

DWBE

Tarea #2: Web api inicial

Desarrollo web: Back-end

Docente: Gustavo Adolfo Rodríguez Martínez.

Grupo: 037

Matrícula	Nombre
1962111	JOHANN JOSEPH VELÁZQUEZ ANTONIO

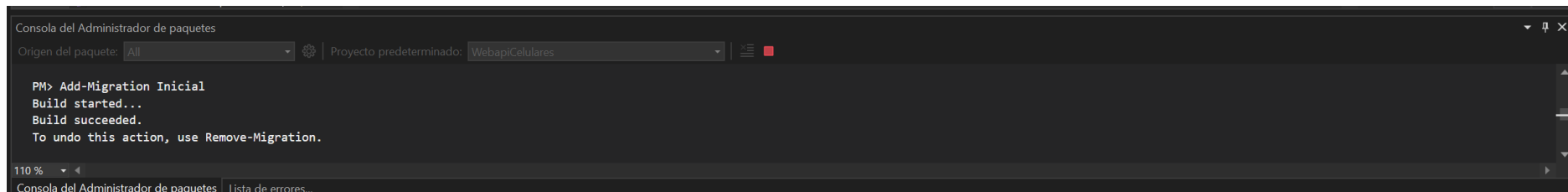
- **1xx:** Mensaje informativo.
- **2xx:** Exito
 - 200 OK
 - 201 Created
 - 202 Accepted
 - 204 No Content
- **3xx:** Redirección
 - 300 Multiple Choice
 - 301 Moved Permanently
 - 302 Found
 - 304 Not Modified
- **4xx:** Error del cliente
 - 400 Bad Request
 - 401 Unauthorized
 - 403 Forbidden
 - 404 Not Found
- **5xx:** Error del servidor
 - 500 Internal Server Error
 - 501 Not Implemented
 - 502 Bad Gateway
 - 503 Service Unavailable

Antes que nada se me hace importante considerar y conocer los cosdigos de estado que nos pueden arrojar al hacer una petición, ya que así podremos entender que es lo que sucede en nuestro sistema y ayudarnos a arreglar problemas

```
services.AddDbContext<AplicationDbContext>(  
    options =>  
        options.UseSqlServer(Configuration.GetConnectionString("defaultConection"))  
    );  
services.AddEndpointsApiExplorer();  
services.AddSwaggerGen( c =>  
    {  
        c.SwaggerDoc("v1", new Microsoft.OpenApi.Models.OpenApiInfo { Title = "ApiCelulares" });  
    }  
);  
}
```

Añadimos la conexión a la base de datos que previamente definimos en el default string modificando el archivo startup.cs

Añadimos la migración inicial



```
Consola del Administrador de paquetes  
Origen del paquete: All Proyecto predeterminado: WebapiCelulares  
  
PM> Add-Migration Inicial  
Build started...  
Build succeeded.  
To undo this action, use Remove-Migration.  
  
110 %  
Consola del Administrador de paquetes Lista de errores...
```

