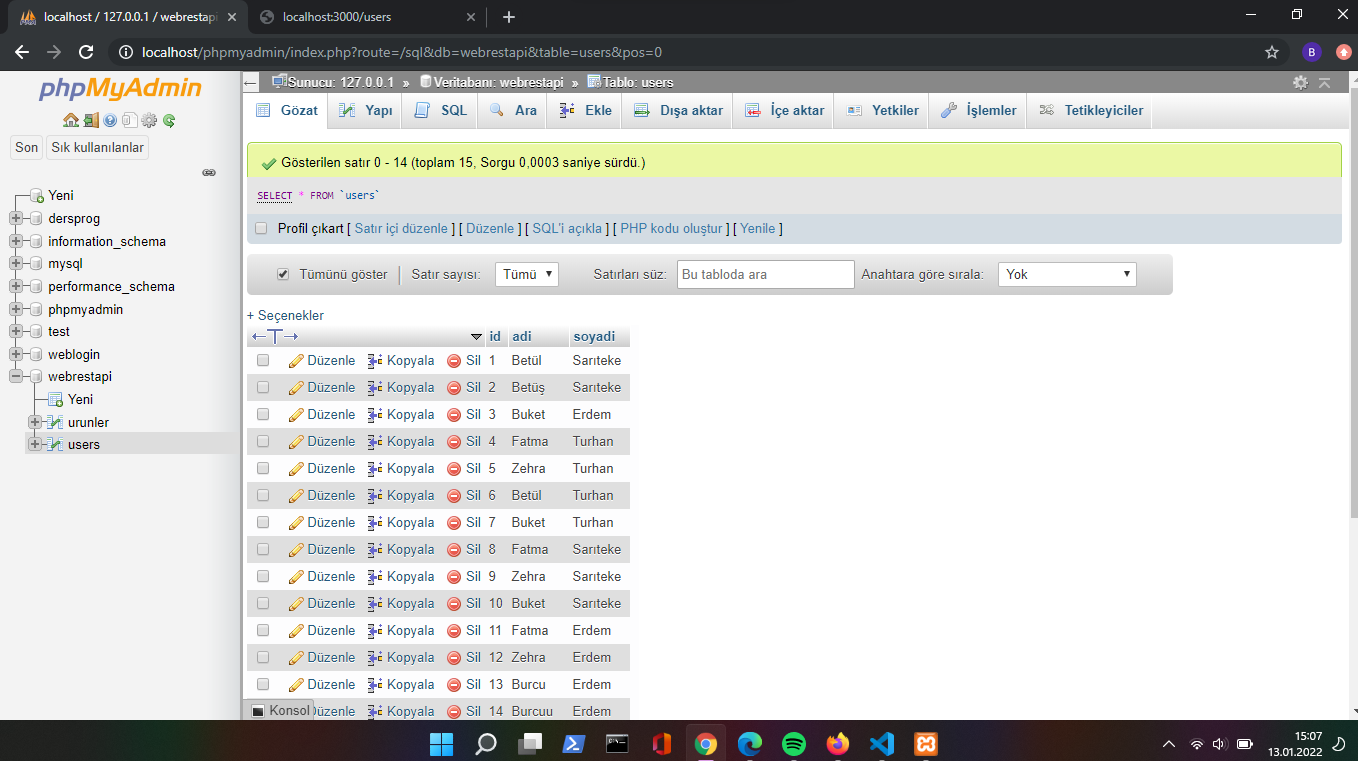
**İLERİ WEB EXPRESS MYSQL**

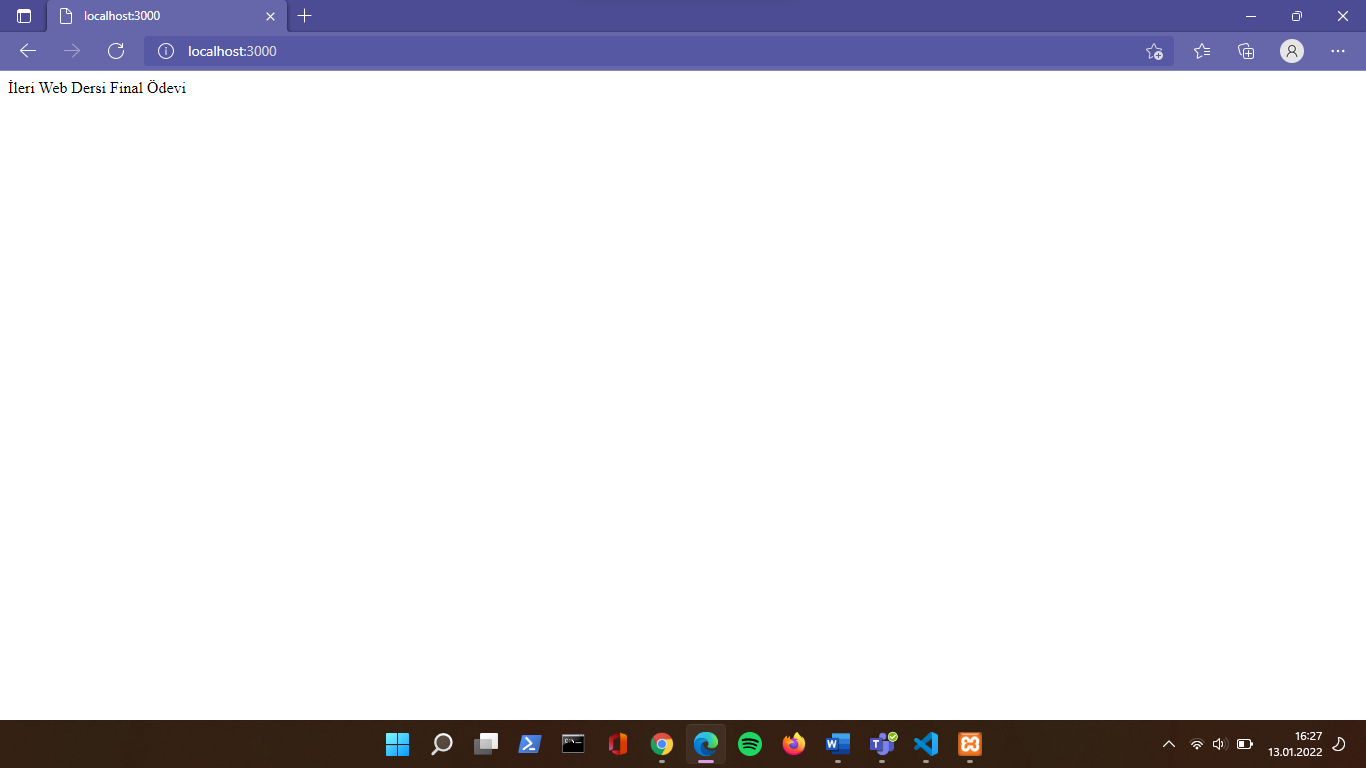
