

NETCORE MVC CHEAT SHEET

BUILDER CADENA DE CONEXIÓN

```
builder.Services.AddDbContext<PROGRAMCONTEXT>(options =>
options.UseSqlServer(builder.Configuration.GetConnectionString("AppSettings
String")));
```

APPSETTINGS STRING

```
"ConnectionStrings": {
  "AppSettingsString": "Server=(localdb)\MSSQLLocalDB; Database=DATABASENAME;
Trusted_connection=true;"
}
```

MIGRACION CODE FIRST

Add-Migration NombreArchivo

Update-Database

DELETE UNNECESSARY RELATIONS IN THE CONTEXT, REMEMBER TO ADD THE NECESSARY RELATIONS WITH ONE TO MANY ETC

MIGRACION DB FIRST

IMPORTANT!! MAKE COPIES TO MAKE SURE NOT TO OVERWRITE ANYTHING

Scaffold-DbContext "Server=(localdb)\MSSQLLocalDB; Database=**DATABASENAME**;
Trusted_Connection=True" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models
-ContextDir Data -Tables ***LISTA DE TABLAS -f***

SEED DATA

CLASS MODEL

```
public static class SeedData
{
    public static void Initialize(IServiceProvider serviceProvider)
    {
        using (var context = new ProgramContext(
            serviceProvider.GetRequiredService<DbContextOptions<ProgramContext>>()))
        {
            if (context.ClassName.Any())
            {
                return;
            }

            context.ClassName.AddRange(
                new ClassName
                {
                    Title = "Sample Title",
                    Description = "Sample Description",
                    Rating = 4
                }
            );
            context.SaveChanges();
        }
    }
}
```

PROGRAM CS INITIALIZE

```
album.AlbumId = _context.Albums.Max(a => a.AlbumId) + 1;
```

SNIPPETS AND REMINDERS

INCREMENT ID IN CONTROLLER

```
album.AlbumId = _context.Albums.Max(a => a.AlbumId) + 1;
```

ViewData

```
//Controller Code
public ActionResult Index()
{
    List<string> Student = new List<string>();
    Student.Add("Jignesh");
    Student.Add("Tejas");
    Student.Add("Rakesh");

    ViewData["Student"] = Student;
    return View();
}
//page code
<ul>
    <% foreach (var student in ViewData["Student"] as List<string>)
        { %>
    <li><%= student%></li>
    <% } %>
</ul>
```

ViewBag

```
//Controller Code
public ActionResult Index()
{
    List<string> Student = new List<string>();
    Student.Add("Jignesh");
    Student.Add("Tejas");
    Student.Add("Rakesh");

    ViewBag.Student = Student;
```

```
        return View();
    }
    //page code
    <ul>
        <% foreach (var student in ViewBag.Student)
            { %>
            <li><%= student%></li>
            <% } %>
    </ul>
```

Enable delete with include

```
var album = await _context.Albums
    .Include(t => t.Tracks)
    .FirstOrDefaultAsync(a => a.AlbumId == id);

var artist = await _context.Artists
    .Include(a => a.Albums)
    .ThenInclude(m => m.Tracks)
    .FirstOrDefaultAsync(m => m.ArtistId == id);
```

Change context to add one to many relation

```
entity.HasMany(e => e.Albums).WithOne(p => p.Artist)
    .HasForeignKey(d => d.ArtistId)
    .OnDelete(DeleteBehavior.Cascade);
```