

# RAPOR

## Spring Boot Projesi

```
@RestController
@RequestMapping("/lessons")
public class LessonController {

    private HashMap<Integer, String> lessonMap = new HashMap<>();

    @GetMapping("/all")
    public HashMap<Integer, String> getLessonMap() { return lessonMap; }

    @PostMapping("/add/{id}/{lesson}")
    public String createLesson(@PathVariable String id, @PathVariable String lesson){
        try {
            Integer parseId = Integer.valueOf(id);
            String put = lessonMap.put(parseId, lesson);
            return "ders başarılı bir şekilde eklendi";
        }
        catch (Exception exception) {
            return "Girilen id değeri numeric olmalıdır. Ders eklenemedi";
        }
    }

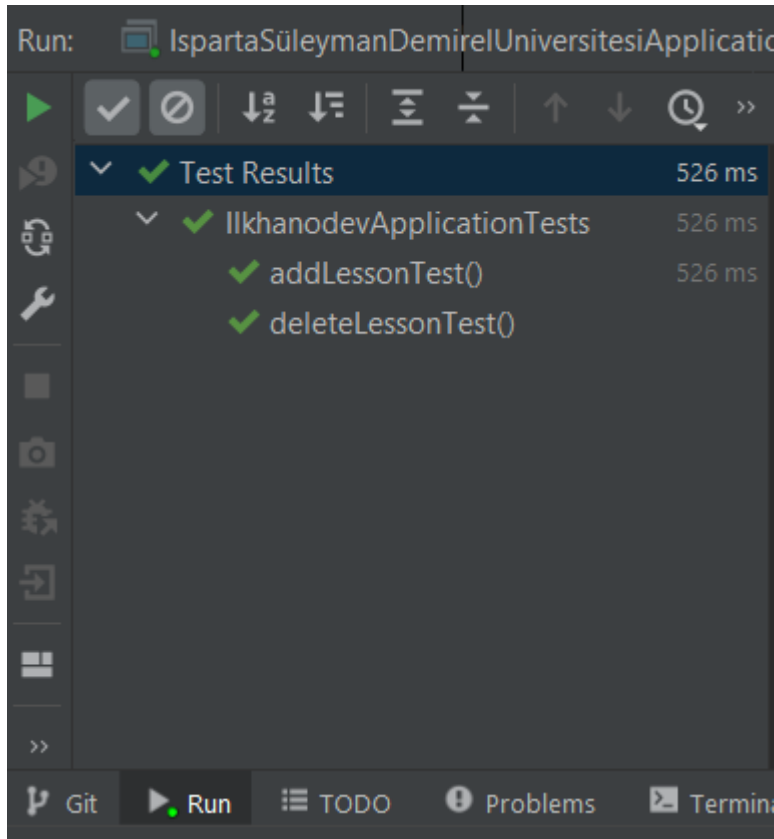
    @DeleteMapping("/delete/{id}")
    public String deleteLesson(@PathVariable String id) {
        try {
            Integer parseId = Integer.valueOf(id);
            lessonMap.remove(parseId);
            return "ders başarılı bir şekilde silindi";
        }
    }
}
```

```
        }
        catch (Exception exception) {
            return "Girilen id değeri numeric olmalıdır. Ders silinemedi";
        }
    }

    @PostConstruct
    public void initLessons() {
        lessonMap.put(1, "Genel matematik");
        lessonMap.put(2, "Veritabanı programlama");
        lessonMap.put(3, "C programlama");
        lessonMap.put(4, "Programcılığa giriş");
        lessonMap.put(5, "Veri yapıları");
        lessonMap.put(6, "Web programlama");
        lessonMap.put(7, "İnkılap");
        lessonMap.put(8, "İngilizce");
        lessonMap.put(9, "Türk dili");
        lessonMap.put(10, "Nesne yönelimli java programlama");
        lessonMap.put(11, "Mobil yazılımlar");
        lessonMap.put(12, "Blockchain teknoloji temelleri");
        lessonMap.put(13, "Yazılım testi ve otomasyonu");
        lessonMap.put(14, "Yazılım geliştirme ortam ve araçları");
    }
}
```

Java kullanarak geliştirdiğimiz Spring Boot projesi.

