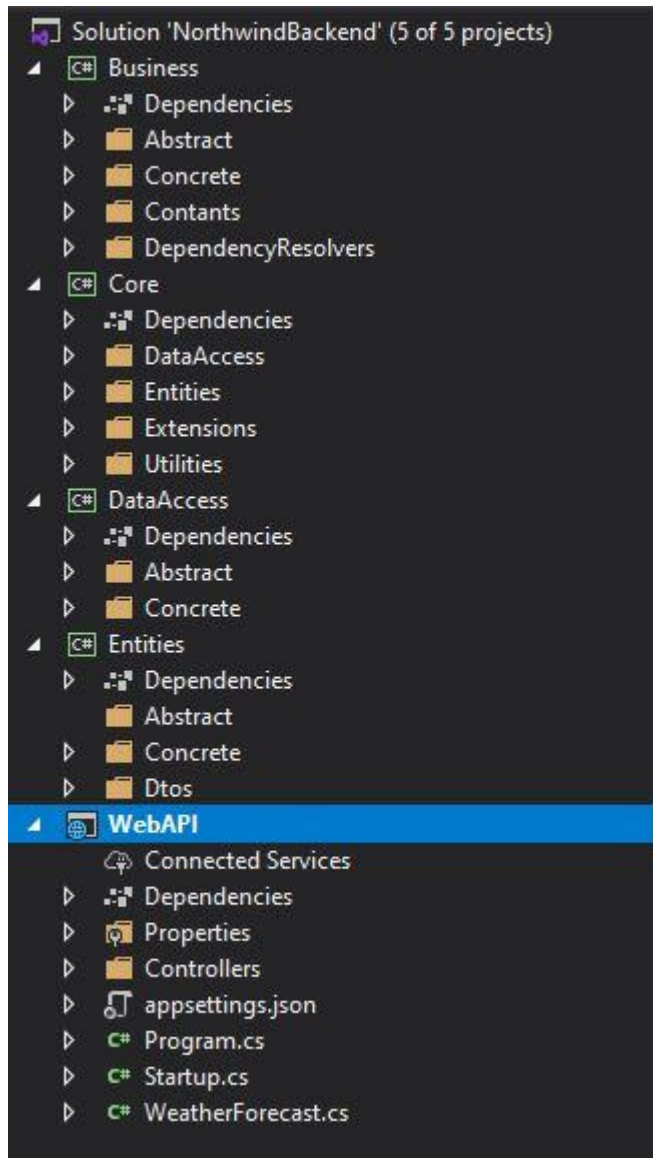


# .Net Core 3.1- JWT Authentication-Web API-MSSQL

## -----Backend-----



Proje Dosya Hiyerarşisi

Katmanlar ve API

```

[Route("api/[controller]")]
[ApiController]
1 reference
public class AuthController : Controller
{
    private IAuthService _authService;
    0 references
    public AuthController(IAuthService authService)
    {
        _authService = authService;
    }

    [HttpPost("login")]
    0 references
    public ActionResult Login(UserForLoginDto userForLoginDto)
    {
        var userToLogin = _authService.Login(userForLoginDto);
        if (!userToLogin.Success)
        {
            return BadRequest(userToLogin.Message);
        }
        var result = _authService.CreateAccessToken(userToLogin.Data);
        if (result.Success)
        {
            return Ok(result.Data);
        }
        return BadRequest(result.Message);
    }
}

```

## AuthController-login ActionResult

```

[HttpPost("register")]
0 references
public ActionResult Register(UserForRegisterDto userForRegisterDto)
{
    var userExists = _authService.UserExists(userForRegisterDto.Email);
    if (!userExists.Success)
    {
        return BadRequest(userExists.Message);
    }
    var registerResult = _authService.Register(userForRegisterDto, userForRegisterDto.Password);
    var result = _authService.CreateAccessToken(registerResult.Data);
    if (result.Success)
    {
        return Ok(result.Data);
    }
    return BadRequest(result.Message);
}
}

