

Задача 5

1. Стартирайте SQL Server Management Studio и изберете базата от данни **PubsYYYYYY**.
2. Създайте потребителска функция с име **dbo.emp_name**, която трябва да връща пълното име на служител по зададен параметър **employee id**.

Използвайте интервали между съставните части на името. Ако служителят има само име и фамилия, използвайте само един интервал между тях.

Ако няма открит служител, функцията да връща празен стринг.

```
CREATE FUNCTION dbo.emp_name (@emp_id char(9))
RETURNS varchar(35)
AS
BEGIN
    DECLARE @emp_name varchar(35)
    SELECT @emp_name = fname+' '+mname+' '+lname FROM employee
        WHERE emp_id = @emp_id
    IF @emp_name is null
        SET @emp_name = ''
    ELSE
        SET @emp_name = REPLACE(@emp_name, ' ', ', ')
    RETURN @emp_name
END
```

3. Използвайте създадената функция, за да напишете и изпълните заявки, които да правят следното:
 - Да избират пълното име на служител с id 'ARD36773F'
 - Да избират пълното име на служител с id 'FFFF'
 - Да избират id и пълните имена на служителите, наети на работа не по-късно от 1993 г.

```
SELECT dbo.emp_name('ARD36773F') as fullname

SELECT dbo.emp_name('FFFF') as fullname

SELECT emp_id, dbo.emp_name(emp_id) as fullname
FROM employee
WHERE year(hire_date) <= 1993
```

4. Създайте потребителска функция с име **dbo.au_name**, която трябва да връща пълното име на автор по зададен параметър **author id**.

Използвайте интервал между името и фамилията.

Ако няма открит автор, функцията да връща празен стринг.

5. Използвайте създадената функция, за да напишете и изпълните заявка, която избира id и пълните имена на авторите от щат 'CA', подредени по пълното име.
6. Създайте потребителска функция с име **dbo.avg_job_lvl**, която трябва да връща средната стойност на **job level** по зададен параметър **job description**.

Резултатът трябва да бъде от тип **money**. Преобразувайте типа на колоната **job_lvl** преди изчисляването на средната стойност.

Ако няма открити записи, функцията да връща стойност 0.

```
CREATE FUNCTION dbo.avg_job_lvl (@job_desc varchar(50))
RETURNS money
AS
BEGIN
    DECLARE @avg_job_lvl money
    SELECT @avg_job_lvl = AVG(CAST(e.job_lvl as money)) FROM employee e
        INNER JOIN jobs j ON e.job_id=j.job_id
        WHERE j.job_desc=@job_desc
    IF @avg_job_lvl is null
        SET @avg_job_lvl = 0
    RETURN @avg_job_lvl
END
```

7. Използвайте създадената функция, за да напишете и изпълните заявки, които да правят следното:
 - Изберете средната стойност на job_lvl за длъжността 'Designer'
 - Изберете id, job description и средната стойност на job_lvl за всички длъжности
 - Изберете id на служител, full name, job level и job description за служителите с job level по-висок от средния за длъжността

```
SELECT dbo.avg_job_lvl('Designer') as avg_level

SELECT job_id, job_desc, dbo.avg_job_lvl(job_desc) as avg_level
FROM jobs

SELECT e.emp_id, dbo.emp_name(emp_id) as fullname, e.job_lvl, j.job_desc
FROM employee e INNER JOIN jobs j ON e.job_id=j.job_id
WHERE e.job_lvl > dbo.avg_job_lvl(j.job_desc)
```

8. Създайте потребителска функция с име **dbo.sum_title_sales**, която трябва да връща сумата на стойността на продадените книги (quantity*price) по зададен параметър **title id**.

Резултатът трябва да бъде от тип **money**.

Ако няма открити записи, функцията да връща стойност 0.

9. Използвайте създадената функция, за да напишете и изпълните заявка, която избира title's id, title, price и сумата на стойността на продадените книги.
10. Създайте потребителска функция с име **dbo.sum_title_qty**, която трябва да избере title от таблицата **titles**, store name и state от таблицата **stores** и сумата количествата от таблицата **sales**, групирани по title, store name и state и подредени по title. Функцията трябва да връща само редовете за магазини от щата, който е зададен като параметър.

```
CREATE FUNCTION dbo.title_qty (@state char(2))
RETURNS TABLE
AS
RETURN
(SELECT t.title, st.stor_name, st.state, SUM(s.qty) as quantity_sum
FROM titles t inner join sales s on t.title_id = s.title_id
inner join stores st on s.stor_id = st.stor_id
GROUP BY t.title, st.stor_name, st.state
HAVING st.state=@state)
```

11. Използвайте създадената функция, за да напишете и изпълните заявка, която да избира данни за магазини от щата 'WA'.

```
SELECT * FROM dbo.title_qty ('WA')
```

12. Създайте потребителска функция с име **dbo.avg_job_emp**, която трябва да избере job description, minimal level и maximal level от таблицата **jobs** и средната стойност на job level от таблицата **employee**, групирани по job description, minimal level и maximal level.
13. Използвайте създадената функция, за да напишете и изпълните заявка, която избира данните за всички длъжности.
14. Създайте потребителска функция с име **dbo.emp_data**, която трябва да избере employee id, first name, middle initial, last name и job level от таблицата **employee**, job id и job description от таблицата **jobs**, publisher id и publisher name от таблицата **publishers**. Функцията трябва да връща само редовете за служителите на длъжността, която е зададена като параметър.

```
CREATE FUNCTION dbo.emp_data (@job_desc varchar(50))
RETURNS TABLE
AS
RETURN
(SELECT e.emp_id, e.fname, e.minit, e.lname, e.job_lvl, j.job_id, j.job_desc, p.pub_id, p.pub_name
FROM jobs j INNER JOIN employee e ON j.job_id = e.job_id
INNER JOIN publishers p ON e.pub_id = p.pub_id
WHERE j.job_desc like @job_desc)
```

15. Използвайте създадената функция, за да напишете и изпълните заявка, която избира данните за служителите:

- На длъжност 'Publisher'
- От всички длъжности

```
SELECT * FROM dbo.emp_data ('Publisher')  
  
SELECT * FROM dbo.emp_data ('%')
```

16. Използвайте създадената функция, за да напишете и изпълните заявка, която избира title и publishing date от таблицата **titles**, first name, last name и publisher name от функцията **dbo.emp_data** за книгите, публикувани през 1991 година и и служителите на длъжност 'Editor', подредени по title.

```
SELECT t.title, t.pubdate, e.fname, e.lname, e.pub_name  
FROM titles t INNER JOIN dbo.emp_data ('Editor') e ON t.pub_id=e.pub_id  
WHERE YEAR(t.pubdate) = 1991  
ORDER BY title
```

17. Създайте потребителска функция с име **dbo.sales_data**, която трябва да избере order number, order date и quantity от таблицата **sales**, title id, title и price от таблицата **titles**, store id и store name от таблицата **stores**. Функцията трябва да връща само редовете за продажбите от магазин с id като този, който е зададен като параметър.

18. Използвайте създадената функция, за да напишете и изпълните заявка, която избира данни:

- За продажбите от магазин с id = '7131'
- За продажбите от всички магазини

19. Използвайте създадената функция, за да напишете и изпълните заявка, която избира first name и last name от таблицата **authors**, royalty percent от таблицата **titleauthor**, title и сумата на стойностите (quantity*price) от функцията **dbo.sales_data**, групирани по first name, last name, royalty percent и title, подредени по first name, last name и title.