

## SQL TILIDA FUNKSIYA VA PROTSEDURALARDAN FOYDALANISH

*Tojimatov Isroil Nurmamatovich*

*Farg'ona davlat universiteti o'qituvchi, [israiltojimatov@gmail.com](mailto:israiltojimatov@gmail.com)*

*Botirova Muqaddas Rahmatulla qizi*

*Farg'ona davlat universiteti 2-kurs talabasi, [botirova0711@gmail.com](mailto:botirova0711@gmail.com)*

### Аннотация

Ushbu maqola, SQL tilida funksiyalar va protseduralardan foydalanishning asosiy tushunchalari, turli turlari va ularning ma'lumotlar bazasi bilan ishlashdagi rolini tahlil qiladi. Maqolada SQL tilining standart funksiyalari va foydalanuvchi tomonidan yaratilgan funksiyalar (UDFs) hamda protseduralar (stored procedures) haqida batafsil ma'lumot berilgan. Funksiyalar va protseduralar yaratish, chaqirish va ularning amaliyotdagi qo'llanilishi orqali ma'lumotlar bazasi bilan ishlash samaradorligini oshirish va dasturiy ta'minotni optimallashtirish imkoniyatlari ko'rib chiqilgan. Shuningdek, xatolarni boshqarish, ma'lumotlarni yangilash, hisobotlar yaratish va boshqa ko'plab vazifalar uchun SQL imkoniyatlaridan qanday foydalanish mumkinligi tushuntiriladi.

**Kalit so'zlar:** SQL, funksiyalar, protseduralar, ma'lumotlar bazasi, foydalanuvchi tomonidan yaratilgan funksiyalar (UDFS), stored procedures, ma'lumotlar bazasi optimallashtirish, ma'lumotlar bazasidan ma'lumot olish.

### Аннотация

В данной статье анализируются основные концепции использования функций и процедур языка SQL, их различные типы и их роль в работе с базой данных. В статье представлена подробная информация о стандартных функциях SQL, а также определяемых пользователем функциях (UDF) и процедурах (хранимые процедуры). Рассмотрены возможности повышения эффективности работы с базой данных и оптимизации программного обеспечения за счет создания, вызова и практического их использования функций и процедур. Здесь также объясняется, как использовать возможности SQL для обработки ошибок, обновления данных, создания отчетов и многих других задач.

**Ключевые слова:** SQL, функции, процедуры, база данных, пользовательские функции (UDFS), хранимые процедуры, оптимизация базы данных, извлечение данных из базы данных.

### Annotation

This article analyzes the basic concepts of using SQL functions and procedures, their different types and their role in working with a database. The article provides detailed information about standard SQL functions, as well as user-defined functions (UDFs) and procedures (stored procedures). The possibilities of increasing the

efficiency of working with a database and optimizing software through the creation, calling and practical use of functions and procedures are considered. It also explains how to use SQL's capabilities to handle errors, update data, create reports, and many other tasks.

**Keywords:** SQL, functions, procedures, database, user-defined functions (UDFS), stored procedures, database optimization, retrieving data from a database.

SQL (Structured Query Language) dasturlash tilida funksiya va protseduralardan foydalanish, ma'lumotlar bazasi bilan ishlashda juda muhim ahamiyatga ega. SQL tilining asoslari va uning ma'lumotlar bazasi bilan ishlashdagi roli haqida qisqacha ma'lumot beriladi. Shuningdek, funktsiyalar va protseduralarning asosiy tushunchalari va ularning ahamiyati tushuntiriladi. SQL funktsiyalari ma'lumotlar bazasidagi ma'lumotlarni qayta ishlash va manipulyatsiya qilish uchun ishlatiladi. SQL funktsiyalari ikki asosiy guruhga bo'linadi: standart funktsiyalar va foydalanuvchi tomonidan aniqlangan funktsiyalar (UDFs).

#### Standart SQL funktsiyalari

SQL tilining standart funktsiyalari turli xil maqsadlarda ishlatiladigan ko'plab built-in funktsiyalarni o'z ichiga oladi. Ular quyidagi kategoriyalarga bo'linadi:

##### 1. Skalyar Funktsiyalar:

- Matematik funktsiyalar: ``ABS()``, ``CEIL()``, ``FLOOR()``, ``ROUND()`` va boshqalar. Bu funktsiyalar sonli ma'lumotlar ustida amaliyotlar bajaradi.

