|  |  |
| --- | --- |
| 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 | |
|  |  |
|  |  |
| **«Основы построения защищенных баз данных»** | |
|  |  |
|  |  |
|  | Выполнил: ст. гр. ТКИ-442 |
|  | Белов С.В. |
|  | Вариант №1 |
|  | Проверил: доц. Васильева М. А. |
|  |  |
|  |  |
| Москва – 2024г. | |

[Введение 2](#_Toc132108495)

[1. Цель работы 2](#_Toc132108496)

[2. Описание задачи 2](#_Toc132108497)

[3. Содержательная часть 3](#_Toc132108498)

Вывод 6

Введение

1. Цель работы

Выполнить первую лабораторную работу, получить первичный опыт при работе с SQL запросами.

1. Описание задачи
2. Показать адреса электронной почты (EmailAddress) и поля PersonType, FirstName, LastName из таблиц Person.Person, Person.EmailAddress.
3. Показать список продуктов (поле Name), в котором указано, есть ли у продукта действительная цена (ActualCost) или нет, из таблиц Production.Product, Production.WorkOrderRouting, используя LEFT OUTER JOIN.
4. Показать список продуктов (поле Name), в котором указано, есть ли у продукта подкатегория или нет, из таблиц Production.ProductSubcategory, Production.Product, используя RIGHT OUTER JOIN.
5. Показать список подкатегорий продуктов (ProductSubcategoryID), которые имеют несколько нормативных цен (StandardCost), из таблицы Production.Product, используя SELF JOIN.
6. Показать список нормативных цен (StandardCost), которые имеют несколько подкатегорий продуктов (ProductSubcategoryID), из таблицы Production.Product, используя SELF JOIN.
7. Показать список типов спецпредложений (Type), которые имеют несколько значений минимального количества (MinQty) не меньше 15, из таблицы Sales.SpecialOffer, используя SELF JOIN.
8. Показать комбинированный список таблиц Person.BusinessEntityAddress, Person.BusinessEntityContact, используя UNION.
9. Показать список BusinessEntityID, которые содержатся в таблице Sales.SalesPerson, но не содержатся в таблице Sales.Store.
10. Экспериментальная часть

Листинг задачи 1:

--1. Показать адреса электронной почты (EmailAddress) и поля PersonType, FirstName, LastName из таблиц Person.Person,

--Person.EmailAddress.

SELECT

"EmailAddress",

P."PersonType",

P."FirstName",

P."LastName"

FROM

"Person"."Person" AS P

JOIN

"Person"."EmailAddress" AS E ON P."BusinessEntityID" = E."BusinessEntityID";

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 1 |

Листинг задачи 2:

--2. Показать список продуктов (поле Name), в котором указано, есть ли у продукта действительная цена (ActualCost) или нет, из таблиц

--Production.Product, Production.WorkOrderRouting, используя LEFT OUTER JOIN.

SELECT

P."Name" AS ProductName,

CASE

WHEN WO."ActualCost" IS NOT NULL THEN 'Есть действительная цена'

ELSE 'Нет действительной цены'

END AS CostStatus

FROM

"Production"."Product" AS P

LEFT OUTER JOIN

"Production"."WorkOrderRouting" AS WO ON P."ProductID" = WO."ProductID";

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 2 |

Листинг задачи 3:

--3. Показать список продуктов (поле Name), в котором указано, есть ли у продукта подкатегория или нет, из таблиц

--Production.ProductSubcategory, Production.Product, используя RIGHT OUTER JOIN.

SELECT

P."Name" AS ProductName,

CASE

WHEN PS."ProductSubcategoryID" IS NOT NULL THEN 'Есть подкатегория'

ELSE 'Нет подкатегории'

END AS SubcategoryStatus

FROM

"Production"."Product" AS P

RIGHT OUTER JOIN

"Production"."ProductSubcategory" AS PS ON P."ProductSubcategoryID" = PS."ProductSubcategoryID";

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 3 |

Листинг задачи 4:

--4. Показать список подкатегорий продуктов (ProductSubcategoryID), которые имеют несколько нормативных цен (StandardCost), из таблицы

--Production.Product, используя SELF JOIN.

SELECT

DISTINCT P1."ProductSubcategoryID",

COUNT(P1."StandardCost") AS NumberOfStandardCost

FROM

"Production"."Product" AS P1

JOIN

"Production"."Product" AS P2 ON P1."ProductSubcategoryID" = P2."ProductSubcategoryID"

WHERE

P1."StandardCost" <> P2."StandardCost"

GROUP BY

P1."ProductSubcategoryID"

HAVING

COUNT(P1."StandardCost") > 1;

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 4 |

Листинг задачи 5:

--5. Показать список нормативных цен (StandardCost), которые имеют несколько подкатегорий продуктов (ProductSubcategoryID),

--из таблицы Production.Product, используя SELF JOIN.

SELECT

DISTINCT P1."StandardCost",

COUNT(P1."ProductSubcategoryID") AS NumberOfProductSubcategories

FROM

"Production"."Product" AS P1

JOIN

"Production"."Product" AS P2 ON P1."StandardCost" = P2."StandardCost"

WHERE

P1."ProductSubcategoryID" <> P2."ProductSubcategoryID"

GROUP BY

P1."StandardCost"

HAVING

COUNT(P1."ProductSubcategoryID") > 1;

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 5 |

Листинг задачи 6:

--6. Показать список типов спецпредложений (Type), которые имеют несколько значений минимального количества (MinQty) не меньше 15,

--из таблицы Sales.SpecialOffer, используя SELF JOIN.

SELECT

DISTINCT S1."Type",

COUNT(S1."MinQty") AS NumberOfMinQtys

FROM

"Sales"."SpecialOffer" AS S1

JOIN

"Sales"."SpecialOffer" AS S2 ON S1."Type" = S2."Type"

WHERE

S1."MinQty" >= 15 AND S1."MinQty" <> S2."MinQty"

GROUP BY

S1."Type"

HAVING

COUNT(S1."MinQty") > 1;

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 6 |

Листинг задачи 7:

--7. Показать комбинированный список таблиц Person.BusinessEntityAddress, Person.BusinessEntityContact, используя UNION

SELECT \* FROM "Person"."BusinessEntityAddress"

UNION

SELECT \* FROM "Person"."BusinessEntityContact";

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 7 |

Листинг задачи 8:

--8. Показать список BusinessEntityID, которые содержатся в таблице Sales.SalesPerson, но не содержатся в таблице Sales.Store.

SELECT "BusinessEntityID"

FROM "Sales"."SalesPerson"

EXCEPT

SELECT "BusinessEntityID"

FROM "Sales"."Store";

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 8 |

Вывод

По ходу выполнения лабораторной работы, были получены первичные знания по работе с SQL запросами. Был получен опыт работы с таблицами. Освоены основные конструкции SQL.