

Yazılım Geliştirme Ortam ve Araçları

Ders Class'ı

```
1 package tr.edu.cukurova.CukurovaUniversitesi;
2
3 public class Ders {
4
5
6
7     public Ders(String adi, String id) {
8         super();
9         this.adi = adi;
10    }
11
12    public Ders(String adi) {
13        super();
14        this.adi = adi;
15    }
16
17    private String adi;
18
19
20    public String getAdi() {
21        return adi;
22    }
23 }
```

Ders class'ında dersimizin adı için private bir adi değişkeni ekledik.

Daha sonra class'a sağ tıklayarak source kısmından Generate constructor using fields'a basarak

değişkenimizi seçtik ardından tekrar sağ tıklayarak source kısmında Getters and setters'a girip getter'ı seçip devam ettik.

DersVeriServisi Class'ı

```
1 package tr.edu.cukurova.CukurovaUniversitesi;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class DersVeriServisi {
7
8     private static final List<Ders> DERSLER = new ArrayList<>();
9     static {
10         DERSLER.add(new Ders("Matematik"));
11         DERSLER.add(new Ders("Türkçe"));
12         DERSLER.add(new Ders("Fen"));
13         DERSLER.add(new Ders("Müzik"));
14     }
15     public static List<Ders> dersleriGetir(){
16         return DERSLER;
17     }
18     public static void dersEkle(Ders ders) {
19         DERSLER.add(ders);
20     }
21     public static void dersSil(String adi) {
22     }
23
24
25 }
26
```

Burada ise listemizi oluşturup dört adet ders ekledik.

dersleriGetir metodumuzda listedeki dersleri döndürdük.

dersEkle metodumuzda ise DersApi kısmında yazacağımız komut ile ders eklenmesini sağladık.

DersApi Class'ı

```
1 package tr.edu.cukurova.CukurovaUniversitesi;
2
3 import java.util.*;
4
5 import org.springframework.web.bind.annotation.*;
6 @RestController
7 @RequestMapping("/ders")
8 public class DersApi {
9
10
11     @GetMapping("/listele")
12     public List<Ders> listele(){
13         return DersVeriServisi.dersleriGetir();
14     }
15
16     @GetMapping("/ekle")
17     public Ders dersEkle(@RequestParam String adi) {
18         Ders ders= new Ders(adi);
19         DersVeriServisi.dersEkle(ders);
20         return ders;
21     }
22
23 }
```

RequestMapping("/ders") ile localhost'tan sonra ne yazıp ulaşabileceğimizi belirledik.

