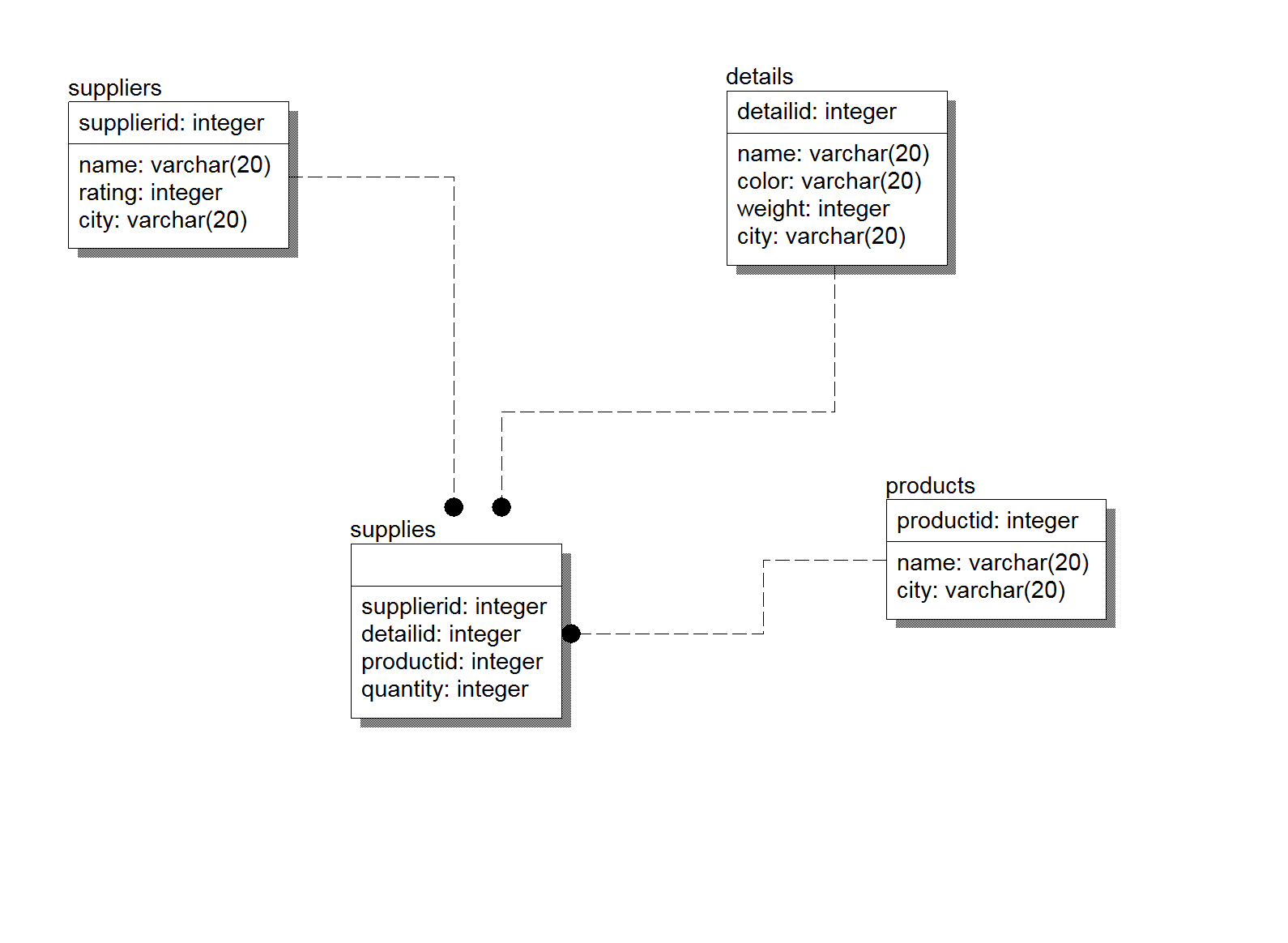
## Домашнє завдання до Модуля № 5

## Схема бази даних

****

## Дані

**Suppliers**

|  |  |  |  |
| --- | --- | --- | --- |
| supplierid | name | rating | city |
| 1 | Smith | 20 | London |
| 2 | Jonth | 10 | Paris |
| 3 | Blacke | 30 | Paris |
| 4 | Clarck | 20 | London |
| 5 | Adams | 30 | Athens |

**Details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| detailid | name | color | weight | city |
| 1 | Screw | Red | 12 | London |
| 2 | Bolt | Green | 17 | Paris |
| 3 | Male-screw | Blue | 17 | Roma |
| 4 | Male-screw | Red | 14 | London |
| 5 | Whell | Blue | 12 | Paris |
| 6 | Bloom | Red | 19 | London |

**Products**

|  |  |  |
| --- | --- | --- |
| productid | name | city |
| 1 | HDD | Paris |
| 2 | Perforator | Roma |
| 3 | Reader | Athens |
| 4 | Printer | Athens |
| 5 | FDD | London |
| 6 | Terminal | Oslo |
| 7 | Ribbon | London |