```

## AuthController-register ActionResult

Ekstra olarak .NET Core eğitim videoları izlerken syntax öğrenmek amaçlı yazılan ActionResultlar.

```
public class ProductController : ControllerBase
{
    private IProductService _productService;
    0 references
    public ProductController(IProductService productService)
    {
        _productService = productService;
    }

    [HttpGet("getall")]
    [Authorize(Roles = "Product.List")] // user login -- authorize olmalı
    0 references
    public IActionResult GetList()
    {
        var result = _productService.GetList();
        if (result.Success)
        {
            return Ok(result.Data);
        }
        return BadRequest(result.Message);
    }

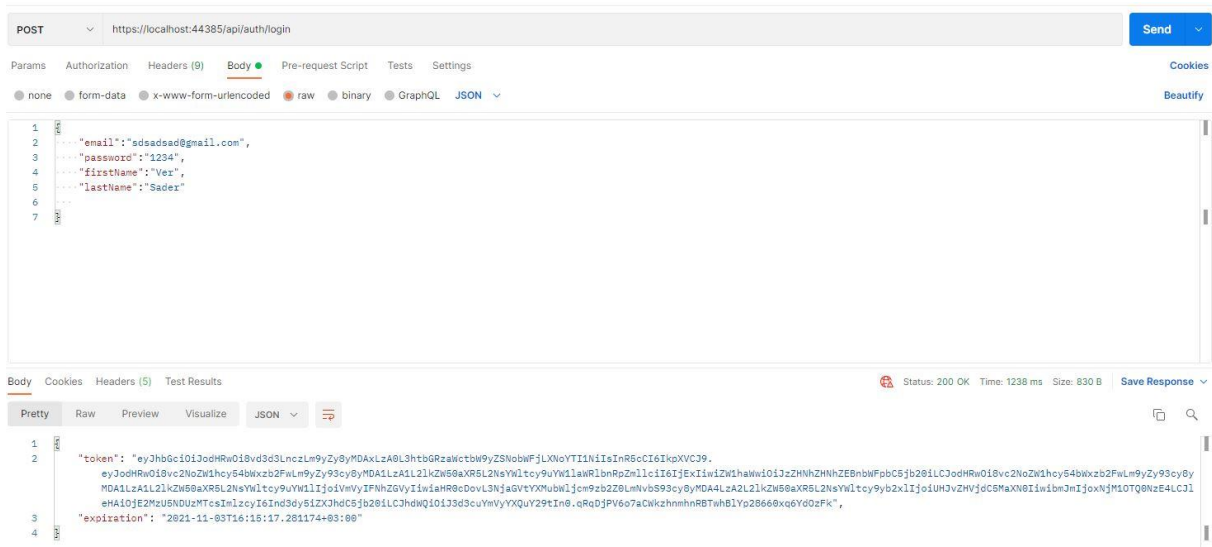
    [HttpGet("getlistbycategory")]
    0 references
    public IActionResult GetListByCategory(int categoryId)
    {
        var result = _productService.GetListByCategory(categoryId);
        if (result.Success)
        {
            return Ok(result.Data);
        }
        return BadRequest(result.Message);
    }
}
```

```
    [HttpPost("add")]
    0 references
    public IActionResult Add(Product product)
    {
        var result = _productService.Add(product);
        if (result.Success)
        {
            return Ok(result.Message);
        }
        return BadRequest(result.Message);
    }

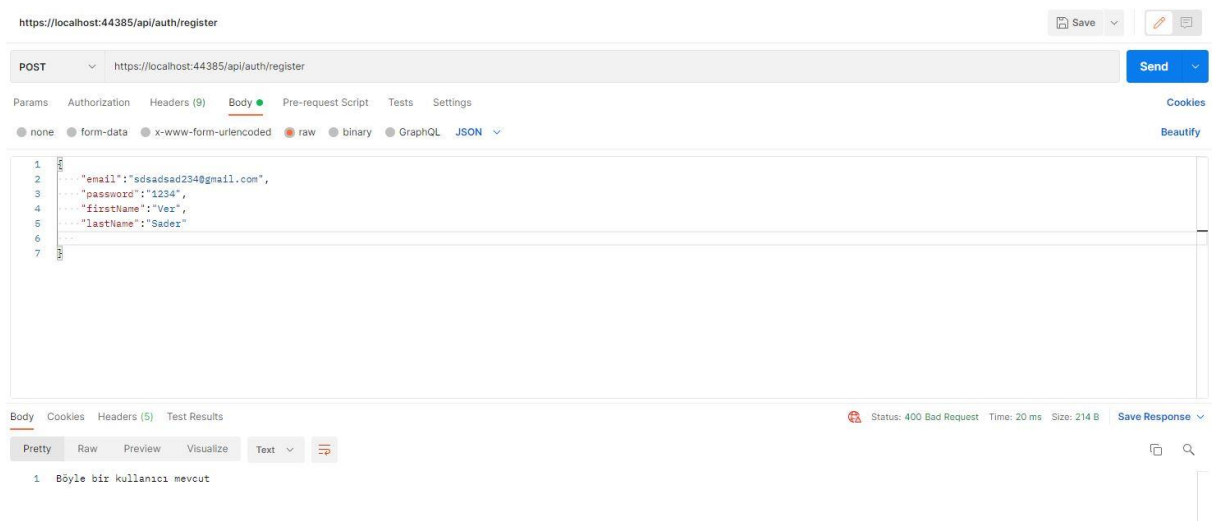
    [HttpPost("update")]
    0 references
    public IActionResult Update(Product product)
    {
        var result = _productService.Update(product);
        if (result.Success)
        {
            return Ok(result.Message);
        }
        return BadRequest(result.Message);
    }

    [HttpPost("delete")]
    0 references
    public IActionResult Delete(Product product)
    {
        var result = _productService.Delete(product);
        if (result.Success)
        {
            return Ok(result.Message);
        }
        return BadRequest(result.Message);
    }
}
```