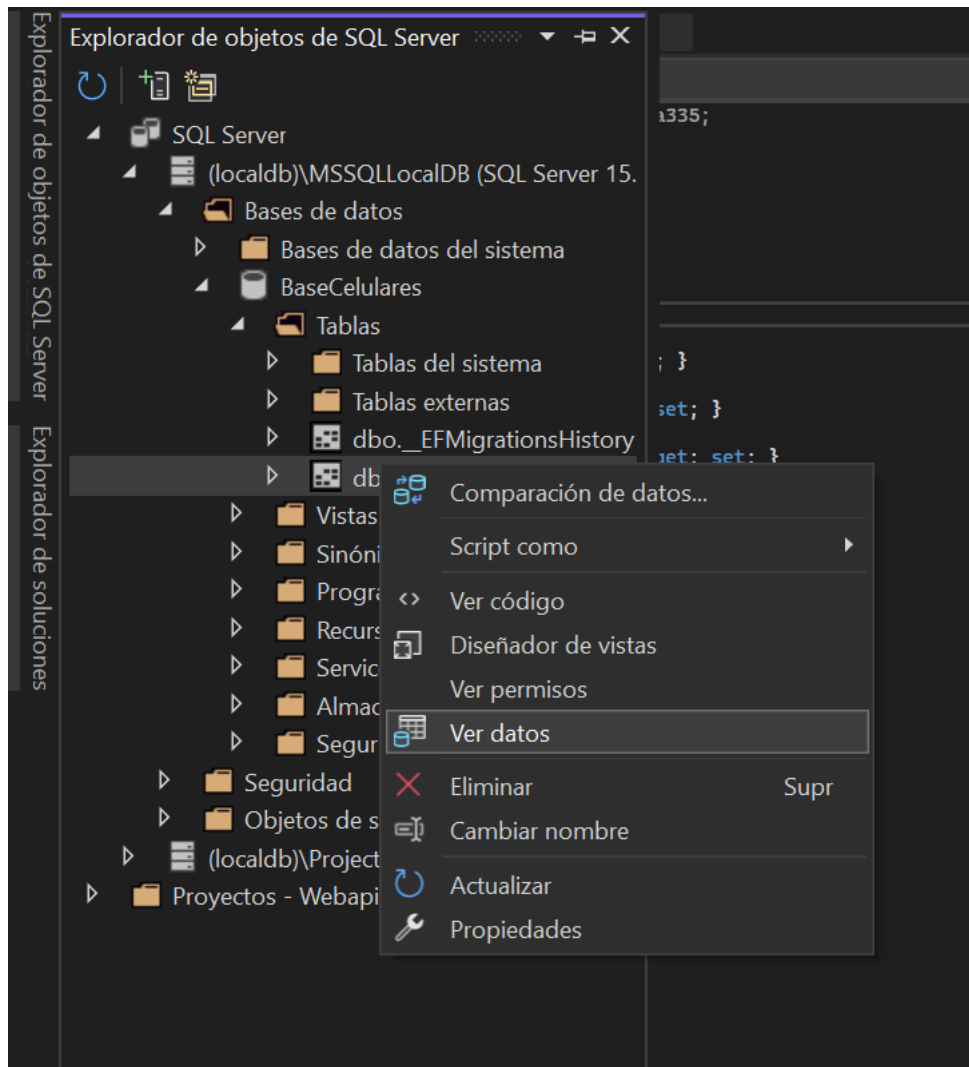
Se nos genera este archivo

```
Startup.cs  MarcaController.cs  20230307223826_Inicial.cs  Marcas.cs  Celular.cs  AplicacionDbContext.cs  Readmde.md  WebapiCelulares.csproj  Program.cs  Novedades  launchSettings.json

WebapiCelulares  WebapiCelulares.Migrations.Inicial  Up(MigrationBuilder migrationBuilder)

1  using Microsoft.EntityFrameworkCore.Migrations;
2
3  #nullable disable
4
5  namespace WebapiCelulares.Migrations
6  {
7      /// <inheritdoc />
8      public partial class Inicial : Migration
9      {
10         /// <inheritdoc />
11         protected override void Up(MigrationBuilder migrationBuilder)
12         {
13             migrationBuilder.CreateTable(
14                 name: "Celulares",
15                 columns: table => new
16                 {
17                     id = table.Column<int>(type: "int", nullable: false)
18                         .Annotation("SqlServer:Identity", "1, 1"),
19                     precio = table.Column<int>(type: "int", nullable: false),
20                     modelo = table.Column<string>(type: "nvarchar(max)", nullable: true),
21                     descripcion = table.Column<string>(type: "nvarchar(max)", nullable: true),
22                     id_marca = table.Column<int>(type: "int", nullable: false)
23                 },
24                 constraints: table =>
25                 {
26                     table.PrimaryKey("PK_Celulares", x => x.id);
27                 });
28         }
29
30         /// <inheritdoc />
31         protected override void Down(MigrationBuilder migrationBuilder)
32         {
33             migrationBuilder.DropTable(
34                 name: "Celulares");
35         }
36     }
37 }
38
```

```
PM> Update-Database
Build started...
Build succeeded.
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (193ms) [Parameters=[], CommandType='Text', CommandTimeout='60']
  CREATE DATABASE [BaseCelulares];
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (52ms) [Parameters=[], CommandType='Text', CommandTimeout='60']
  IF SERVERPROPERTY('EngineEdition') <> 5
  BEGIN
    ALTER DATABASE [BaseCelulares] SET READ_COMMITTED_SNAPSHOT ON;
  END;
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (6ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (6ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  CREATE TABLE [_EFMigrationsHistory] (
    [MigrationId] nvarchar(150) NOT NULL,
    [ProductVersion] nvarchar(32) NOT NULL,
    CONSTRAINT [PK__EFMigrationsHistory] PRIMARY KEY ([MigrationId])
  );
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (14ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  SELECT OBJECT_ID(N'[_EFMigrationsHistory]');
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (3ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  SELECT [MigrationId], [ProductVersion]
  FROM [_EFMigrationsHistory]
  ORDER BY [MigrationId];
Microsoft.EntityFrameworkCore.Migrations[20402]
  Applying migration '20230307224524_Inicial'.
Applying migration '20230307224524_Inicial'.
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (2ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  CREATE TABLE [Celulares] (
    [id] int NOT NULL IDENTITY,
    [precio] int NOT NULL,
    [modelo] nvarchar(max) NULL,
    [descripcion] nvarchar(max) NULL,
    [id_marca] int NOT NULL,
    CONSTRAINT [PK_Celulares] PRIMARY KEY ([id])
  );
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (2ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  INSERT INTO [_EFMigrationsHistory] ([MigrationId], [ProductVersion])
  VALUES (N'20230307224524_Inicial', N'8.0.0-preview.1.23111.4');
Done.
PM> |
```



