



REACT

11 FETCH AND JQUERY AJAX CALLS



SYLLABUS

- 01 Fundamentals And Installation
- 02 Components And Props
- 03 State
- 04 Handling Events
- 05 SetState And Lifecycle
- 06 Lists And Keys
- 07 Forms
- 08 PropTypes
- 09 Refs
- 10 React Router v4
- **11 Fetch And jQuery Ajax calls**

RESTFUL API AND FETCH GET

- כאשר נבצע קריאה ל RESTFUL API ניתן לעשות זאת ע"י שימוש בפונקציית FETCH של הדפדפן.

```
fetch("http://localhost:63597/home/GetStudentIfExists/1")  
  .then(res => res.json())  
  .then(  
    (result) => {  
      console.log(result);  
      console.log("fetch GET="+result.firstName);  
    },  
    (error) => {  
      console.log(error);  
    }  
  );
```

```
public IActionResult GetStudentIfExists(int id)  
{  
    if (id == 1)  
    {  
        return Json(new Person()  
        {  
            FirstName = "avi",  
            LastName = "coehn",  
            Age = 30  
        });  
    }  
    return Json(null);  
}
```

```
public class Person  
{  
    public string FirstName { get; set; }  
    public string LastName { get; set; }  
    public int Age { get; set; }  
}
```

RESTFUL API AND FETCH POST

```
fetch('http://localhost:63597/home/MyAction', {
  ↑ method: 'post',
    headers: new Headers({
      'Content-Type': 'application/json;',
    }),
    body: JSON.stringify({ name: 'avi', address: 'address1' })
  })
  .then(res => res.json())
  .then(
    (result) => {
      console.log("fetch POST= " + result);
    },
    (error) => {
      console.log("err post=" + error);
    }
  );
```

- כאשר נבצע קריאה ל RESTFUL API ניתן לעשות זאת ע"י שימוש בפונקציית FETCH של הדפדפן.

- וניתן גם לשלוח פרמטרים

- חשוב מאוד לשים לב שחובה להוסיף לקובץ ה- Startup.cs ב MVC.core שתי תוספות:

1. services.AddCors();
2. app.UseCors(options => options.AllowAnyOrigin().AllowAnyHeader().AllowAnyMethod().AllowCredentials());

```
[HttpPost]
public IActionResult MyAction([FromBody] Person2 person)
{
    var myName = person.Name;
    return Json(myName);
}

public class Person2
{
    public string Name { get; set; }
    public string Address { get; set; }
}
```

```
public void ConfigureServices(IServiceCollection)
{
    services.AddMvc();
    services.AddScoped<IStudentsDB, InMemor
    services.AddCors();
}
```

```
public void Configure(IApplicationBuilder app, IHostingEnvironment env, ILoggerFactory loggerFactory)
{
    app.UseCors(options => options.AllowAnyOrigin().AllowAnyHeader().AllowAnyMethod().AllowCredentials());

    loggerFactory.AddConsole();
}
```

RESTFUL API AND \$.AJAX GET/POST

```
var data = {
  FirstName: 'Andrew',
  LastName: 'Lock',
  Age: 31
}

$.ajax({
  type: 'POST',
  url: 'http://localhost:63597/home/PersonAsJson',
  dataType: 'json',
  contentType: 'application/x-www-form-urlencoded; charset=utf-8',
  data: data,
  success: function (result) {
    console.log("$.AJAX POST=" + result);
    console.log("$.AJAX POST=" + result.firstName);
  }
});
```

- כאשר נבצע קריאה ל RESTFUL API ניתן לעשות זאת ע"י שימוש בפונקציית \$.ajax.

- יש צורך להתקין JQUERY ע"י

npm install jquery

- import \$ from 'jquery';

```
[HttpPost]
public IActionResult PersonAsJson(Person person)
{
    string str = person.FirstName;
    return Json(person);
}
```

WEB SERVICE AND \$.AJAX

```
$.ajax({  
  ↑ url: "http://ruppinmobile.ac.il.preview26.livedns.co.il/site01/usersws.asmx/GetAllUsers",  
    dataType: "json",  
    type: "POST", //use only POST!  
    data: "{}",  
    contentType: "application/json; charset=utf-8",  
    error: function (jqXHR, exception) {  
      alert(jqXHR + exception);  
    },  
    success: function (data) {  
      console.log(data.d);  
      var users = JSON.parse(data.d);  
      users.forEach(user => {  
        console.log(user);  
        console.log(user.Name);  
      });  
    }  
  });  
});
```

• כאשר נבצע קריאה ל WEB SERVICE ניתן לעשות זאת ע"י שימוש בפונקציית \$.ajax של JQUERY.

• יש צורך להתקין JQUERY ע"י
npm install jquery