- String funktsiyalar: ``CONCAT()``, ``LENGTH()``, ``SUBSTRING()``, ``LOWER()``, ``UPPER()`` va hokazo. Bu funktsiyalar matnli ma'lumotlar ustida turli manipulyatsiyalar bajarish uchun mo'ljallangan.

- San'a va vaqt funktsiyalar: ``CURRENT_DATE()``, ``DATE_PART()``, ``NOW()``, va boshqalar. Bu funktsiyalar sana va vaqt bilan bog'liq operatsiyalarni bajaradi.

##### 2. Agregat Funktsiyalar:

- Bu funktsiyalar ma'lumotlar to'plamidan yagona qiymatni hisoblash uchun ishlatiladi. Masalan, ``COUNT()``, ``SUM()``, ``AVG()``, ``MAX()``, va ``MIN()``.

#### Foydalanuvchi Tomonidan Aniqlangan Funktsiyalar (UDFs)

Foydalanuvchilar o'zlarining maxsus ehtiyojlarini qondirish uchun SQL'da o'z funktsiyalarini yaratishlari mumkin. Bunday funktsiyalar ikki asosiy turga bo'linadi:

##### 1. Skalyar UDFs:

- Bular bitta qiymat qaytaradigan funktsiyalar. Masalan, biror maxsus hisob-kitob yoki mantiqiy tekshiruvlarni bajarish uchun yaratilishi mumkin.

##### 2. Tarkibiy Funktsiyalar (Table-Valued Functions, TVFs):

- Bu funksiyalar jadvallarni qaytaradi va ko'pincha jadvallar ustida qo'shimcha so'rovlar bajarish uchun ishlatiladi. TVFs ko'pincha boshqa SQL so'rovlari ichida chaqirilishi mumkin.

#### SQL Funksiyalarini Yaratish va Chaqirish

SQL'da funksiya yaratish sintaksisi ma'lumotlar bazasi boshqaruv tizimiga (DBMS) qarab farqlanishi mumkin, lekin umumiy ko'rinish quyidagicha bo'ladi:

```
CREATE FUNCTION my_function (param1 TYPE, param2 TYPE, ...)
```

```
RETURNS RETURN_TYPE
```

```
AS
```

```
BEGIN
```

```
-- Funksiya tanasi
```

```
RETURN (expression);
```

```
END;
```

Funksiyani chaqirish juda oddiy:

```
SELECT my_function(arg1, arg2, ...) FROM table_name;
```

SQL funksiyalari, ma'lumotlar bazasida ma'lumotlarni tezkor va samarali qayta ishlash uchun juda muhim vositalardir. Standart funksiyalardan tashqari, foydalanuvchilar o'z ehtiyojlariga mos funksiyalar yaratish orqali SQL'ning imkoniyatlarini yanada kengaytirishlari mumkin. Ushbu vositalar ma'lumotlar bazasini yanada moslashuvchan va kuchli qilishga yordam beradi.

SQL protseduralari yoki "stored procedures" ma'lumotlar bazasida oldindan tayyorlangan va saqlangan kod bloklari bo'lib, ular ma'lum bir vazifani bajarish uchun ishlatiladi. Ular ma'lumotlar bazasi serverida bajariladi va turli xil operatsiyalar uchun ishlatilishi mumkin, masalan, ma'lumotlarni qayta ishlash, hisobotlar yaratish, ma'lumotlar o'zgartirish va boshqa ko'plab vazifalar.

#### SQL protseduralarining afzalliklari

1. Takrorlanadigan Kodni Kamaytirish: Protseduralar kodni ma'lumotlar bazasi serverida saqlash orqali dastur kodini soddalashtiradi va takrorlanishini kamaytiradi.

2. Ishonchlilik va Xavfsizlik: Protseduralar ma'lumotlar bazasi darajasida bajarilgani uchun, ular orqali ma'lumotlarni qayta ishlash va ularga kirishni yanada xavfsiz va nazorat qilinadigan qilish mumkin.

3. Optimallashtirish: Protseduralar server tomonida bajariladi, bu esa tarmoq orqali uzatiladigan ma'lumotlar hajmini kamaytiradi va ilova ish faoliyatini yaxshilaydi.