id	precio	modelo	descripcion	id_marca
NULL	NULL	NULL	NULL	NULL

Podemos apreciar el explorador de objetos de sql server y si presionamos en ver datos aun

no aparece nada pues no hemos hecho ningún insert

```
11 public class CelularesController : ControllerBase
12 {
13     private readonly ApplicationDbContext dbContext;
14
15     0 referencias
16     public CelularesController(ApplicationDbContext dbContext)
17     {
18         this.dbContext = dbContext;
19     }
20
21     [HttpGet]
22     0 referencias
23     public async Task<ActionResult<List<Celular>>> Get()
24     {
25         return await dbContext.Celulares.ToListAsync();
26     }
27
28     [HttpPost]
29     0 referencias
30     public async Task<ActionResult> Post(Celular celular)
31     {
32         dbContext.Add(celular);
33
34         await dbContext.SaveChangesAsync();
35
36         return Ok();
37     }
38 }
```

En el controlador de celulares se añaden los siguientes métodos, y al probarlos nos genera lo siguiente:

The image shows a REST client interface with a light green header bar. The header bar contains a green tab labeled "POST" and the endpoint "/api/celulares" on the left, and a small upward-pointing arrow on the right. Below the header, there are three main sections: "Parameters", "Request body", and a large text area for the request body. The "Parameters" section has a "Cancel" button (outlined in red) and a "Reset" button. The "Request body" section has a dropdown menu set to "application/json". The large text area contains a JSON object:

```
{  "id": 0,  "precio": 0,  "modelo": "iphone",  "descripcion": "string",  "id_marca": 0}
```

 At the bottom of the interface, there is a blue "Execute" button and a "Clear" button.

POST /api/celulares

Parameters

No parameters

Request body

application/json

```
{  "id": 0,  "precio": 0,  "modelo": "iphone",  "descripcion": "string",  "id_marca": 0}
```

Execute Clear

Hacemos un post y nos genera un response el cual es

Curl

```
curl -X 'POST' \
  'https://localhost:7170/api/celulares' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 0,
    "precio": 0,
    "modelo": "iphone",
    "descripcion": "string",
    "id_marca": 0
  }'
```



Request URL

https://localhost:7170/api/celulares

Server response

Code	Details
------	---------

200

Response headers

```
content-length: 0
date: Tue, 07 Mar 2023 23:50:20 GMT
server: Kestrel
```

Responses

Code	Description
------	-------------

200

Success

Links

No links

```

info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (2ms) [Parameters=[@p0='?' (Size = 4000), @p1='?' (DbType = Int32), @p2='?' (Size = 4000), @p3='?' (DbType = Int32)], CommandType='Text', CommandTimeout='30']
      SET IMPLICIT_TRANSACTIONS OFF;
      SET NOCOUNT ON;
      INSERT INTO [Celulares] ([descripcion], [id_marca], [modelo], [precio])
      OUTPUT INSERTED.[id]
      VALUES (@p0, @p1, @p2, @p3);
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT [c].[id], [c].[descripcion], [c].[id_marca], [c].[modelo], [c].[precio]
      FROM [Celulares] AS [c]
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (11ms) [Parameters=[@p0='?' (Size = 4000), @p1='?' (DbType = Int32), @p2='?' (Size = 4000), @p3='?' (DbType = Int32)], CommandType='Text', CommandTimeout='30']
      SET IMPLICIT_TRANSACTIONS OFF;
      SET NOCOUNT ON;
      INSERT INTO [Celulares] ([descripcion], [id_marca], [modelo], [precio])
      OUTPUT INSERTED.[id]
      VALUES (@p0, @p1, @p2, @p3);
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT [c].[id], [c].[descripcion], [c].[id_marca], [c].[modelo], [c].[precio]
      FROM [Celulares] AS [c]
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (1ms) [Parameters=[@p0='?' (Size = 4000), @p1='?' (DbType = Int32), @p2='?' (Size = 4000), @p3='?' (DbType = Int32)], CommandType='Text', CommandTimeout='30']
      SET IMPLICIT_TRANSACTIONS OFF;
      SET NOCOUNT ON;
      INSERT INTO [Celulares] ([descripcion], [id_marca], [modelo], [precio])
      OUTPUT INSERTED.[id]
      VALUES (@p0, @p1, @p2, @p3);
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (1ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT [c].[id], [c].[descripcion], [c].[id_marca], [c].[modelo], [c].[precio]
      FROM [Celulares] AS [c]

