<https://www.milanjovanovic.tech/blog/working-with-transactions-in-ef-core>

public List<Student> GetStudents()

{

//return \_context.Students.FromSql($"exec dbo.GetStudents").ToList();

//\_context.Database.ExecuteSql($"GetStudents");

var list = \_context.SpResults.FromSqlInterpolated($"GetStudents").ToList();

List<Student> lists = new List<Student>();

foreach(var obj in list)

{

var student = new Student()

{ Id = obj.Id, Name = obj.Name };

lists.Add(student);

}

return lists;

}

[How to Execute Stored Procedures With EF Core 7 - Code Maze (code-maze.com)](https://code-maze.com/efcore-execute-stored-procedures/)

**1**

Add SELECT \* FROM the same way you'd do in SQL, eg :

var query = Set.FromSqlInterpolated($"SELECT \* FROM dbo.tf\_ArtikelSearch ({filter.SprachCode})")

.Where(result => result.PackungId == 38673);

Functions can't be called directly in a SQL query, so EF Core assumes you're trying to call a stored procedure which can't be composed in SQL.

This [doc example](https://learn.microsoft.com/en-us/ef/core/querying/raw-sql#including-related-data) shows just that case :

var blogs = context.Blogs

.FromSqlInterpolated($"SELECT \* FROM dbo.SearchBlogs({searchTerm})")

.Include(b => b.Posts)

.ToList();