Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Кафедра информатики

Отчет по лабораторной работе №6

«Построение рекурсивных SQL запросов»

Выполнила: студентка гр. 953501

Голубович Ю.И.

Поверил: Чащин С.В.

Минск 2021

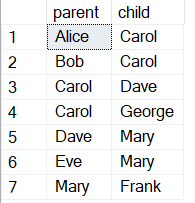
**Цель работы**: получить практические навыки построения рекурсивных запросов.

**Задача 1**. Построить и заполнить таблицу

create table ParentOf(parent text, child text);  
  
insert into ParentOf values ('Alice', 'Carol');  
insert into ParentOf values ('Bob', 'Carol');  
insert into ParentOf values ('Carol', 'Dave');  
insert into ParentOf values ('Carol', 'George');  
insert into ParentOf values ('Dave', 'Mary');  
insert into ParentOf values ('Eve', 'Mary');  
insert into ParentOf values ('Mary', 'Frank');

SELECT \* FROM ParentOf

GO



Используя рекурсивные запросы найти:

1. Предков Mary;

WITH Mary\_CTE(parent, child)

AS

(

SELECT parent, child

FROM ParentOf

WHERE CONVERT(VARCHAR, child) = 'Mary'

UNION ALL

SELECT P.parent, P.child

FROM ParentOf AS P

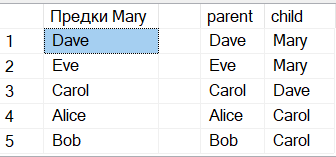
JOIN Mary\_CTE AS CTE ON CONVERT(VARCHAR, P.child) = CONVERT(VARCHAR, CTE.parent)

)

SELECT parent AS 'Предки Mary', ' ' AS ' ', parent, child

FROM Mary\_CTE

GO



1. Потомков Carol.

WITH Carol\_CTE(parent, child)

AS

(

SELECT parent, child

FROM ParentOf

WHERE CONVERT(VARCHAR, parent) = 'Carol'

UNION ALL

SELECT P.parent, P.child

FROM ParentOf AS P

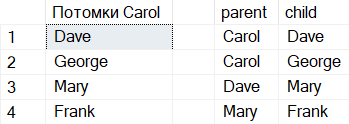
JOIN Carol\_CTE AS CTE ON CONVERT(VARCHAR, CTE.child) = CONVERT(VARCHAR, P.parent)

)

SELECT child AS 'Потомки Carol', ' ' AS ' ', parent, child

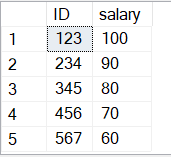
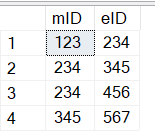
FROM Carol\_CTE

GO



**Задача 2**. Построить и заполнить таблицы

create table Employee(ID int, salary int); --ID работников и их зарплаты  
create table Manager(mID int, eID int); -- ID менеджера, ID работника  
create table Project(name text, mgrID int); -- название проекта и ID руководителя  
  
insert into Employee values (123, 100);  
insert into Employee values (234, 90);  
insert into Employee values (345, 80);  
insert into Employee values (456, 70);  
insert into Employee values (567, 60);  
insert into Manager values (123, 234);  
insert into Manager values (234, 345);  
insert into Manager values (234, 456);  
insert into Manager values (345, 567);  
insert into Project values ('X', 123);

Используя рекурсивный запрос найти совокупную зарплату на проекте.

WITH TotalSalary\_CTE(mID, eID, salary)

AS

(

SELECT 0, mgrID, salary

FROM Project

JOIN Employee ON mgrID = ID

WHERE CONVERT(VARCHAR, name) = 'X'

UNION ALL

SELECT M.mID, M.eID, E.salary

FROM Manager AS M

JOIN Employee AS E ON E.ID = M.eID

JOIN TotalSalary\_CTE AS CTE ON CTE.eID = M.mID

)

SELECT \*

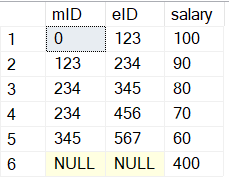
FROM TotalSalary\_CTE

UNION ALL

SELECT NULL, NULL, SUM(salary)

FROM TotalSalary\_CTE

GO



**Задача 3.** Используя рекурсивный запрос, вывести первые10 чисел Фибоначчи

WITH Fib\_CTE(i, a, b, c) AS

(

SELECT i = 1, a = 0, b = 1, c = 0+1

UNION ALL

SELECT i+1, a = b, b = c, c = b+c

FROM Fib\_CTE WHERE i < 10

)

SELECT \*, '' AS ' ', a AS 'п. Фибоначчи'

FROM Fib\_CTE

GO