**supplies**

|  |  |  |  |
| --- | --- | --- | --- |
| supplierid | detailid | productid | quantity |
| 1 | 1 | 1 | 200 |
| 1 | 1 | 4 | 700 |
| 2 | 3 | 1 | 400 |
| 2 | 3 | 2 | 200 |
| 2 | 3 | 3 | 200 |
| 2 | 3 | 4 | 500 |
| 2 | 3 | 5 | 600 |
| 2 | 3 | 6 | 400 |
| 2 | 3 | 7 | 800 |
| 2 | 5 | 2 | 100 |
| 3 | 3 | 1 | 200 |
| 3 | 4 | 2 | 500 |
| 4 | 6 | 3 | 300 |
| 4 | 6 | 7 | 300 |
| 5 | 2 | 2 | 200 |
| 5 | 2 | 4 | 100 |
| 5 | 5 | 5 | 500 |
| 5 | 5 | 7 | 100 |
| 5 | 6 | 2 | 200 |
| 5 | 1 | 4 | 100 |
| 5 | 3 | 4 | 200 |
| 5 | 4 | 4 | 800 |
| 5 | 5 | 4 | 400 |
| 5 | 6 | 4 | 500 |

***CTE***

create table [geography]

(id int not null primary key, name varchar(20), region\_id int);

ALTER TABLE [geography]

ADD CONSTRAINT R\_GB

FOREIGN KEY (region\_id)

REFERENCES [geography] (id);

insert into [geography] values (1, 'Ukraine', null);

insert into [geography] values (2, 'Lviv', 1);

insert into [geography] values (8, 'Brody', 2);

insert into [geography] values (18, 'Gayi', 8);

insert into [geography] values (9, 'Sambir', 2);

insert into [geography] values (17, 'St.Sambir', 9);

insert into [geography] values (10, 'Striy', 2);

insert into [geography] values (11, 'Drogobych', 2);

insert into [geography] values (15, 'Shidnytsja', 11);

insert into [geography] values (16, 'Truskavets', 11);

insert into [geography] values (12, 'Busk', 2);

insert into [geography] values (13, 'Olesko', 12);

insert into [geography] values (30, 'Lvivska st', 13);

insert into [geography] values (14, 'Verbljany', 12);

insert into [geography] values (3, 'Rivne', 1);

insert into [geography] values (19, 'Dubno', 3);

insert into [geography] values (31, 'Lvivska st', 19);

insert into [geography] values (20, 'Zdolbuniv', 3);

insert into [geography] values (4, 'Ivano-Frankivsk', 1);

insert into [geography] values (21, 'Galych', 4);

insert into [geography] values (32, 'Svobody st', 21);

insert into [geography] values (22, 'Kalush', 4);

insert into [geography] values (23, 'Dolyna', 4);

insert into [geography] values (5, 'Kiyv', 1);

insert into [geography] values (24, 'Boryspil', 5);

insert into [geography] values (25, 'Vasylkiv', 5);

insert into [geography] values (6, 'Sumy', 1);

insert into [geography] values (26, 'Shostka', 6);

insert into [geography] values (27, 'Trostjanets', 6);

insert into [geography] values (7, 'Crimea', 1);

insert into [geography] values (28, 'Yalta', 7);

insert into [geography] values (29, 'Sudack', 7);

## Написати запити з використанням під-запитів:

* 1. Отримати номери виробів, для яких всі деталі постачає постачальник 3
  2. Отримати номера і прізвища постачальників, які постачають деталі для якого-небудь виробу з деталлю 1 в кількості більшій, ніж середній об’єм поставок деталі 1 для цього виробу
  3. Отримати повний список деталей для всіх виробів, які виготовляються в Лондоні
  4. Показати номери і назви постачальників, що постачають принаймні одну червону деталь
  5. Показати номери деталей, які використовуються принаймні в одному виробі, який поставляється постачальником 2
  6. Отримати номери виробів, для яких середній об’єм поставки деталей більший за найбільший об’єм поставки будь-якої деталі для виробу 1
  7. Вибрати вироби, що ніколи не постачались (під-запит)

## Написати запити використовуючи CTE або Hierarchical queries

* Написати довільний запит з двома СТЕ (в одному є звертання до іншого)
* Обчислити за допомогою рекурсивної CTE факторіал від 10  та вивести у форматі таблиці з колонками Position та Value :

**Position Value**

10 3628800

* Обчислити за допомогою рекурсивної CTE перші 20 елементів ряду Фібоначчі та вивести у форматі таблиці з колонками Position та Value :

**Position Value**

1 1

2 1

3 2

4 3

5 5

6 8

7 13

8 21

9 34

10 55

11 89

12 144

13 233

14 377

15 610

16 987

17 1597

18 2584

19 4181

20 6765

* Розділити вхідний період 2013-11-25 до 2014-03-05 на періоди по календарним місяцям за допомогою рекурсивної CTE та вивести у форматі таблиці з колонками StartDate та EndDate