**CRUD İŞLEMLERİ**

* Betül Sarıteke

**Veritabanı MySql:**

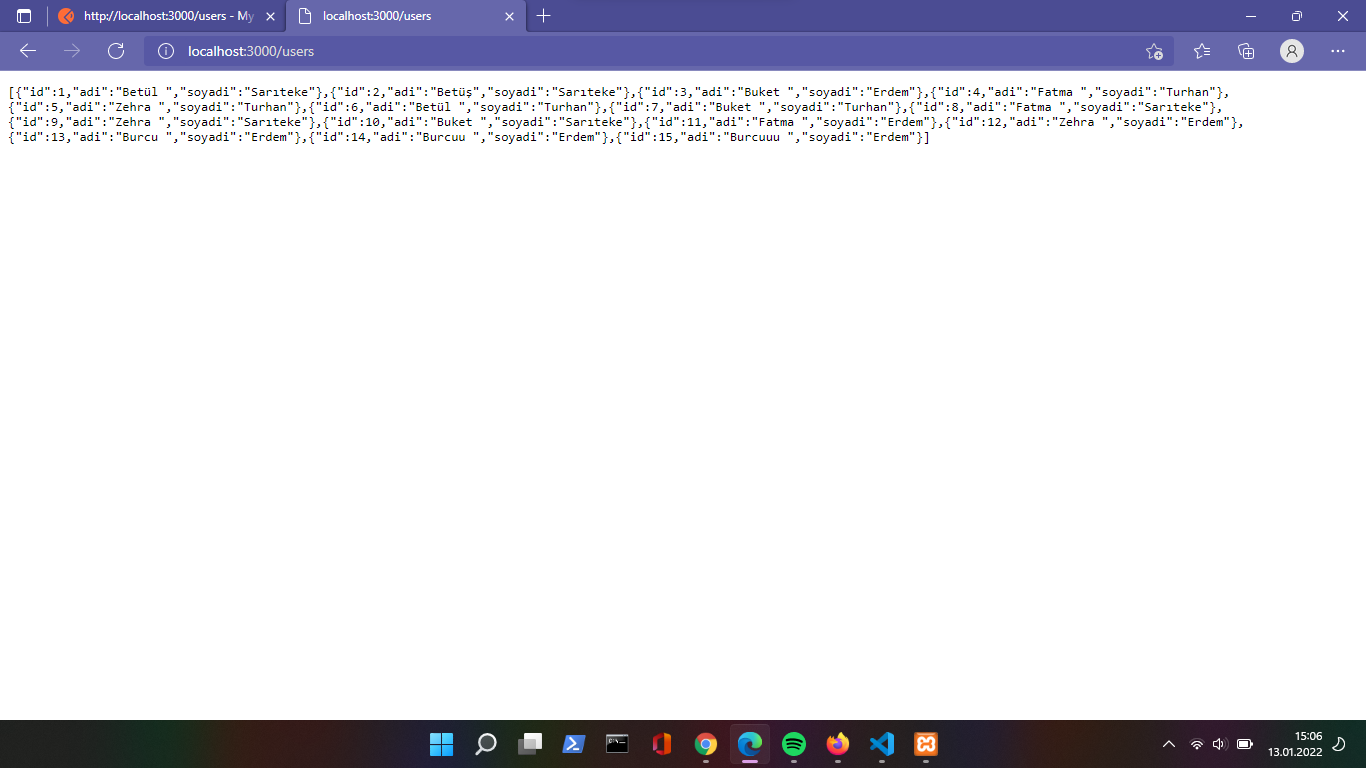


**Browser Görüntüsü:**