## Junit Test



Junitte edindiğimiz test sonuçları

## Jenkins

```
C:\Windows\System32\cmd.exe - java -jar jenkins.war.
C:\Users\User\Desktop\jenkins war>java -jar jenkins.war.
Running from: C:\Users\User\Desktop\jenkins war\jenkins.war
webroot: $user.home/.jenkins
2022-01-22 17:22:57.563+0000 [id=1] INFO org.eclipse.jetty.util.log.Log#initialized: Logging initialized @491ms t
o org.eclipse.jetty.util.log.JavaUtilLog
2022-01-22 17:22:57.641+0000 [id=1] INFO winstone.Logger#logInternal: Beginning extraction from war file
2022-01-22 17:22:57.689+0000 [id=1] WARNING o.e.j.s.handler.ContextHandler#setContextPath: Empty contextPath
2022-01-22 17:22:57.736+0000 [id=1] INFO org.eclipse.jetty.server.Server#doStart: jetty-9.4.43.v20210629; built:
2021-06-30T11:07:22.254Z; git: 526006ecfa3af7f1a27ef3a288e2bef7ea9dd7e8; jvm 1.8.0_312-b07
2022-01-22 17:22:58.491+0000 [id=1] INFO o.e.j.w.StandardDescriptorProcessor#visitServlet: NO JSP Support for /,
did not find org.eclipse.jetty.jsp.JettyJspServlet
2022-01-22 17:22:58.535+0000 [id=1] INFO o.e.j.s.s.DefaultSessionIdManager#doStart: DefaultSessionIdManager worke
rName=node0
2022-01-22 17:22:58.535+0000 [id=1] INFO o.e.j.s.s.DefaultSessionIdManager#doStart: No SessionScavenger set, usin
g defaults
2022-01-22 17:22:58.536+0000 [id=1] INFO o.e.j.server.session.HouseKeeper#startScavenging: node0 Scavenging every
660000ms
2022-01-22 17:22:59.046+0000 [id=1] INFO hudson.WebAppMain#contextInitialized: Jenkins home directory: C:\Users\U
ser\jenkins found at: $user.home/.jenkins
2022-01-22 17:22:59.162+0000 [id=1] INFO o.e.j.s.handler.ContextHandler#doStart: Started w.@51a9ad5e{Jenkins v2.3
19.2,/,file:///C:/Users/User/.jenkins/war/,AVAILABLE}{C:\Users\User\jenkins\war}
2022-01-22 17:22:59.202+0000 [id=1] INFO o.e.j.server.AbstractConnector#doStart: Started ServerConnector@3551a94{
HTTP/1.1, (http/1.1)}{0.0.0.0:8080}
2022-01-22 17:22:59.203+0000 [id=1] INFO org.eclipse.jetty.server.Server#doStart: Started @2117ms
2022-01-22 17:22:59.203+0000 [id=24] INFO winstone.Logger#logInternal: Winstone Servlet Engine running: controlPor
t-disabled
2022-01-22 17:23:00.282+0000 [id=30] INFO jenkins.InitReactorRunner$1#onAttained: Started initialization
2022-01-22 17:23:00.922+0000 [id=49] INFO jenkins.InitReactorRunner$1#onAttained: Listed all plugins
2022-01-22 17:23:05.490+0000 [id=32] INFO jenkins.InitReactorRunner$1#onAttained: Prepared all plugins
2022-01-22 17:23:05.490+0000 [id=30] INFO jenkins.InitReactorRunner$1#onAttained: Started all plugins
```

Jenkins i internetten indirdiğimiz rar dosyasını cmd de açıyoruz.



Jenkins'e hoş geldiniz!

admin

\*\*\*\*\*

Giriş yap

☐ Beni hatırla





Cmd'de verilen şifremizi kullanarak panele giriş yapıyoruz.

