

STORE PROCEDURE (Saklı Prosedürler), veritabanında önceden derlenmiş ve saklanmış SQL sorgularıdır. Bu prosedürler, genellikle bir veya daha fazla SQL komutunu içeren, parametre alabilen ve sonuçlar döndürebilen adlandırılmış SQL kod bloklarıdır. STORE PROCEDURE'ler, veritabanı yöneticileri veya uygulama geliştiricileri tarafından oluşturulur ve veritabanında saklanır.

STORE PROCEDURE'lerin avantajları şunlardır:

- * **Performans:** Sorgular bir kez derlenir ve saklandığı için daha hızlı çalışır.
- * **Güvenlik:** Parametreler aracılığıyla güvenlik kontrolleri yapılabilir.
- * **Tekrar Kullanım:** Aynı prosedür birden çok yerde kullanılabilir.
- * **Modülerlik:** Kod tekrarını azaltır ve bakımı kolaylaştırır.

STORE PROCEDURE Örneği

Örneğin, aşağıdaki gibi bir STORE PROCEDURE, Products tablosuna yeni bir ürün eklemek için parametre olarak ürün adı ve fiyatı alır:

```
CREATE PROCEDURE sp_AddProduct
```

```
    @ProductName NVARCHAR(100),
```

```
    @Price DECIMAL(10, 2)
```

```
AS
```

```
BEGIN
```

```
    INSERT INTO Products (Name, Price)
```

```
    VALUES (@ProductName, @Price);
```

```
END;
```

Bu STORE PROCEDURE, sp_AddProduct adı altında tanımlanmıştır. Parametre olarak @ProductName (ürün adı) ve @Price (ürün fiyatı) alır. İçinde INSERT INTO ifadesi ile Products tablosuna yeni bir ürün ekler.

STORE PROCEDURE Kullanımı C#

C# programında bu STORE PROCEDURE'ü kullanmak için SqlCommand sınıfı ile parametreler ekleyerek çağırabiliriz:

```
using System;
```

```
using System.Data;

using System.Data.SqlClient;

public class Program
{
    public static void Main()
    {
        // Veritabanı bağlantı dizesi

        string connectionString = "Server=myServerAddress;Database=myDataBase;User
Id=myUsername;Password=myPassword;";

        // SqlConnection nesnesi ve SqlCommand nesnesi oluşturma

        using (SqlConnection connection = new SqlConnection(connectionString))

        using (SqlCommand command = new SqlCommand("sp_AddProduct", connection))
        {
            try
            {
                // Komut tipini STORE PROCEDURE olarak belirtme

                command.CommandType = CommandType.StoredProcedure;

                // STORE PROCEDURE parametrelerini ekleme

                command.Parameters.AddWithValue("@ProductName", "Keyboard");

                command.Parameters.AddWithValue("@Price", 29.99);

                // Bağlantıyı açma

                connection.Open();

                Console.WriteLine("Bağlantı açıldı.");

                // Komutu yürütme

                command.ExecuteNonQuery();

                Console.WriteLine("Ürün başarıyla eklendi.");
```

```

    }

    catch (Exception ex)

    {

        Console.WriteLine("Hata: " + ex.Message);

    }

    finally

    {

        // Bağlantıyı kapatma

        connection.Close();

        Console.WriteLine("Bağlantı kapatıldı.");

    }

}

}

```

Yukarıdaki örnekte:

*SqlCommand nesnesi oluşturulmuş ve sp_AddProduct adlı STORE PROCEDURE çağrılmıştır.

*command.CommandType = CommandType.StoredProcedure; ile komut tipi STORE PROCEDURE olarak belirtilmiştir.

*command.Parameters.AddWithValue() ile STORE PROCEDURE'ün beklediği parametreler eklenmiştir (@ProductName ve @Price).

*command.ExecuteNonQuery() ile STORE PROCEDURE çağrılmış ve ürün veritabanına eklenmiştir.

Bu örnek, STORE PROCEDURE'lerin nasıl oluşturulup kullanılacağını ve C# programında nasıl çağrılacağını göstermektedir. STORE PROCEDURE içeriğini ve parametrelerini kendi veritabanınıza göre ayarlamalısınız (Server, Database, User Id, Password vb.).