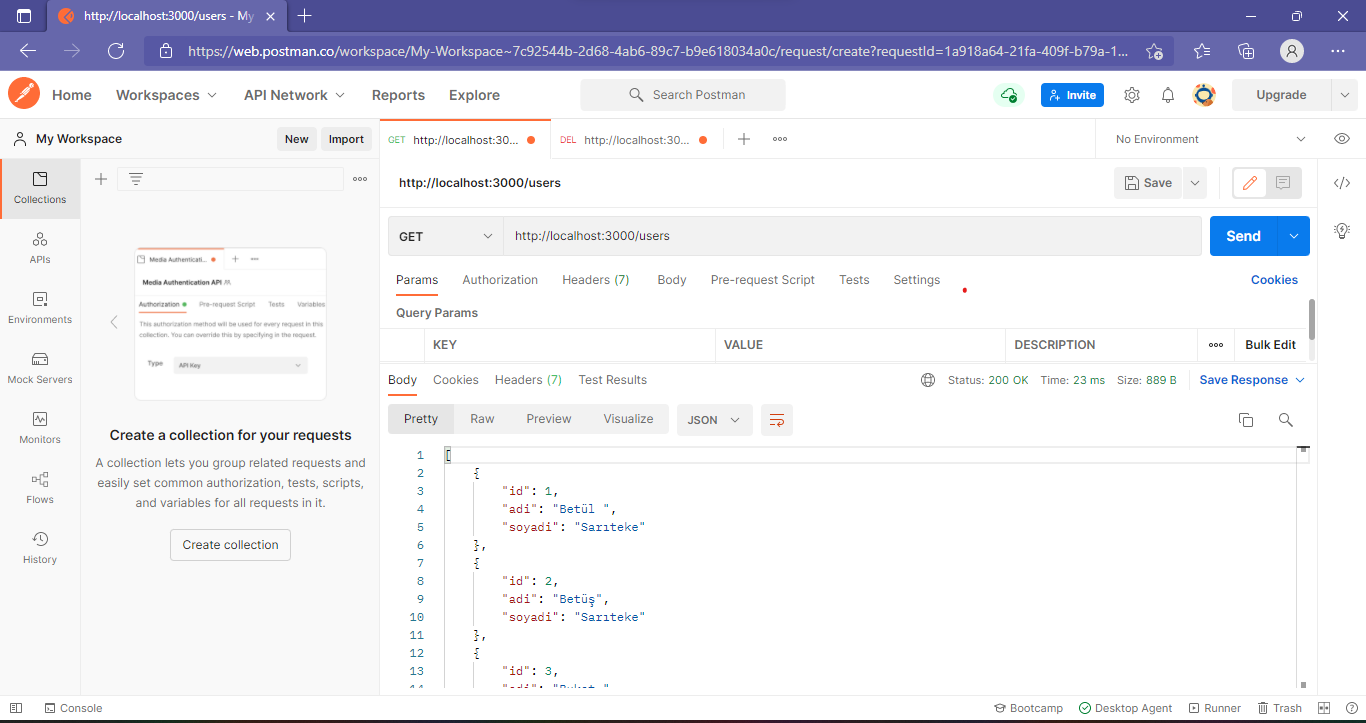
<http://localhost:3000/>

**Get işlemi tüm veriler için:**

<http://localhost:3000/users>