Kontrol Merkezi > Hepsi >

Enter an item name

final

\* Required field

-  **Serbest-stil yazılım projesi yapılırdır**  
Jenkins'in merkezi özelliği, projelerinizi yapılandırmanıza yardım etmesidir. Bu proje türünü kullanarak, herhangi bir yapılandırma sisteminin herhangi bir Kaynak Kodu Yönetimi aracı ile birleştirilebilirsiniz ve hatta yazılım yapılandırmanın dışında başka tür projeler için dahi kullanabilirsiniz.
-  **Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
-  **Çoklu-konfigürasyona sahip proje yapılandır**  
Platforma Özgü yapılandırmalara sahip veya çeşitli ortamlarda test işlemi yapan projeler gibi çok fazla sayıda konfigürasyona ihtiyaç duyan projeler için uygundur.
-  **Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

**Multibranch Pipeline**  
OK Pipeline projects according to detected branches in one SCM repository.

Panele girdikten sonra projemizi oluşturuyoruz.

**Kaynak Kodu Yönetimi**

☐ Hiöbirisi  
☒ Git

**Repositories**

**Repository URL**

**Credentials**

Kodumuzu githubdan çekeceğimiz için github repository url mizi ekliyoruz.

**En üst seviye Maven hedeflerini çalıştır**

**Maven Versiyonu**

**Hedefler**

**POM**

**Properties**

Kodumuzun çalışması için maven ayarlarımızı da giriyoruz.

**Kontrol Merkezi** » final1 »

**Proje final1**

**Çalışma Alanı**

**Son Değişiklikler**

**Permalinkler**

- Son yapılandırma (#5), 16 saat önce
- Son stabil yapılandırma (#5), 16 saat önce
- Son başarılı yapılandırma (#5), 16 saat önce
- Last completed build (#5), 16 saat önce

Ayarlarımızı yaptıktan sonra jenkins ile yapılandırmayı başlatıyoruz.

✓ #6	23.Oca.2022 15:54	
✓ #5	22.Oca.2022 23:28	▼
✓ #4	22.Oca.2022 23:24	
✓ #3	22.Oca.2022 23:16	
✓ #2	22.Oca.2022 23:11	
✓ #1	22.Oca.2022 23:09	

Yapılanmasını bekliyoruz.

Proje geri dön

Durum

Değişiklikler

Console Output

Sürüm Bilgisini Düzenle

Delete build '#6'

Git Build Data

Önceki Yapılandırma

✓ Yapılandırma #6 (23.Oca.2022 15:54:58)

Bu yapılandırmayı sonsuza kadar tut

28 saniye önce başladı  
Sürdü 11 saniye

No changes.

admin kullanıcısı tarafından başlatıldı

Revision: afad37072dc715ba8d9796cbb20b70c489db6d6

Repository: <https://github.com/ilkhantertik/ispartasuleymanDemirelUniversityApiApplicationTests>