**StartDate EndDate**

2013-11-25 2013-11-30

2013-12-01 2013-12-31

2014-01-01 2014-01-31

2014-02-01 2014-02-28

* + 1. 2014-03-05
* Розрахувати календар поточного місяця за допомогою рекурсивної CTE та вивести дні місяця у форматі таблиці з колонками

**Monday,Tuesday,Wednesday,Thursday,Friday,Saturday,Sunday**

Результат для травня 2019 року:

Monday Tuesday Wednesday Thursday Friday Saturday Sunday

NULL NULL 1 2 3 4 5

6 7 8 9 10 11 12

13 14 15 16 17 18 19

20 21 22 23 24 25 26

27 28 29 30 31 NULL NULL

Примітка: звернути увагу на значення системної функції @@DATEFIRST

Підказка: використати функції роботи з датами та

оператор залишку від ділення dividend % divisor

* Створити таблицю **geography** та занести в неї дані
* Написати запит який повертає регіони першого рівня (результат нижче)

regionID place\_ID name PlaceLevel

----------- ----------- -------------------- -----------

1 2 Lviv 1

1 3 Rivne 1

1 4 Ivano-Frankivsk 1

1 5 Kiyv 1

1 6 Sumy 1

1 7 Crimea 1

(6 row(s) affected)

* Написати запит який повертає під-дерево для конкретного регіону (наприклад, Івано-Франківськ). Результат має виглядати наступним чином (колонки можуть називатися інакше)

regionID place\_ID name PlaceLevel

----------- ----------- -------------------- -----------

4 21 Galych 0

4 22 Kalush 0

4 23 Dolyna 0

21 32 Svobody st 1

(4 row(s) affected)

* Написати запит котрий вертає повне дерево від root ('Ukraine') і додаткову колонку, яка вказує на рівень в ієрархії
* Написати запит який повертає дерево для регіону Lviv . Результат має виглядати наступним чином (назви колонок можуть не співпадати):

**Name id region\_id level**

Lviv 2 1 1

Brody 8 2 2

Gayi 18 8 3

Busk 12 2 2

Olesko 13 12 3

Lvivska st 30 13 4

Verbljany 14 12 3

Drogobych 11 2 2

Shidnytsja 15 11 3

Truskavets 16 11 3

Sambir 9 2 2

St.Sambir 17 9 3

Striy 10 2 2

* Написати запит який повертає дерево зі шляхами для регіону Lviv. Результат має виглядати наступним чином (назви колонок можуть не співпадати):

**Name id path**

Lviv 1 /Lviv

Brody 2 /Lviv/Brody

Gayi 3 /Lviv/Brody/Gayi

Sambir 2 /Lviv/Sambir

St.Sambir 3 /Lviv/Sambir/St.Sambir

Striy 2 /Lviv/Striy

Drogobych 2 /Lviv/Drogobych

Shidnytsja 3 /Lviv/Drogobych/Shidnytsja

Truskavets 3 /Lviv/Drogobych/Truskavets

Busk 2 /Lviv/Busk

Olesko 3 /Lviv/Busk/Olesko

Lvivska st 4 /Lviv/Busk/Olesko/Lvivska st

Verbljany 3 /Lviv/Busk/Verbljany

* Написати запит, який повертає дерево зі шляхами і довжиною шляхів для регіону Lviv. Результат має виглядати наступним чином (назви колонок можуть не співпадати):

**Region center pathlen path**

Brody Lviv 1 /Lviv/Brody

Gayi Lviv 2 /Lviv/Brody/Gayi

Sambir Lviv 1 /Lviv/Sambir

St.Sambir Lviv 2 /Lviv/Sambir/St.Sambir

Striy Lviv 1 /Lviv/Striy

Drogobych Lviv 1 /Lviv/Drogobych

Shidnytsja Lviv 2 /Lviv/Drogobych/Shidnytsja

Truskavets Lviv 2 /Lviv/Drogobych/Truskavets

Busk Lviv 1 /Lviv/Busk

Olesko Lviv 2 /Lviv/Busk/Olesko

Lvivska st Lviv 3 /Lviv/Busk/Olesko/Lvivska st

Verbljany Lviv 2 /Lviv/Busk/Verbljany

***Написати запити використовуючи UNION, UNION ALL , EXCEPT, INTERSECT***

* 1. Вибрати постачальників з Лондона або Парижу
  2. Вибрати всі міста, де є постачальники і/або деталі (два запити – перший повертає міста з дублікатами, другий без дублікатів) . Міста у кожному запиті відсортувати в алфавітному порядку
  3. Вибрати всіх постачальників за вийнятком тих, що постачають деталі з Лондона
  4. Знайти різницю між множиною продуктів, які знаходяться в Лондоні та Парижі і множиною продуктів, які знаходяться в Парижі та Римі
  5. Вибрати поставки, що зробив постачальник з Лондона, а також поставки зелених деталей за виключенням поставлених виробів з Парижу (код постачальника, код деталі, код виробу)