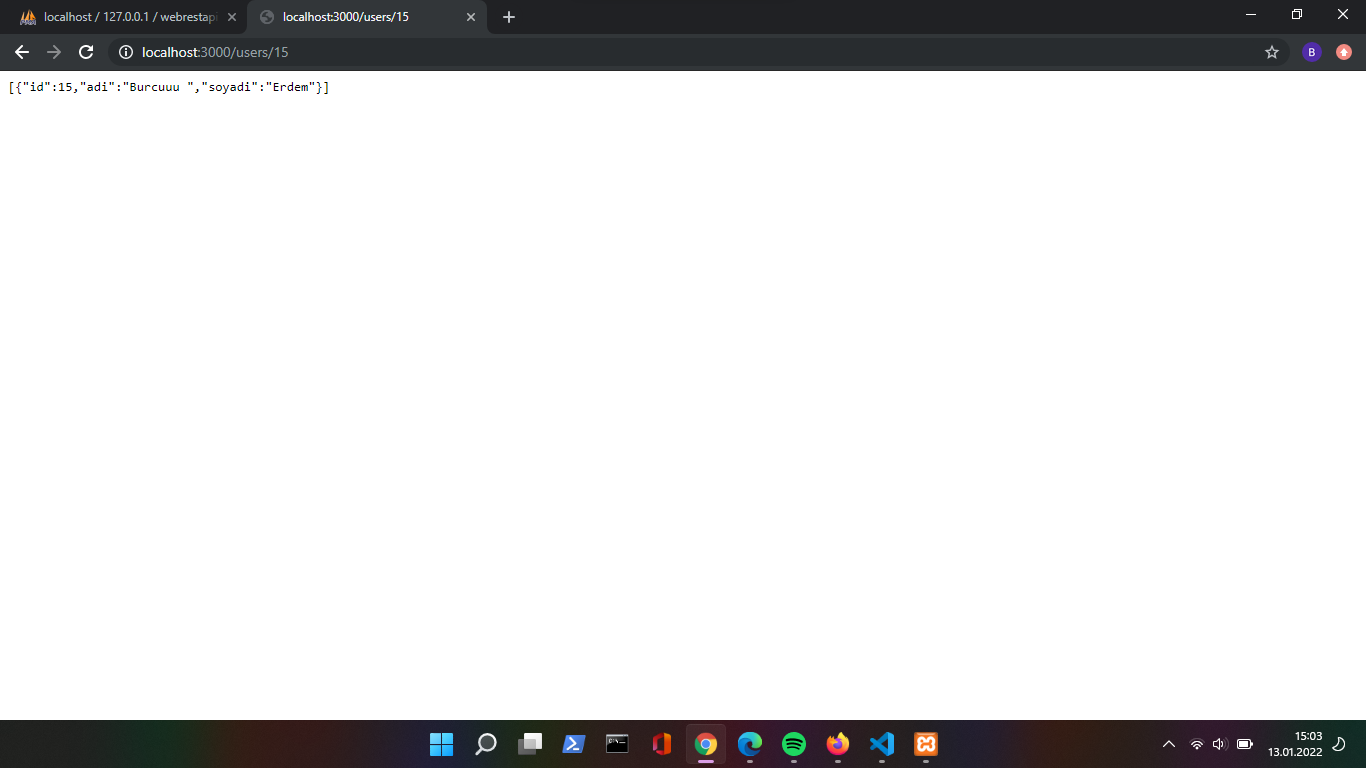


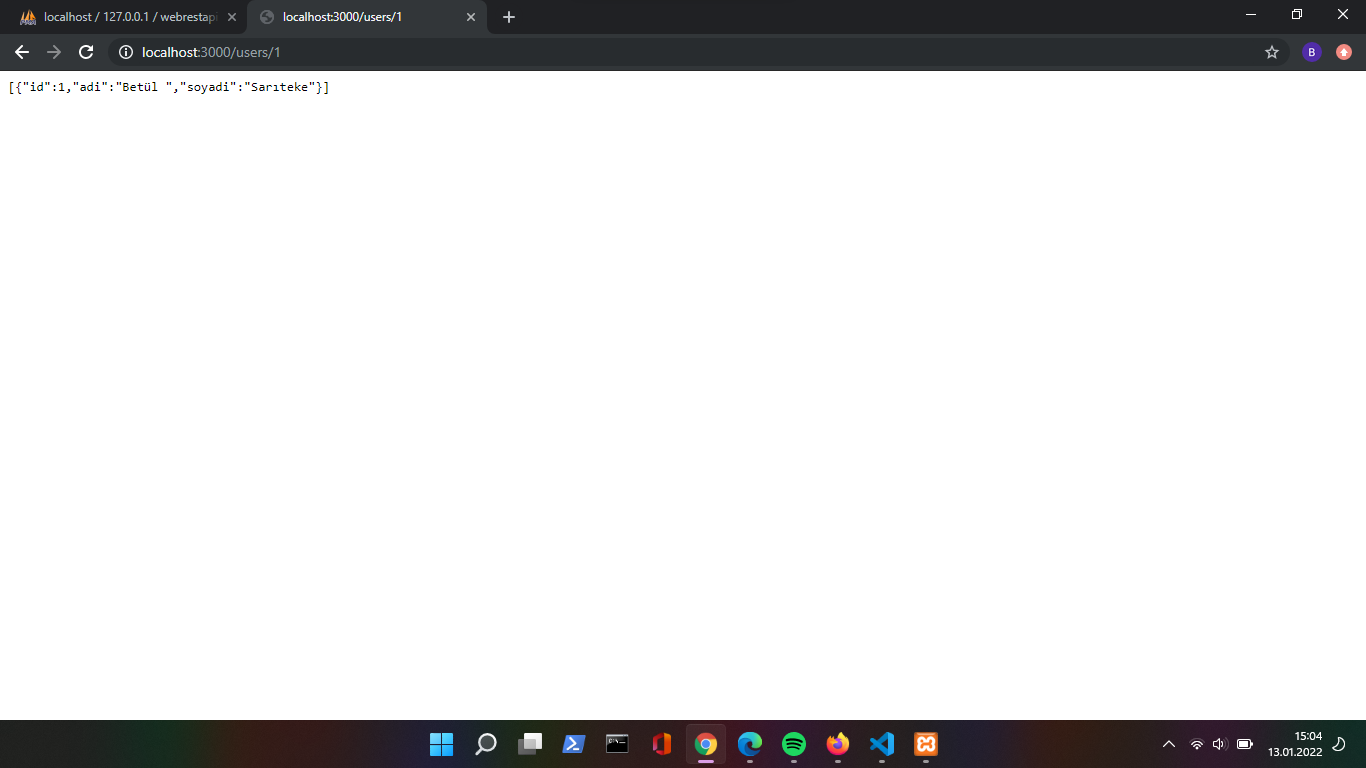
**Get işlemi tüm veriler için (Postman):**



