  |
| 1. – Выполнение скрипта №10 |

Результат проверки преподавателем

|  |
| --- |
|  |
| 1. – Подтверждение от преподавателя |

Вывод

В результате выполнения лабораторной работы были изучены операторы SQL диалекта PostgreSQL, необходимые для работы с данными, находящимися в разных таблицах. Получен навык создания запросов на соединение данных из множества таблиц.