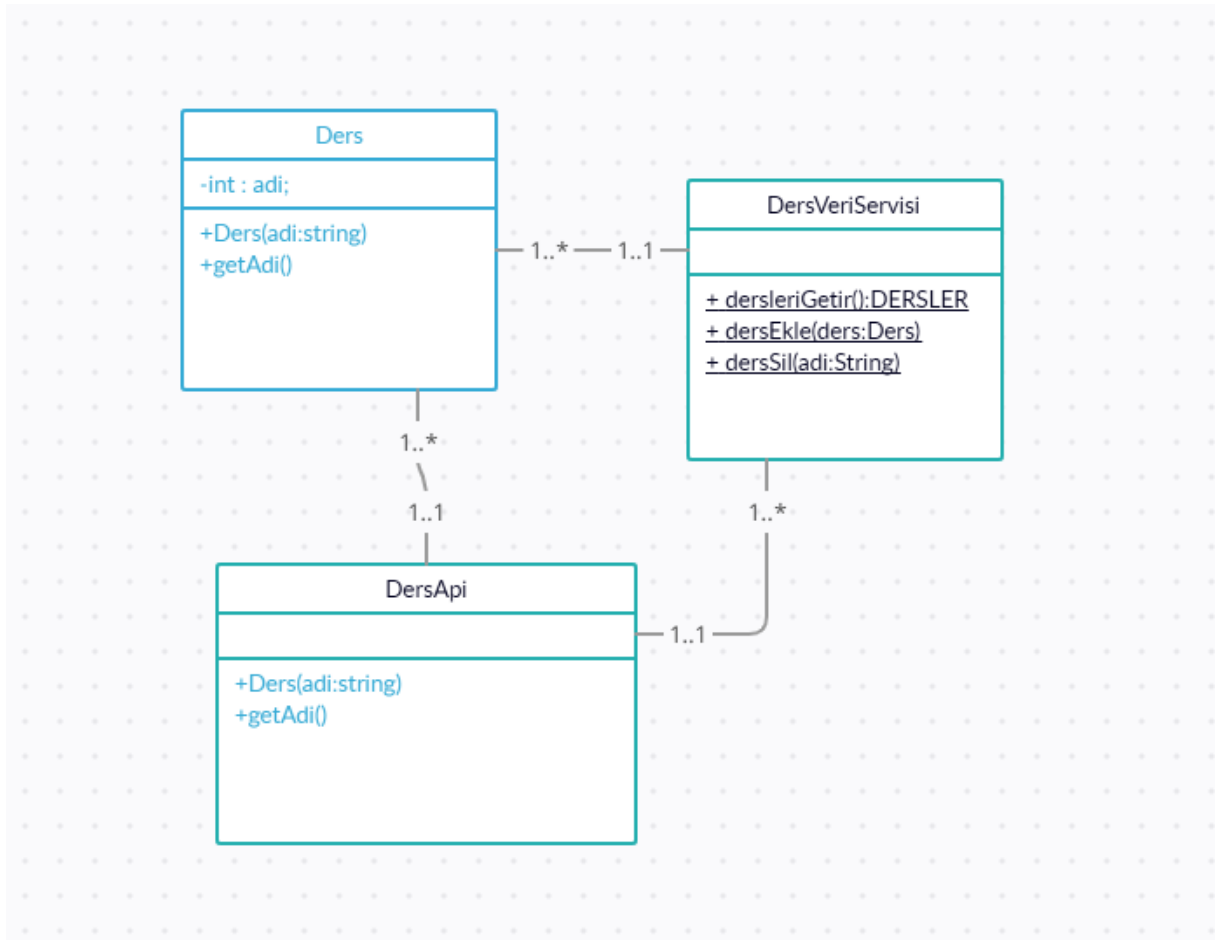
GetMapping("/listele") ile /ders sonrası ne yazacağımızı belirleyip bu sayfada çıkmasını istediğimiz ders listesini DersVeriServisi class'ından çektik.

@GetMapping("ekle") ile eklemek için /ders sonrası ne yazacağımızı belirleyip

@RequestParam string adi ile girmemiz gereken parametreyi yazdık.

Ardından girilen dersi döndürdük.

UML diyagramı



Postman Listeleme

http://localhost:8070/ders/listele

Save

Send

GET http://localhost:8070/ders/listele

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Headers 6 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
Key	Value	Description			

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 9 ms Size: 251 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "adi": "Matematik"
3 },
4 {
5   "adi": "Türkçe"
6 },
7 {
8   "adi": "Fen"
9 },
10 {
11   "adi": "Müzik"
12 },
13 {
14   "adi": "ders6"
15 },
16 }
```

Postman Ekleme

http://localhost:8070/ders/ekle?adi=ders6

Save

Send

GET http://localhost:8070/ders/ekle?adi=ders6

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> adi	ders6			
Key	Value	Description		

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 26 ms Size: 179 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "adi": "ders6"
3 }
```

Jmeter

Thread Group

Name:

Thread Group

Comments:

Action to be taken after a Sampler error

☒ Continue ☐ Start Next Thread Loop ☐ Stop Thread ☐ Stop Test ☐ Stop Test Now

Thread Properties

Number of Threads (users):

100

Ramp-up period (seconds):

1

Loop Count:

☐ Infinite

1

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

Loop Controller

Name:

Loop Controller

Comments:

Loop Count:

☐ Infinite

10

Counter

Name:

Counter

Comments:

Starting value

15

Increment

5

Maximum value

Number format

Exported Variable Name

cnt

☐ Track counter independently for each user

☐ Reset counter on each Thread Group Iteration

HTTP İsteği

İsim:

dersListele

Yorumlar:

Basic

Advanced

Ağ Sunucusu

Protokol (http):

http

Sunucu İsmi veya IP:

localhost

Port Numarası:

8070

HTTP İsteği

GET

Yol:

/ders/listele

İçerik kodlaması:

☐ Otomatik Olarak Yönlendir

☒ Yönlendirmeleri İzle

☒ CanlıTut (KeepAlive) kullan

☐ HTTP POST için multipart/form-data kullan

☐ Browser-compatible headers

Parameters

Body Data

Files Upload

İstekle parametreleri gönder:

İsim:	Değer	Kodlama?	Content-Type	Eşleniyorsa içer?
-------	-------	----------	--------------	-------------------

Detail

Ekle

Add from Clipboard

Sil

Up

Down