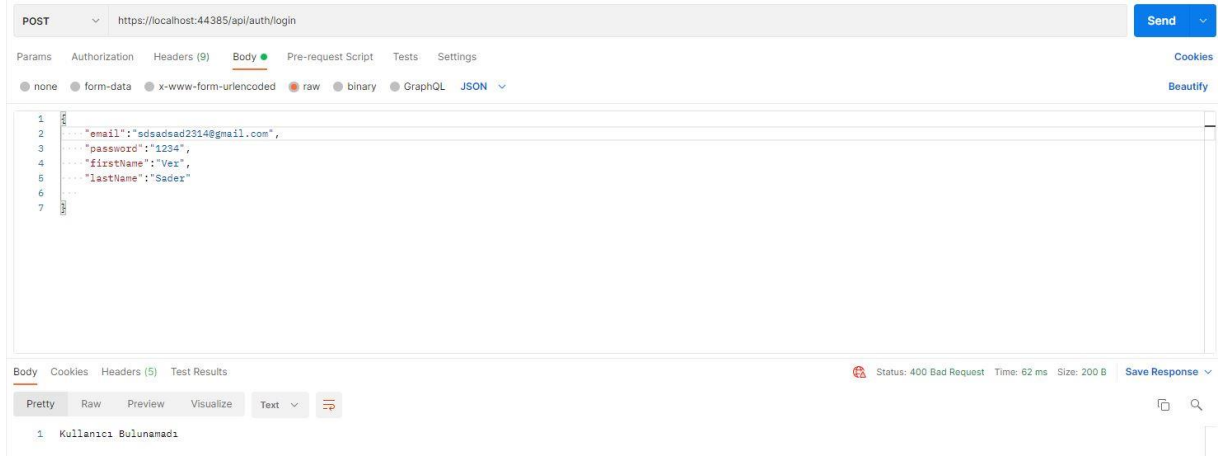
## Backend Testleri-Postman



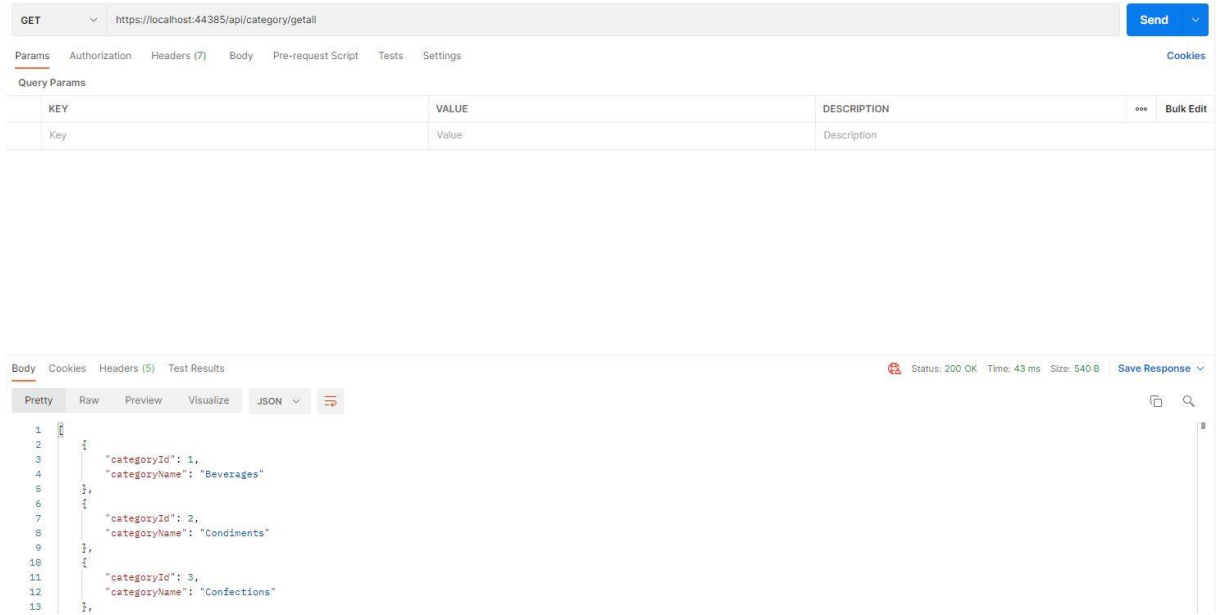
### Login işlemi sonucu token oluşturulması



