

# Sprawozdanie z realizacji API w .NET

## 1. Wprowadzenie

Celem zadania było stworzenie API w technologii .NET umożliwiającego zarządzanie danymi dotyczącymi samochodów, modeli, marek oraz kontraktów. API pozwala na dodawanie, edycję, usuwanie oraz pobieranie informacji o dostępnych pojazdach i ich powiązaniach.

## 2. Technologia i narzędzia

- ASP.NET Core 9.0,
- InMemory Database (do testów),
- JSON jako format wymiany danych,
- Swagger (NSwag) do testowania.

## 3. Struktura projektu

Projekt został podzielony na następujące warstwy:

- **Models** - Definicje encji: Car, Model, Marka, Kontrakt,
- **Controllers** - Kontrolery obsługujące żądania HTTP,
- **Data** - Konfiguracja DbContext,
- **ModelsDTO** - Modele do transferu danych.

## 4. Przykład modelu danych Car

```
public class Car
{
    5 references
    public int Id { get; set; }
    4 references
    public string LicensePlate { get; set; } = string.Empty;
    5 references
    public int ModelId { get; set; }
    2 references
    public Model Model { get; set; } = null!;
    4 references
    public int Year { get; set; }
}
```

Opis pól:

- **Id** (*int*) - Unikalny identyfikator samochodu.
- **LicensePlate** (*string*) - Numer rejestracyjny pojazdu.
- **ModelId** (*int*) - Identyfikator modelu pojazdu (klucz obcy do tabeli Model).
- **Model** (*Model*) - Obiekt reprezentujący powiązany model (referencja do encji Model).
- **Year** (*int*) - Rok produkcji pojazdu.

## 5. Przykład metody get kontrolera Car

```
// GET: api/Car
[HttpGet]
0 references
public async Task<ActionResult<IEnumerable<Car>>> GetCars()
{
    return await _context.Cars.Include(m => m.Model).ThenInclude(c => c.Mark).ToListAsync();
}
```

Metoda ta zwraca wszystkie dane z tabeli Car wraz z danymi z powiązanych tabel.

## 6. Przykład relacji tabeli Car

```
modelBuilder.Entity<Car>()
    .HasOne(c => c.Model)
    .WithMany()
    .HasForeignKey(c => c.ModelId);
```

Opis:

- **HasOne(c => c.Model)** - Definiuje, że encja Car ma jedną relację do encji Model.
- **WithMany()** - Określa, że encja Model może być powiązana z wieloma Car (relacja jeden-do-wielu).
- **HasForeignKey(c => c.ModelId)** - Ustawia ModelId jako klucz obcy w tabeli Car, co pozwala powiązać każdy samochód z odpowiednim modelem.

## 7. Dodanie testowych danych tabeli Car

```
if (!context.Cars.Any())
{
    context.Cars.AddRange(
        new Car { Id = 1, LicensePlate = "ABC123", ModelId = 1, Year = 2020 },
        new Car { Id = 2, LicensePlate = "XYZ789", ModelId = 2, Year = 2018 },
        new Car { Id = 3, LicensePlate = "LMN456", ModelId = 3, Year = 2022 }
    );
    context.SaveChanges();
}
```

# 8. Testowanie api Swaggerem

GET

/api/Car

^

Parameters

Cancel

No parameters

Execute

Clear

Responses

Curl

curl -X 'GET' \

'https://localhost:7254/api/Car' \

-H 'accept: text/plain'

Request URL

https://localhost:7254/api/Car

Server response

Code

Details

200

Response body

"modelId": 2,

"model": {

"id": 2,

"name": "Camry",

"markId": 1,

"mark": {

"id": 1,

"name": "Toyota"

}

},

"year": 2018

},

{

"id": 1,

"licensePlate": "ABC123",

"modelId": 1,

"model": {

"id": 1,

"name": "Corolla",

"markId": 1,

"mark": {

"id": 1,

"name": "Toyota"

}

}

},

"year": 2020

}

]

Download

DELETE

/api/Car/{id}

^

Parameters

Cancel

Name

Description

id \* required

integer(\$int32)

(path)

4

Execute

Clear

Responses

Curl

curl -X 'DELETE' \

'https://localhost:7254/api/Car/4' \

-H 'accept: \*/\*'

Request URL

https://localhost:7254/api/Car/4

Server response

Code

Details

204

undocumented

Response headers

date: Fri,28 Mar 2025 13:12:17 GMT

server: Kestrel

x-firefox-spy: h2

Responses

Code

Description

Links

200

OK

No links

POST

/api/Car

Cancel

Reset

Parameters

No parameters

Request body required

application/json

```
{  "licensePlate": "WPC199",  "modelId": 1,  "year": 1999}
```

Execute

Clear

Responses

Curl

```
curl -X 'POST' \  'https://localhost:7254/api/Car' \  -H 'accept: */*' \  -H 'Content-Type: application/json' \  -d '{  "licensePlate": "WPC199",  "modelId": 1,  "year": 1999  }'
```

Request URL

https://localhost:7254/api/Car

Server response

Code	Details
200	<div>Response body<div><pre>{  "id": 4,  "licensePlate": "WPC199",  "modelId": 1,  "model": null,  "year": 1999}</pre></div><div>Download</div></div>