|  |  |
| --- | --- |
| 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 |
|  | Пшеницын |
|  | Вариант №10 |
|  | 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

1) Показать hash пароля пользователей (PasswordHash) и поля FirstName, LastName из таблиц Person.Password, Person.Person.

2) Показать список продуктов (поле Name), в котором указано, есть ли у продукта количество списаний или нет, из таблиц Production.Product, Production.WorkOrder, используя LEFT OUTER JOIN.

3) Показать список людей (поля FirstName, LastName), в котором отражается, указан у человека пол или нет, из таблиц HumanResources.Employee, Person.Person, используя RIGHT OUTER JOIN.

4) Показать список покупателей (CustomerID), которые имеют несколько сумм налога (TaxAmt), из таблицы Sales.SalesOrderHeader, используя SELF JOIN.

5) Показать список сумм налога (TaxAmt), к которым относятся несколько покупателей (CustomerID), из таблицы Sales.SalesOrderHeader, используя SELF JOIN.

6) Показать список подкатегорий продуктов (ProductSubcategoryID), которые имеют несколько цен реализации (ListPrice) меньше 300, из таблицы Production.Product, используя SELF JOIN.

7) Показать комбинированный список таблиц Production.Product по полям Name, SizeUnitMeasureCode, Production.UnitMeasure по полям Name, UnitMeasureCode, используя UNION.

8) Показать список BusinessEntityID, которые содержатся и в таблице Person.Person, и в таблице HumanResources.Employee.

9) Ограничить результирующий набор, полученный в п.8.

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

1. Экспериментальная часть

Листинг задачи 1:

Показать hash пароля пользователей (PasswordHash) и поля FirstName, LastName из таблиц Person.Password, Person.Person.

SELECT

P."PasswordHash",

P2."FirstName",

P2."LastName"

FROM "Person"."Password" AS P

JOIN

"Person"."Person" AS P2

ON P."BusinessEntityID" = P2."BusinessEntityID";

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

Листинг задачи 2:

Показать список продуктов (поле Name), в котором указано, есть ли у продукта количество списаний или нет, из таблиц Production.Product, Production.WorkOrder, используя LEFT OUTER JOIN.

SELECT

P."Name",

COALESCE(WO."QtyScrapped", 0) AS "ScrapCount"

FROM

"Production"."Product" P

LEFT OUTER JOIN

(SELECT

WO."ProductID",

COUNT(WO."ScrapReasonID") AS "QtyScrapped"

FROM

"Production"."WorkOrder" WO

WHERE

WO."ScrapReasonID" IS NOT NULL

GROUP BY

WO."ProductID") WO

ON

P."ProductID" = WO."ProductID"

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

Листинг задачи 3:

Показать список людей (поля FirstName, LastName), в котором отражается, указан у человека пол или нет, из таблиц HumanResources.Employee, Person.Person, используя RIGHT OUTER JOIN.

SELECT

P."FirstName",

P."LastName",

E."Gender"

FROM

"HumanResources"."Employee" E

RIGHT OUTER JOIN

"Person"."Person" P ON E."BusinessEntityID" = P."BusinessEntityID"

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

Листинг задачи 4:

Показать список покупателей (CustomerID), которые имеют несколько сумм налога (TaxAmt), из таблицы Sales.SalesOrderHeader, используя SELF JOIN.

SELECT DISTINCT P."CustomerID"

FROM "Sales"."SalesOrderHeader" P

INNER JOIN "Sales"."SalesOrderHeader" P2

ON P."CustomerID" = P2."CustomerID" AND P."SalesOrderID" <> P2."SalesOrderID"

WHERE P."TaxAmt" != P2."TaxAmt"

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

Листинг задачи 5:

Показать список сумм налога (TaxAmt), к которым относятся несколько покупателей (CustomerID), из таблицы Sales.SalesOrderHeader, используя SELF JOIN

SELECT DISTINCT P."TaxAmt"

FROM "Sales"."SalesOrderHeader" P

INNER JOIN "Sales"."SalesOrderHeader" P2

ON P."CustomerID" = P2."CustomerID" AND P."SalesOrderID" <> P2."SalesOrderID"

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

Листинг задачи 6:

Показать список подкатегорий продуктов (ProductSubcategoryID), которые имеют несколько цен реализации (ListPrice) меньше 300, из таблицы Production.Product, используя SELF JOIN.

SELECT DISTINCT P1."ProductSubcategoryID"

FROM "Production"."Product" P1

INNER JOIN "Production"."Product" P2 ON P1."ProductSubcategoryID" = P2."ProductSubcategoryID" AND

P1."ListPrice" < 300 AND P2."ListPrice" < 300

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

Листинг задачи 7:

Показать комбинированный список таблиц Production.Product по полям Name, SizeUnitMeasureCode, Production.UnitMeasure по полям Name, UnitMeasureCode, используя UNION

SELECT "Name", "SizeUnitMeasureCode"

FROM "Production"."Product"

UNION

SELECT "Name", "UnitMeasureCode"

FROM "Production"."UnitMeasure";

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

Листинг задачи 8:

Показать список BusinessEntityID, которые содержатся и в таблице Person.Person, и в таблице HumanResources.Employee.

SELECT "BusinessEntityID"

FROM "Person"."Person"

INTERSECT

SELECT "BusinessEntityID"

FROM "HumanResources"."Employee";

|  |
| --- |
|  |
| 1. Визуализация выполненного запроса к задаче 8   Ограничить результирующий набор, полученный в п.8.  SELECT "BusinessEntityID"  FROM "Person"."Person"  INTERSECT  SELECT "BusinessEntityID"  FROM "HumanResources"."Employee"  limit 10;    Пропустить 15 строк из результирующего набора, полученного в п.9.  SELECT "BusinessEntityID"  FROM "Person"."Person"  INTERSECT  SELECT "BusinessEntityID"  FROM "HumanResources"."Employee"  OFFSET 5;    Было => Стало |

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.