**Get işlemi belirli bir id ile veri çekme:**

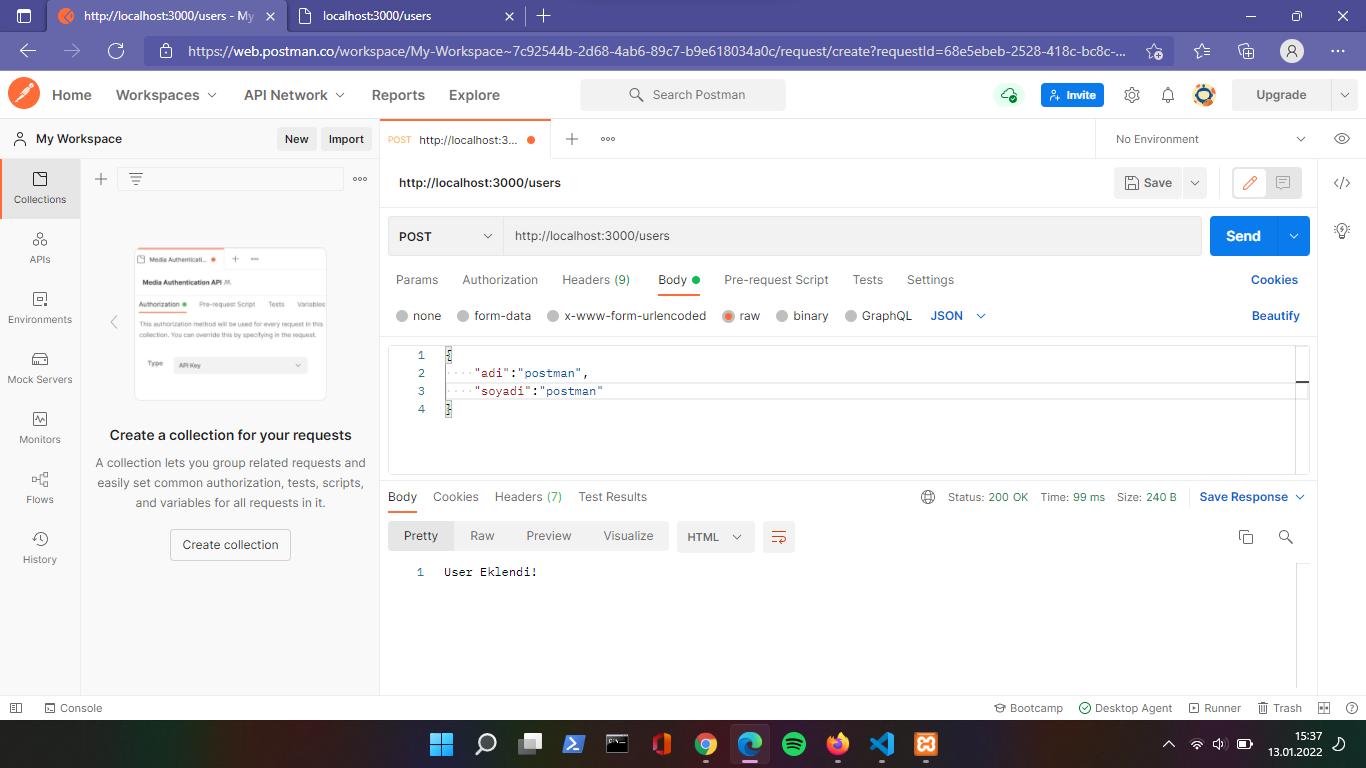
<http://localhost:3000/users/:id>





**Post işlemleri (Postman):**

<http://localhost:3000/users>



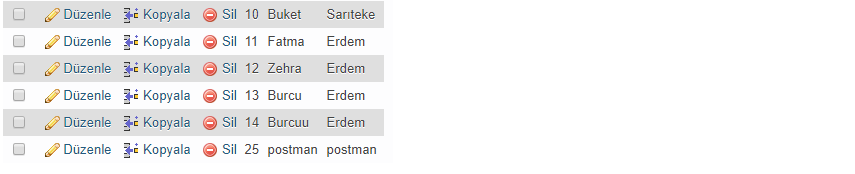
**Sql:**

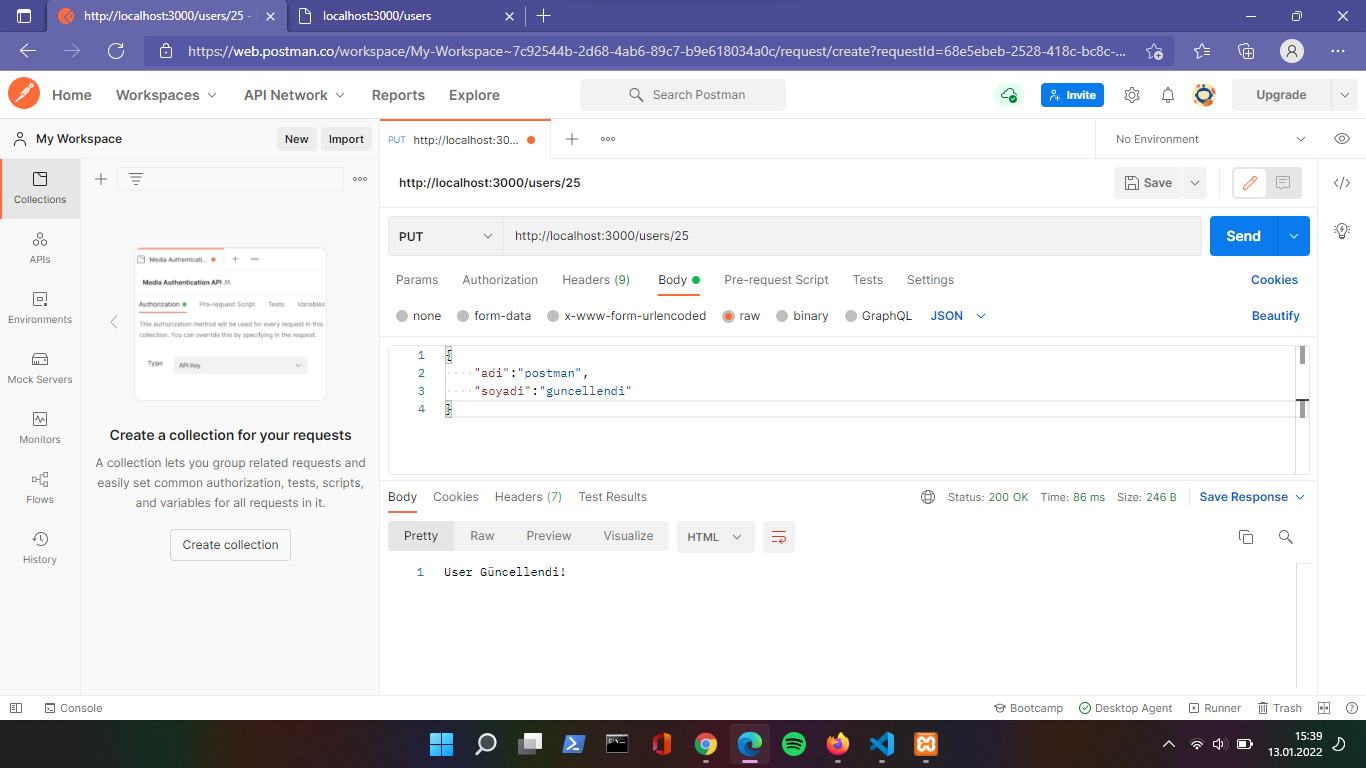