```

Hice varios inserts, y cuando generamos una petición get, como lo establecimos en el cotnrolador nos mostrara una lista de los celulares registrados

GET

/api/celulares

Parameters

Cancel

No parameters

Execute

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

```
[
  {
    "id": 0,
    "precio": 0,
    "modelo": "string",
    "descripcion": "string",
    "id_marca": 0
  }
]
```

Haciendo esta petición nos genera

CodeDetails

200

Response body

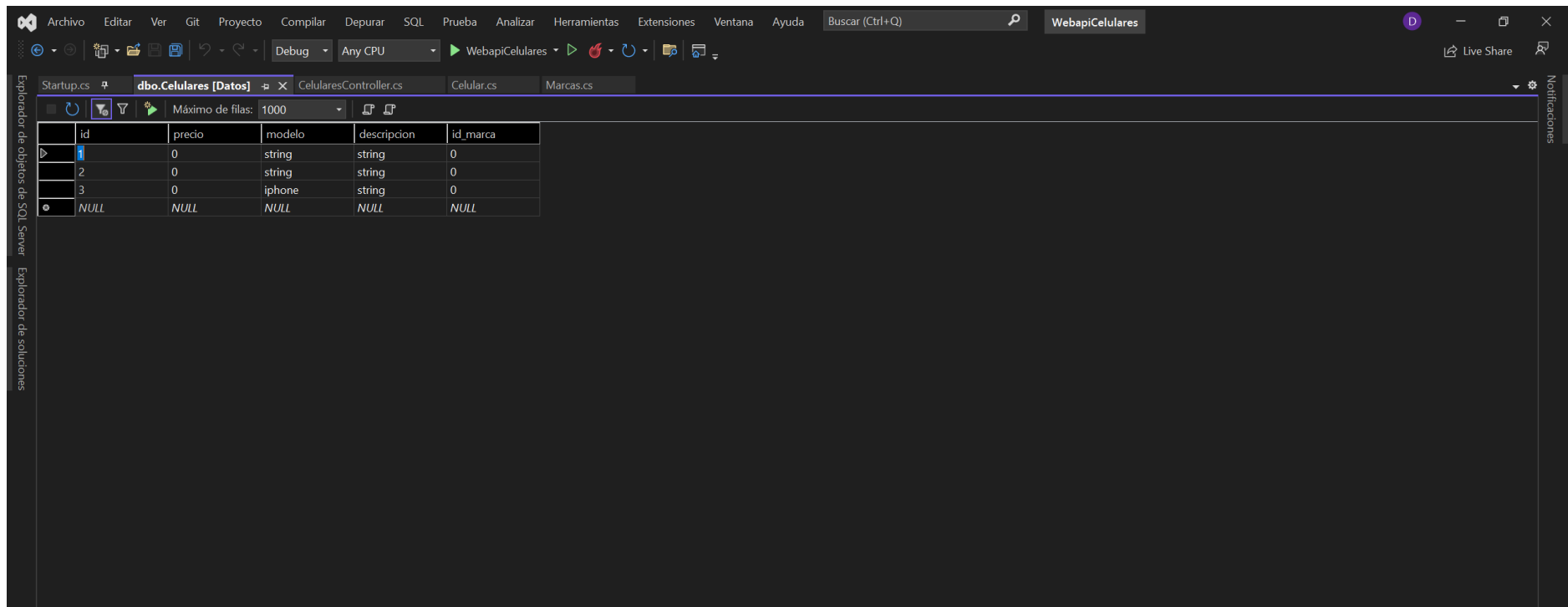
```
[
  {
    "id": 1,
    "precio": 0,
    "modelo": "string",
    "descripcion": "string",
    "id_marca": 0
  },
  {
    "id": 2,
    "precio": 0,
    "modelo": "string",
    "descripcion": "string",
    "id_marca": 0
  },
  {
    "id": 3,
    "precio": 0,
    "modelo": "iphone",
    "descripcion": "string",
    "id_marca": 0
  }
]
```

Download

Response headers

```
content-type: application/json; charset=utf-8
date: Tue, 07 Mar 2023 23:53:56 GMT
server: Kestrel
```