Aynı bilgilerle kayıt edilince döndürülen hata mesajı



Yanlış bilgilerle girilince döndürülen hata mesajı



Diğer Action'lara örnek olarak "getall" action

GET https://localhost:44385/api/product/getall Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Headers 7 hidden

KEY	VALUE	DESCRIPTION
Authorization	Bearer eyJhbGciOiJIodHwOi8vd3d3LnczLm9yZy8yMDAxLzA0L3htbGRzaWctb...	
Key	Value	Description

Body Cookies Headers (5) Test Results Status: 200 OK Time: 49 ms Size: 10.76 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "productId": 1,
4     "productName": "Chai",
5     "categoryId": 1,
6     "quantityPerUnit": "18 boxes x 20 bags",
7     "unitPrice": 18.0000,
8     "unitsInStock": 39
9   },
10  {
11    "productId": 2,
12    "productName": "Chang",
13    "categoryId": 1,
14    "quantityPerUnit": "24 - 12 oz bottles",
15    "unitPrice": 19.0000,
16    "unitsInStock": 17
17  }
18 }
```

Giriş yapan kullanıcın tokenine ve rolüne göre actionlara erişebilme yetkisi

```
[HttpGet("getall")]
[Authorize(Roles = "Product.List")] // user login -- authorize olmalı
public IActionResult GetList()
{
    var result = _productService.GetList();
    if (result.Success)
    {
        return Ok(result.Data);
    }
    return BadRequest(result.Message);
}
```

## -----Frontend-----

Anasayfa Giriş Yap Kayıt Ol

Please sign in

[Sign In](#)

[Register](#)

Default Page'miz olarak Sign In page'i. Navbardan veya butondan kayıt olma sayfasına gidebilir.

Anasayfa Giriş Yap Kayıt Ol

Register

[Submit](#)

[Direct to Login Page](#)

Register Page'i.

Direkt olarak login page'ne geri dönebilir.

Anasayfa Giriş Yap Kayıt Ol

HOSGELDİNİZ

Basarili bir şekilde giriş yapıldıktan sonra “Hoşgeldiniz” main ekranı.

```

1
2 import React, { SyntheticEvent, useState } from "react";
3 import { Link, Redirect } from "react-router-dom";
4 const Register = () => {
5   const [FirstName, setName] = useState('');
6   const [LastName, setLastName] = useState('');
7   const [Email, setEmail] = useState('');
8   const [Password, setPassword] = useState('');
9   const [redirect, setRedirect] = useState(false);
10  const data = {
11    Email,
12    Password,
13    FirstName,
14    LastName
15  }
16  }
17
18  const submit = async (e: SyntheticEvent) => {
19    e.preventDefault();
20    await fetch('https://localhost:44385/api/auth/register', {
21      method: 'POST',
22      headers: { 'Content-Type': 'application/json' },
23      body: JSON.stringify(data)
24    });
25  });
26
27  setRedirect(true);
28
29  }
30
31  if (redirect) {
32    return <Redirect to="/Home" />
33  }
34
35
36

```

Register-Backend-Frontend iletişimi

API'mizle iletişim kurduğumuz fetch'miz.



```
1  import React, { SyntheticEvent, useState } from "react";
2  import { Link, Redirect } from "react-router-dom";
3
4  const Home = () => {
5
6      const [Email, setEmail] = useState('');
7      const [Password, setPassword] = useState('');
8      const [redirect, setRedirect] = useState(false);
9      const data = {
10         Email,
11         Password
12     }
13
14
15     const submit = async (e: SyntheticEvent) => {
16         e.preventDefault();
17         await fetch('https://localhost:44385/api/auth/login', {
18             method: 'POST',
19             headers: { 'Content-Type': 'application/json' },
20             body: JSON.stringify(data)
21         });
22
23     });
24
25     setRedirect(true);
26
27
28 }
29 if (redirect) {
30     return <Redirect to ="/Login" />;
31 }
32
33
```

Login-Backend-Frontend iletişimi

API'mizle iletişim kurduğumuz fetch'miz.