1. instalam pachetele - Core, Core.Design, Core.SqlServer, Core.Tools

2. baza de date - View->SqlServerObjectExplorer

3. facem un models->new numit AppDBContext

```
namespace NumeAplicatie.Models
        {
                public class AppDBContext : DbContext
         {
        // constructorul contextului bazei de date
        public AppDBContext(DbContextOptions<AppDBContext> options) : base(options) { }

         public DbSet<GiftCard> GiftCards { get; set; }
         public DbSet<Brand> Brands { get; set; }
        }
  }
```

4. facem modelele, in fiecare includem clasa cu particularitatile

5. facem un controller in care enumeram metodele disponibile pentru clasa

6. pentru fiecare metoda facem un view

    --!

    in fiecare view la inceput punem asta:

```
@if (ViewBag.message != null)
{
        <h2> @ViewBag.message</h2>
}
```

7. in controller, la metoda, punem asta:

```
if (TempData.ContainsKey("message"))
   ViewBag.message = TempData["message"];
```

– appsettings.json:

```
{
  "ConnectionStrings": {
    "DefaultConnection": "Data Source=(localdb)\\mssqllocaldb;Initial Catalog=ModelNr1;Integrated Security=True;MultipleActiveResultSets=True"
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*"
}
```

– program.cs

```
// conexiunea cu baza de date
// extragem stringul de conexiune, stocat in appsettings.json
var connectionString = builder.Configuration.GetConnectionString("DefaultConnection");
// si realizam conexiunea
builder.Services.AddDbContext<AppDBContext>(options =>
options.UseSqlServer(connectionString));
```