Podemos notar que el id automáticamente se va incrementando



The screenshot shows the Visual Studio Code interface with the SQL Explorer on the left. The active table is 'dbo.Celulares' with the following data:

	id	precio	modelo	descripcion	id_marca
▶	1	0	string	string	0
	2	0	string	string	0
	3	0	iphone	string	0
✱	NULL	NULL	NULL	NULL	NULL

Si nos vamos al explorador de objetos de sql

```
35
36 [HttpPut("{id : int}")]
    0 referencias
37 public async Task<ActionResult> Put(Celular celular, int id)
38 {
39     if(celular.id != id)
40     {
41         return BadRequest("El id del alumno no coincide con el establecido en la url");
42     }
43
44     dbContext.Update(celular);
45     await dbContext.SaveChangesAsync();
46     return Ok();
47 }
48
49
```

En el archivo CelularesController, se añade un método put para actualizar un registro con un id dado

PUT

/api/celulares

Parameters

Cancel

Reset

Name	Description
id	
integer(\$int32)	1
(query)	

Request body

application/json

```
{  "id": 1,  "precio": 0,  "modelo": "string",  "descripcion": "string",  "id_marca": 0}
```

Execute

Si hago una consulta dando un celular y un id

Curl

```
curl -X 'PUT' \
  'https://localhost:7170/api/celulares?id=1' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 1,
    "precio": 0,
    "modelo": "modeloActualizado",
    "descripcion": "string",
    "id_marca": 0
  }'
```

Request URL

https://localhost:7170/api/celulares?id=1

Server response

Code	Details
------	---------

200	
-----	--

Response headers

```
content-length: 0
date: Wed, 08 Mar 2023 03:15:31 GMT
server: Kestrel
```

Responses

Code	Description
------	-------------

200	
-----	--

Success

Links

No links

me genera la siguiente respuesta, y si nos checamos en la base de datos

Startup.cs dbo.Celulares [Datos] CelularesController.cs Celular.cs Marcas.cs					
<div> Máximo de filas: 1000 </div>					
	id	precio	modelo	descripcion	id_marca
▶	1	0	modeloActualizado	string	0
	2	0	string	string	0
	3	0	iphone	string	0
⚙	NULL	NULL	NULL	NULL	NULL

Apreciamos que, efectivamente se actualizo el registro

En dado caso que los id's no correspondan, nos manda el mensaje

400
Undocumented Error: response status is 400

Response body

El id del alumno no coincide con el establecido en la url

Download

Response headers

Añadimos la función de delete

```
[HttpDelete("{id}")]
0 referencias
public async Task<IActionResult> Delete(int id)
{
    var exist = await dbContext.Celulares.AnyAsync(x => x.id == id);

    if(!exist)
    {
        return NotFound();
    }

    dbContext.Remove(new Celular()
    {
        id = id
    });
    await dbContext.SaveChangesAsync();
    return Ok();
}
```

Por ejemplo, borremos el registro con el id 1 en nuestra bdd

```
[
  {
    "id": 1,
    "precio": 0,
    "modelo": "modeloActualizado",
    "descripcion": "string",
    "id_marca": 0
  },
  {
    "id": 2,
    "precio": 0,
    "modelo": "string",
    "descripcion": "string",
    "id_marca": 0
  },
  {
    "id": 3,
    "precio": 0,
    "modelo": "iphone",
    "descripcion": "string",
    "id_marca": 0
  }
]
```

Si ejecutamos

Parameters

Cancel

Name	Description
id * required integer(\$int32) (path)	<input type="text" value="1"/>

Execute

Clear

Code	Details	
200	<div>Response headers</div> <div><div>content-length: 0</div><div>date: Wed, 08 Mar 2023 03:30:54 GMT</div><div>server: Kestrel</div></div>	
Responses		
Code	Description	Links
200	Success	No links

Nos genera una response 200

Y si volvemos a ejecutar el método get podemos apreciar que

Response body

```
[
  {
    "id": 2,
    "precio": 0,
    "modelo": "string",
    "descripcion": "string",
    "id_marca": 0
  },
  {
    "id": 3,
    "precio": 0,
    "modelo": "iphone",
    "descripcion": "string",
    "id_marca": 0
  }
]
```

Efectivamente se borro el registro con el id 1

DELETE

/api/celulares/{id}

Parameters

Cancel

Name	Description
id * required integer(\$int32) (path)	<input type="text" value="31"/>

Execute

Clear

Si hacemos una mala petición con un id que no existe

404

Undocumented

Error: response status is 404

Response body

```
{
  "type": "https://tools.ietf.org/html/rfc7231#section-6.5.4",
  "title": "Not Found",
  "status": 404,
  "traceId": "00-63557750f1b4ff3b3b554e9bf83f642f-d5a272503321eb1d-00"
}
```