#### SQL protseduralarini yaratish

SQL protseduralarini yaratish uchun "CREATE PROCEDURE" buyrug'i ishlatiladi. Quyida oddiy SQL protsedurasi yaratishning umumiy sintaksisi keltirilgan:

```
CREATE PROCEDURE procedure_name (param1 datatype, param2 datatype,
...)
AS
BEGIN
    -- Protsedura tanasi
    SET NOCOUNT ON;
    -- SQL buyruqlari
END;
```

Bu yerda `procedure\_name` protsedurani chaqirishda ishlatiladigan nom, `param1`, `param2` esa protseduraga uzatiladigan parametrlar. `NOCOUNT ON` buyrug'i faqat zarur bo'lganda ishlatiladi va u SQL serverdan har bir SQL operatsiyasi uchun qaytariladigan qatorlar soni haqidagi xabarlarni o'chirib qo'yadi.

SQL protseduralarini chaqirish

Mavjud protsedurani chaqirish uchun "EXECUTE" yoki "EXEC" buyrug'idan foydalaniladi. Misol uchun:

```
EXECUTE procedure_name @param1 = value1, @param2 = value2;
```

Bu yerda `procedure\_name` chaqirilayotgan protsedura nomi, va `@param1`, `@param2` esa unga uzatilayotgan parametrlar.

Parametrlar va Qaytish Qiymatlari

Protseduralar parametrlarni qabul qilishi va qiymatlarni qaytarishi mumkin. Parametrlar ma'lumotlar turlari bo'yicha aniqlanadi va protsedura chaqirilganda qiymatlar uzatiladi. Protseduralar, shuningdek, boshqa jadvallar, skalyar qiymatlar yoki hech narsa qaytarishi mumkin.

Xatolarni boshqarish

SQL protseduralari ichida xatolarni boshqarish uchun `TRY...CATCH` bloklaridan foydalaniladi. Bu xatolarni ushlab va ularni munosib tarzda boshqarish imkonini beradi.

SQL protseduralari kuchli vosita bo'lib, ular kodni qayta ishlash, ma'lumotlar bazasini boshqarish va ilovalarni optimallashtirishda muhim rol o'ynaydi. Ular ilovalarni samaraliroq va xavfsizroq qilishga yordam beradi va dastur yozuvchilarga ko'plab dasturiy muammolarni hal qilishda yordam beradi.

SQL funksiyalari va protseduralarini amaliy qo'llash uchun bir necha turli xil vaziyatlar va ularning kod namunalari taqdim etiladi. Ushbu amaliy qo'llanma, ma'lumotlar bazasidan ma'lumot olish, ma'lumotlarni yangilash, hisobotlar tuzish va

ma'lumotlar bazasini boshqarish kabi turli xil vazifalar uchun SQL imkoniyatlarini qanday qo'llash mumkinligini ko'rsatadi.

### 1. Ma'lumotlarni O'qish va Yangilash

#### a. Ma'lumotlarni O'qish

Ma'lumotlar bazasidan ma'lumotlarni o'qish uchun quyidagi SQL so'rovini ishlatamiz:

```
SELECT first_name, last_name, email  
FROM users  
WHERE status = 'active';
```

Bu so'rov `users` jadvalidan faqat faol foydalanuvchilarning ismi, familiyasi va elektron pochta manzillarini tanlab oladi.

#### b. Ma'lumotlarni Yangilash

Foydalanuvchining holatini yangilash uchun quyidagi SQL so'rovini bajarish mumkin:

```
UPDATE users  
SET status = 'inactive'  
WHERE last_login < '2021-01-01';
```

Bu so'rov `users` jadvalidagi oxirgi kirish sanasi 2021-yil 1-yanvardan oldin bo'lgan foydalanuvchilarning holatini "inactive" ga o'zgartiradi.

### 2. Hisobotlar Yaratish

Hisobot yaratish uchun, biz ma'lum bir vaqt oralig'ida ro'y bergan savdolar bo'yicha hisobot tuzamiz. Buning uchun SQL'da agregat funksiyalardan foydalanamiz:

```
SELECT product_id, COUNT(*) AS total_sales, SUM(amount) AS  
total_revenue  
FROM sales  
WHERE sale_date BETWEEN '2022-01-01' AND '2022-12-31'  
GROUP BY product_id  
ORDER BY total_revenue DESC;
```

Bu so'rov 2022-yil davomida har bir mahsulot bo'yicha umumiy savdo soni va umumiy daromadni hisoblaydi va mahsulotlarni umumiy daromad bo'yicha kamayish tartibida saralaydi.