**Put işlemi (Postman):**

<http://localhost:3000/users/:id>

**Sql:**



**Postman put işlemi:** 

**Sql:**



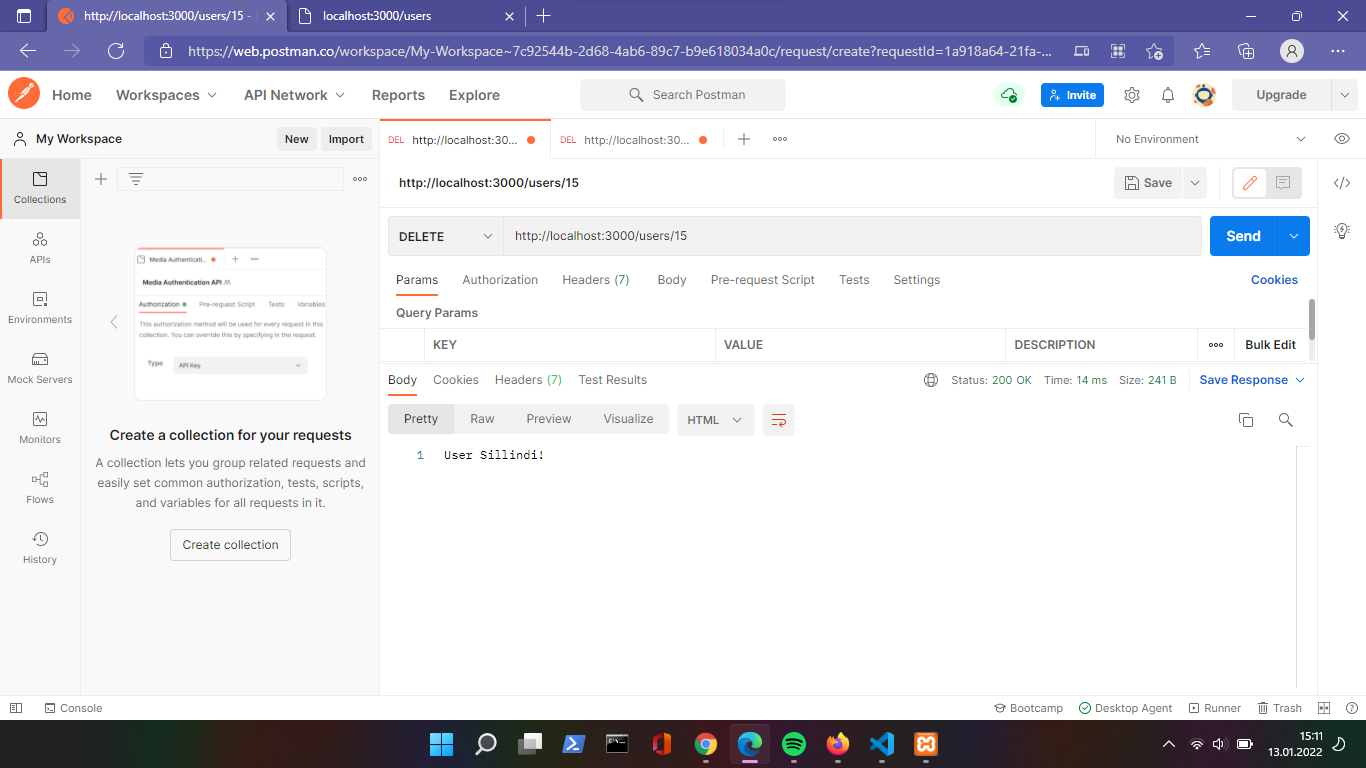
**Delete işlemi (Postman):**

<http://localhost:3000/users/:id>

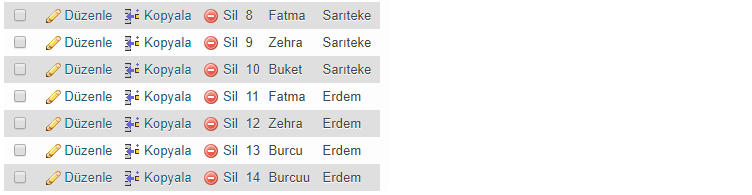
**Sql:**

****

**Postman delete işlemi:**



**Sql :**



# APİ KODLARI

**.env**

DATABASE = webrestapi

DATABASE\_HOST = localhost

DATABASE\_USER = root

DATABASE\_PASSWORD =

**App.js**

// KULLANILAN KÜTÜPHANELERİ EKLİYORUZ

//express mysql nodemon morgan body-parser

const express = require("express");