Download

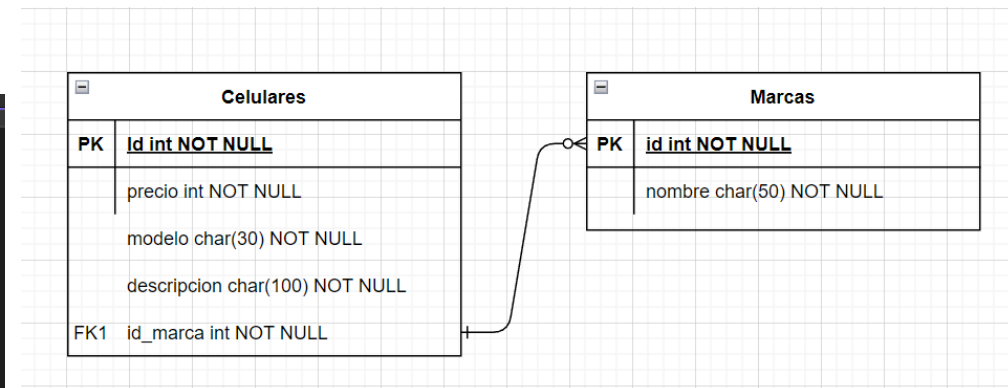
Response headers

```
content-type: application/problem+json; charset=utf-8
date: Wed, 08 Mar 2023 03:35:10 GMT
server: Kestrel
```

Nos arroja un badrequest

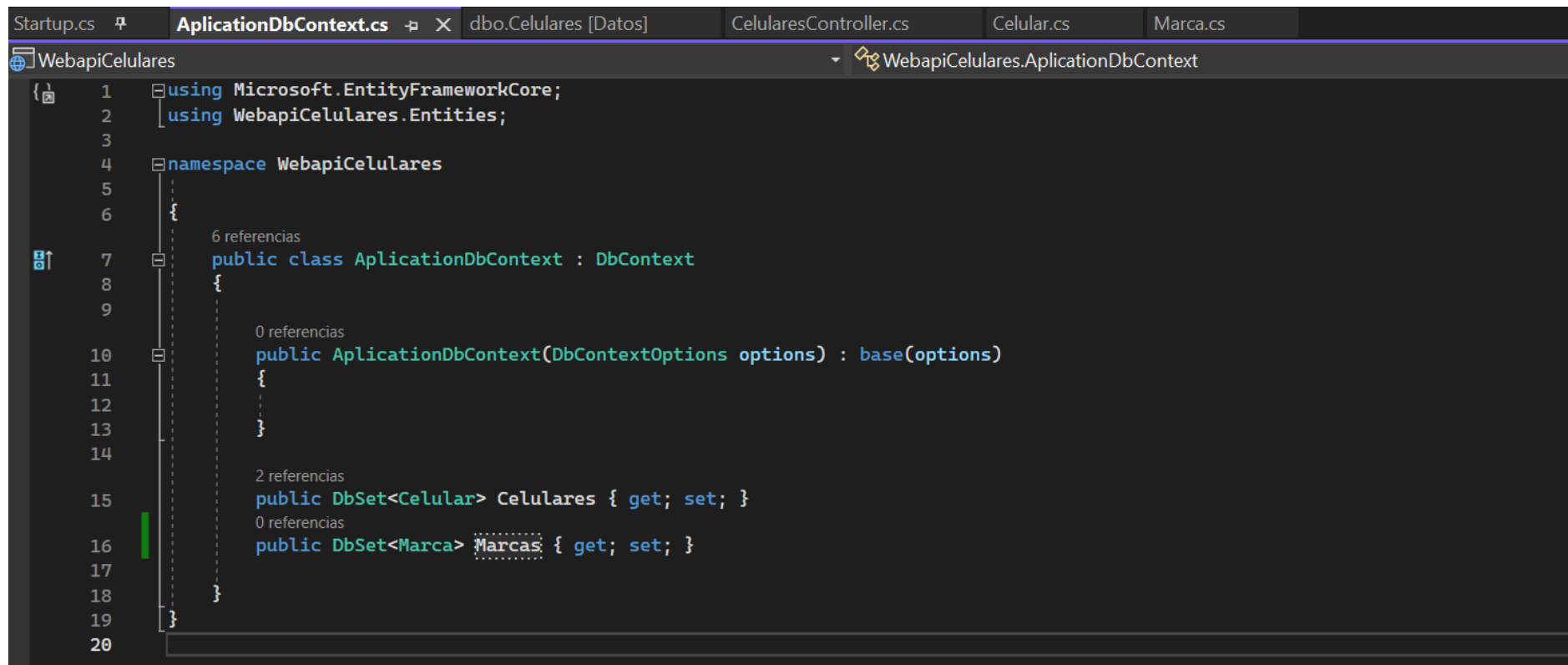
Añadimos la relación entre los atributos

```
Startup.cs 4  MarcaController.cs  CelularesController.cs  Marcas.cs  Celular.cs  20230308034414_Pasajeros.cs  ApplicationDbContext.cs  dbo.Celulares [Datos]
WebapiCelulares
1  using System.Reflection.Metadata.Ecma335;
2  using System.Text.Json.Serialization;
3
4  namespace WebapiCelulares.Entities
5  {
6      6 referencias
7      public class Celular
8      {
9          3 referencias
10         public int id { get; set; }
11         0 referencias
12         public int precio { get; set; }
13         0 referencias
14         public string modelo { get; set; }
15         0 referencias
16         public string descripcion { get; set; }
17         0 referencias
18         public int id_marca { get; set; }
19
20         [JsonIgnore]
21         1 referencia
22         public Marca marca { get; set; }
23     }
24 }
```