### 3. Ma'lumotlar Bazasini Boshqarish

Ma'lumotlar bazasini optimallashtirish uchun biz SQL protseduralarini yaratishimiz va ishlatishimiz mumkin. Masalan, ma'lumotlar bazasidagi eskirgan yozuvlarni tozalash uchun quyidagi protsedura yaratiladi:

```
CREATE PROCEDURE CleanOldData
AS
BEGIN
    DELETE FROM logs
    WHERE log_date < DATEADD(year, -2, GETDATE());
END;
```

Bu protsedura `logs` jadvalidagi ikki yildan eski yozuvlarni o'chiradi.

#### 4. Ma'lumotlarni Avtomatik Yangilash

Avtomatik yangilash uchun, biz bir protsedurani yaratamiz va uni ma'lum bir vaqtlar oralig'ida avtomatik ravishda ishga tushiriladigan tarzda sozlaymiz:

```
CREATE PROCEDURE UpdateUserStatus
AS
BEGIN
    UPDATE users
    SET status = 'inactive'
    WHERE last_login < DATEADD(month, -6, GETDATE());
END;
```

Bu protsedura oxirgi kirish vaqti olti oydan ko'proq bo'lgan foydalanuvchilarning holatini "inactive"ga o'zgartiradi.

Har bir namuna SQL tilining turli xil imkoniyatlaridan foydalanishni va ma'lumotlar bazasi bilan samarali ishlashni ta'minlaydi. Ushbu amaliy qo'llanma orqali siz SQL'ni turli xil real dunyo vazifalari uchun qanday qo'llash mumkinligini ko'rishingiz mumkin.

Xulosa qilib aytganda, SQL funksiyalar va protseduralarining ahamiyati va ularning ma'lumotlar bazasi bilan ishlashdagi roli haqida umumiy xulosa qilinadi. Shuningdek, kelajakda ushbu texnologiyalardan qanday qilib yanada samarali foydalanish mumkinligi haqida fikr yuritiladi.

Har bir bo'limda mavzuni yanada chuqurroq tushunish uchun aniq misollar va kod parchalari taqdim etiladi. Bu maqola SQL'da funksiya va protseduralardan foydalanishni o'rganishni istagan dasturchilar, ma'lumotlar bazasi administratorlari va IT mutaxassisleri uchun foydali bo'ladi.

**Foydalanilgan adabiyotlar**

1. Celko, J. (2005). "SQL for Smarties: Advanced SQL Programming". Morgan Kaufmann Publishers.
2. Date, C. J. (2012). "SQL and Relational Theory: How to Write Accurate SQL Code". O'Reilly Media.
3. Molinaro, A. (2007). "SQL Cookbook: Query Solutions and Techniques for Database Developers". O'Reilly Media.
4. Tojimamatov, I., & Jo'rayeva, M. (2024). BERILGANLAR TUZILISHINING UCH DARAJALI ARXITEKTURASI. Current approaches and new research in modern sciences, 3(4), 194-197.
5. Tojiddinov, A., Gulsumoy, N., Muntazam, H., & Tojimamatov, I. (2023). BIG DATA. Journal of Integrated Education and Research, 2(3), 35-42.
6. Tojimamatov, I., Mirkomil, M. M., & Saidmurod, S. (2023). BIG DATANING TURLI SOHALARDA QO 'LLANILISHI. ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ, 18(6), 61-65.
7. Ne'matjonov, F. F., Jahongirova, J. J., Murodov, B. S., & Tojimamatov, I. N. (2023). CREATE DATA CUBE WITH MS EXCEL. European International Journal of Multidisciplinary Research and Management Studies, 3(03), 77-86.
8. Tojimamatov, I., & Doniyorbek, A. (2023). KATTA HAJMLI MA'LUMOTLAR AFZALLIKLARI VA KAMCHILIKLARI. ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ, 18(6), 66-70.
9. Kimyonazarova, D., Ne'matjonova, D., Ergasheva, B., & Tojimamatov, I. (2023, March). KATTA MA'LUMOTLAR BILAN ISHLASHDA HADOOP ARXITEKTURASI. In Международная конференция академических наук (Vol. 2, No. 3, pp. 96-99).
10. Nurmamatovich, T. I., & Nozimaxon, E. (2024). Chiqish qatlami vaznlarni sozlash va xatoliklarni teskari tarqalishi algoritmi. ILM FAN XABARNOMASI, 1(1), 29-35.