Abylkas Saginov Karaganda Technical University

Information Technologies and Security Department

**Laboratory work №7**

**Discipline**: Database management systems

**Topic**: Creating Database Stored Procedures

**Prepared by:**

Student of SIB-22-5 group

Batyrkhanov Arman

**Approved by:**

Senior Lecturer

Klyuyeva Ye.G.

**Karaganda 2024**

1. **Selection of vacancies with the condition (total number of units - number of vacancies)**

USE [Staff]

GO

CREATE PROCEDURE [dbo].[ВыбратьСвободныеВакансии]

AS

SELECT \*

FROM Штатное\_расписание

WHERE общее\_количество - количество\_вакансий > 0;

GO

1. **Selection of employees of the i-th division with the j-th type of education**

USE [Staff]

GO

CREATE PROCEDURE [dbo].[ВыборРаботников]

@код\_подразделения INT,

@код\_образования INT

AS

SELECT \*

FROM Работники

WHERE код\_подразделения = @код\_подразделения

AND код\_образования = @код\_образования;

GO

1. **Inserting three new rows into the Job Categories table**

USE [Staff]

GO

CREATE PROCEDURE [dbo].[ВставитьСтрокиКатегорииДолжностей]

@КодКатегории1 INT, @Наименование1 NVARCHAR(255),

@КодКатегории2 INT, @Наименование2 NVARCHAR(255),

@КодКатегории3 INT, @Наименование3 NVARCHAR(255)

AS

DECLARE @i INT = 0

WHILE @i < 3

BEGIN

IF @i = 1

INSERT INTO Категории\_должностей (код\_категории, наименование)

VALUES (@КодКатегории1, @Наименование1);

ELSE IF @i = 2

INSERT INTO Категории\_должностей (код\_категории, наименование)

VALUES (@КодКатегории2, @Наименование2);

ELSE IF @i = 3

INSERT INTO Категории\_должностей (код\_категории, наименование)

VALUES (@КодКатегории3, @Наименование3);

SET @i = @i + 1;

END;

GO

1. **Output of the number of employed staff units of the 1st division**

USE [Staff]

GO

CREATE PROCEDURE [dbo].[ВывестиКоличествоЗанятыхРабочихМест]

@код\_подразделения INT,

@код\_должности INT

AS

SELECT

код\_подразделения,

код\_должности,

общее\_количество - количество\_вакансий AS КоличествоЗанятыхМест

FROM

Штатное\_расписание

WHERE

код\_подразделения = @код\_подразделения

AND код\_должности = @код\_должности;

GO

1. **Delete row in categories**

USE [Staff]

GO

CREATE PROCEDURE [dbo].[УдалитьКатегориюДолжностиПоКоду]

@КодКатегории INT

AS

DELETE FROM Категории\_должностей

WHERE код\_категории = @КодКатегории;

GO

**Control questions**

1. **What is the purpose of stored procedures?**

Stored procedures in MS SQL Server are precompiled collections of SQL statements and optional control-of-flow logic. Their purpose is to encapsulate and reuse database logic, improve performance by reducing query parsing and execution time, enhance security by limiting direct table access, and simplify complex operations by bundling them into callable routines.

1. **What are the classifications of stored procedures?**

Stored procedures are classified into system procedures, user-defined procedures, and temporary procedures. System procedures are predefined by SQL Server and perform various administrative tasks. User-defined procedures are created by users to implement custom logic. Temporary procedures exist for the duration of a session or connection and are stored in the tempdb database.

1. **How can the flow of a procedure be managed?**

The flow of a procedure can be managed using control-of-flow statements such as IF...ELSE, WHILE, CASE, BEGIN...END, and RETURN. These statements allow for conditional execution, iteration, branching, and early termination of the procedure’s logic.

1. **What are the methods for creating stored procedures?**

Stored procedures can be created using the CREATE PROCEDURE statement followed by the procedure name and the SQL logic it encapsulates. For example, CREATE PROCEDURE procedure\_name AS BEGIN SQL statements END. They can also be defined with parameters for dynamic behavior and optional result sets.

1. **What are the methods for passing and returning parameters in a stored procedure?**

Parameters can be passed to stored procedures using input parameters defined in the procedure signature, allowing dynamic input values. Output parameters can be used to return values back to the caller. Procedures can also return a single integer value using the RETURN statement to indicate status or success.

1. **What does managing stored procedures involve?**

Managing stored procedures involves tasks such as creating, modifying, or deleting procedures using CREATE, ALTER, or DROP statements. It also includes granting, revoking, or auditing permissions for executing procedures, monitoring their performance, and debugging or optimizing their logic to ensure efficiency and maintainability.