correspondientes

```
Startup.cs 7  dbo.Celulares [Datos]  CelularesController.cs  Celular.cs  Marca.cs  WebapiCelulares
WebapiCelulares
1  namespace WebapiCelulares.Entities
2  {
3      5 referencias
4      public class Marca
5      {
6          2 referencias
7          public int id { get; set; }
8          2 referencias
9          public string nombre { get; set; }
10
11          0 referencias
12          public List<Celular> celulares { get; set; }
13      }
14 }
```

```
1 using Microsoft.EntityFrameworkCore;
2 using WebapiCelulares.Entities;
3
4 namespace WebapiCelulares
5 {
6     6 referencias
7     public class ApplicationDbContext : DbContext
8     {
9         0 referencias
10     public ApplicationDbContext(DbContextOptions options) : base(options)
11     {
12     }
13
14     2 referencias
15     public DbSet<Celular> Celulares { get; set; }
16     0 referencias
17     public DbSet<Marca> Marcas { get; set; }
18 }
19
20
```

Añadimos nuestra entidad a nuestro dbContext

Each package is licensed to you by its owner. NuGet is not responsible for, nor does it grant any licenses to, third-party packages. Some packages may include dependencies which are governed by additional licenses. Follow the package source (feed) URL to determine any dependencies.

Package Manager Console Host Version 6.5.0.154

Type 'get-help NuGet' to see all available NuGet commands.

PM> Add-Migration Pasajeros

Build started...

Build succeeded.

Añadimos la migración

```
Startup.cs 202303080344... Pasajeros.cs ApplicationDbContext.cs dbo.Celulares [Datos] CelularesController.cs Celular.cs Marca.cs
WebapiCelulares WebapiCelulares.Migrations.Pasajeros Up(MigrationBuilder migration)

4
5 namespace WebapiCelulares.Migrations
6 {
7     /// <inheritdoc />
8     public partial class Pasajeros : Migration
9     {
10         /// <inheritdoc />
11         protected override void Up(MigrationBuilder migrationBuilder)
12         {
13             migrationBuilder.AddColumn<int>(
14                 name: "marcaid",
15                 table: "Celulares",
16                 type: "int",
17                 nullable: true);
18
19             migrationBuilder.CreateTable(
20                 name: "Marcas",
21                 columns: table => new
22                 {
23                     id = table.Column<int>(type: "int", nullable: false)
24                         .Annotation("SqlServer:Identity", "1, 1"),
25                     nomber = table.Column<string>(type: "nvarchar(max)", nullable: true)
26                 },
27                 constraints: table =>
28                 {
29                     table.PrimaryKey("PK_Marcas", x => x.id);
30                 });
31
32             migrationBuilder.CreateIndex(
33                 name: "IX_Celulares_marcaid",
34                 table: "Celulares",
35                 column: "marcaid");
36
37             migrationBuilder.AddForeignKey(
38                 name: "FK_Celulares_Marcas_marcaid",
39                 table: "Celulares",
40                 column: "marcaid",
41                 principalTable: "Marcas",
42                 principalColumn: "id");
43         }
44
45         /// <inheritdoc />
46         protected override void Down(MigrationBuilder migrationBuilder)
47         {
48             migrationBuilder.DropForeignKey(
49                 name: "FK_Celulares_Marcas_marcaid",
50                 table: "Celulares");
51
52             migrationBuilder.DropTable(
53                 name: "Marcas");
54
55             migrationBuilder.DropIndex(
56                 name: "IX_Celulares_marcaid",
57                 table: "Celulares");
58
59             migrationBuilder.DropColumn(
60                 name: "marcaid",
61                 table: "Celulares");
62         }
63     }
64 }
```