const path = require("path");

const morgan = require("morgan");//loglama işlemi için kullanılır

const bodyParser = require("body-parser"); //alınan verilri json formatı için

const app = express();

app.use(bodyParser.urlencoded({ extended: false }));

app.use(bodyParser.json());

app.use(morgan('short'));//kullanıcının yapmış olduğu işlem loglarını alır terminale yazar

const router = require("./routes/user.js")

app.use(router);

app.get('/', (*req*, *res*) => {

    //console.log("Ana Sayfa"),

*res*.send("İleri Web Dersi Final Ödevi")

});

//localhost server başlatmak için

app.listen(3000, () => {

    console.log("Server Started On Port 3000 !");

    // http://localhost:3000/

    // başlatmak için : nodemon app.js

});

**Routers/Users.js**

//kullanılan kütüphaneler import edildi

const express = require("express");

const mysql = require("mysql");

const dotenv = require("dotenv");

const path = require("path");

const bodyParser = require("body-parser");

dotenv.config({ path: './.env' }); //veritabanı bilgilerini alır

const app = express();

const router = express.Router();

//post verilerini json formatına çevirmek için

app.use(bodyParser.urlencoded({ extended: false }));

app.use(express.json());

//veritabanı bilgileri

const db = mysql.createConnection({

    host: process.env.DATABASE\_HOST,

    user: process.env.DATABASE\_USER,

    password: process.env.DATABASE\_PASSWORD,

    database: process.env.DATABASE

});

//veritabanı bağlantısı

db.connect((*error*) => {

    if (error) {

        console.log(error)

    } else {

        console.log("Mysql Connected!")

    }

})

//tüm kullanıcılar için get

router.get('/users', (*req*, *res*) => {

    const sqlQuery = "SELECT \* FROM users";

    db.query(sqlQuery, (*error*, *rows*) => {

        if (error) {

            console.log(error);

            res.sendStatus(500)

            res.end();

        } else {

            const users = rows.map((*row*) => {

                return {

                    id: row.id,

                    adi: row.adi,

                    soyadi: row.soyadi

                }

            });

            res.json(users)

        }

    });

});

//id verisine göre kullanıcı için get

router.get('/users/:id', (*req*, *res*) => {

    const userId = *req*.params.id;

    const sqlQuery = "SELECT \* FROM users WHERE id= ? ";

    db.query(sqlQuery, [userId], (*error*, *row*) => {

        if (*error*) {

            console.log(*error*);

*res*.sendStatus(500)

*res*.end();

        } else {

            const users = *row*.map((*row*) => {

                return {

                    id: *row*.id,

                    adi: *row*.adi,

                    soyadi: *row*.soyadi

                }

            });

*res*.json(users)

        }

    });

});

//user ekleme işlemi post ile

router.post("/users", (*req*, *res*,) => {

    console.log("Kullanıcı post işlemi yapıldı!")

    const adi = *req*.body.adi;

    const soyadi= *req*.body.soyadi;

    const sqlQuery = "INSERT INTO users (adi, soyadi) VALUES (?,?)";

    db.query(sqlQuery, [adi, soyadi], (*error*, *results*) => {

        if (*error*) {

            console.log(*error*);

*res*.sendStatus(500)

            return

        }

        console.log("User Eklendi!");

*res*.send("User Eklendi!")

*res*.end();

    })

});

//user silme işlemi delete

router.delete("/users/:id", (*req*, *res*,) => {

    console.log("Kullanıcı delete işlemi yapıldı!")

    const userId = *req*.params.id;

    const sqlQuery = " DELETE FROM users WHERE id= ?";

    db.query(sqlQuery, userId, (*error*, *results*) => {

        if (*error*) {

            console.log(*error*);

*res*.sendStatus(500)

            return

        }

        console.log("User Silindi!");

*res*.send("User Sillindi!")

*res*.end();

    })

});

//user güncelleme işlemi put

router.put("/users/:id", (*req*, *res*,) => {

    console.log("Kullanıcı put işlemi yapıldı!")

    const userId = *req*.params.id;

    const { adi, soyadi } = *req*.body;

    console.log(*req*.body);

    const sqlQuery = "UPDATE users SET adi=?, soyadi=? WHERE id=?";

    db.query(sqlQuery, [adi, soyadi,userId], (*error*, *results*) => {

        if (*error*) {

            console.log(*error*);

*res*.sendStatus(500)

            return

        }

        console.log("User Güncellendi!");

*res*.send("User Güncellendi!")

*res*.end();

    })

});

*module*.*exports* = router