Nos genera este archivo

```
Consola del Administrador de paquetes
Origen del paquete: All Proyecto predeterminado: WebapiCelulares
Each package is licensed to you by its owner. NuGet is not responsible for, nor does it grant any licenses to, third-party packages. Some packages may include dependencies which are governed by additional licenses. Follow the package source (feed) URL to determine any dependencies.
Package Manager Console Host Version 6.5.0.154
Type 'get-help NuGet' to see all available NuGet commands.
PM> Add-Migration Pasajeros
Build started...
Build succeeded.
To undo this action, use Remove-Migration.
PM> Update-Database
Build started...
Build succeeded.
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (13ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (12ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID(N'[_EFMigrationsHistory]');
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (1ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID(N'[_EFMigrationsHistory]');
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (2ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT [MigrationId], [ProductVersion]
    FROM [_EFMigrationsHistory]
    ORDER BY [MigrationId];
Microsoft.EntityFrameworkCore.Migrations[20402]
    Applying migration '20230308034414_Pasajeros'.
Applying migration '20230308034414_Pasajeros'.
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (10ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    ALTER TABLE [Celulares] ADD [marcaid] int NULL;
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (7ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    CREATE TABLE [Marcas] (
        [id] int NOT NULL IDENTITY,
        [nombre] nvarchar(max) NULL,
        CONSTRAINT [PK_Marcas] PRIMARY KEY ([id])
    );
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (22ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    CREATE INDEX [IX_Celulares_marcaid] ON [Celulares] ([marcaid]);
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (5ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    ALTER TABLE [Celulares] ADD CONSTRAINT [FK_Celulares_Marcas_marcaid] FOREIGN KEY ([marcaid]) REFERENCES [Marcas] ([id]);
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (3ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    INSERT INTO [_EFMigrationsHistory] ([MigrationId], [ProductVersion])
    VALUES (N'20230308034414_Pasajeros', N'8.0.0-preview.1.23111.4');
Done.
PM>
```

Actualizamos nuestra base de datos

Así queda el respectivo controlador de marca

```
Startup.cs  #  MarCaController.cs  20230308034414_Pasajeros.cs  AplicacionDbContext.cs  dbo.Celulares [Datos]  CelularesController.cs  Celular.cs  Marca.cs
WebapiCelulares  WebapiCelulares.Controllers.MarCaController  Put(Marca marca, int id)

1  using Microsoft.AspNetCore.Mvc;
2  using Microsoft.EntityFrameworkCore;
3  using WebapiCelulares.Entities;
4
5  namespace WebapiCelulares.Controllers
6  {
7      [ApiController]
8      [Route("api/marca")]
9      public class MarCaController : ControllerBase
10     {
11
12         private readonly AplicacionDbContext _context;
13
14         public MarCaController(AplicacionDbContext context)
15         {
16             _context = context;
17         }
18
19         [HttpGet("{id}")]
20         public async Task<ActionResult<Marca>> GetById(int id)
21         {
22             return await _context.Marcas.FirstOrDefaultAsync(x => x.id == id);
23         }
24
25         [HttpGet]
26         public async Task<ActionResult<List<Marca>>> GetAll()
27         {
28             return await _context.Marcas.ToListAsync();
29         }
30
31         [HttpPost]
32         public async Task<ActionResult> Post(Marca marca)
33         {
34             var existeMarca = await _context.Marcas.AnyAsync(x => x.id == marca.id);
35
36             if (!existeMarca) {
37                 return BadRequest("No existe la marca");
38             }
39
40             await _context.SaveChangesAsync();
41             return Ok();
42         }
43
44         [HttpPut("{id}")]
45         public async Task<ActionResult> Put(Marca marca, int id)
46         {
47             var exist = await _context.Marcas.AnyAsync(x => x.id == id);
48
49             if (!exist)
50             {
51                 return BadRequest("La marca no existe");
52             }
53
54             _context.Update(marca);
55             await _context.SaveChangesAsync();
56             return Ok();
57         }
58     }
59
60
61
62
63
64
```

Marca



GET

/api/marca/{id}



PUT

/api/marca/{id}



DELETE

/api/marca/{id}



GET

/api/marca



POST

/api/marca



GET

/api/marca



Parameters

Try it out

No parameters

Responses

Code

Description

Links

200

Success

No links

Media type

text/plain



Controls Accept header.

Example Value | Schema

```
[
  {
    "id": 0,
    "number": "string",
    "celulares": {
      "id": 0,
      "precio": 0,
      "modelo": "string",
      "descripcion": "string",
      "id_marca": 0
    }
  }
]
```