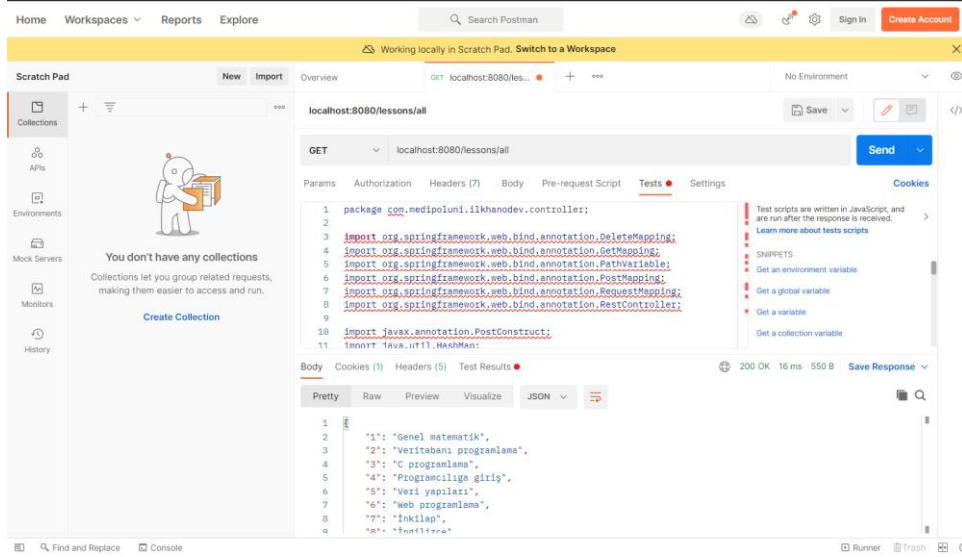
• refs/remotes/origin/main

Kodumuzu yapılandırdıktan sonra Console Output a tıklayarak kodumuzun çalışır halde olup olmadığına bakıyoruz.

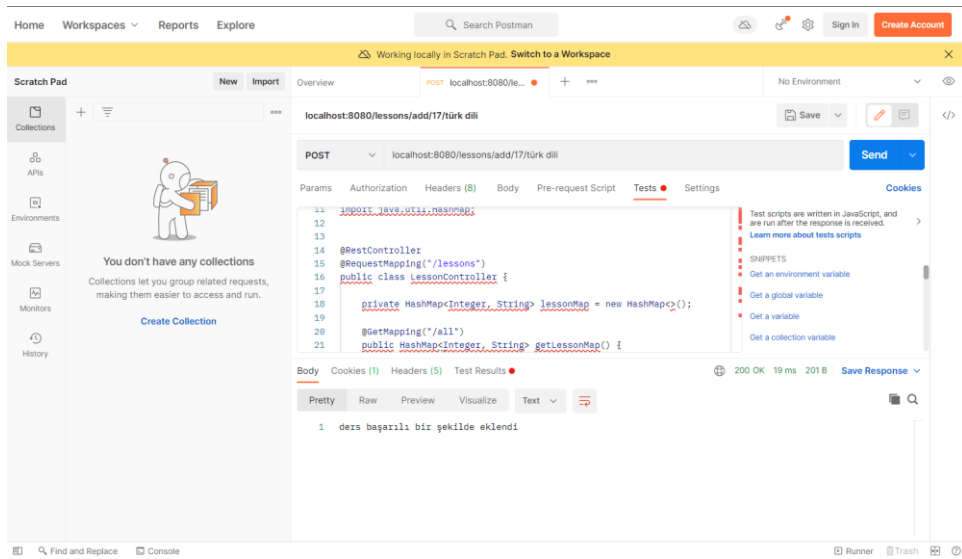
```
2022-01-23 15:55:07.549 INFO 6080 --- [main] ymanDemirelUniversityApiApplicationTests : Started
IspartaSuleymanDemirelUniversityApiApplicationTests in 1.975 seconds (JVM running for 3.052)
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 2.9 s - in
com.isparta.suleyman.demireluniversitesiapi.IspartaSuleymanDemirelUniversityApiApplicationTests
[INFO] Results:
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] --- maven-jar-plugin:3.2.2:jar (default-jar) @ finalv ---
[INFO] Building jar: C:\Users\User\.jenkins\workspace\finalv\target\finalv-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- spring-boot-maven-plugin:2.6.3:repackage (repackage) @ finalv ---
[INFO] Replacing main artifact with repackaged archive
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ finalv ---
[INFO] Installing C:\Users\User\.jenkins\workspace\finalv\target\finalv-0.0.1-SNAPSHOT.jar to
C:\Users\User\.m2\repository\com\example\finalv\0.0.1-SNAPSHOT\finalv-0.0.1-SNAPSHOT.jar
[INFO] Installing C:\Users\User\.jenkins\workspace\finalv\pom.xml to C:\Users\User\.m2\repository\com\example\finalv\0.0.1-SNAPSHOT\finalv-
0.0.1-SNAPSHOT.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 8.374 s
[INFO] Finished at: 2022-01-23T15:55:09+03:00
[INFO] -----
Finished: SUCCESS
```

Yapılandırmamızın olumlu olduğunu görüyoruz. Başarıyla sonuçlanıyor.

# Postman



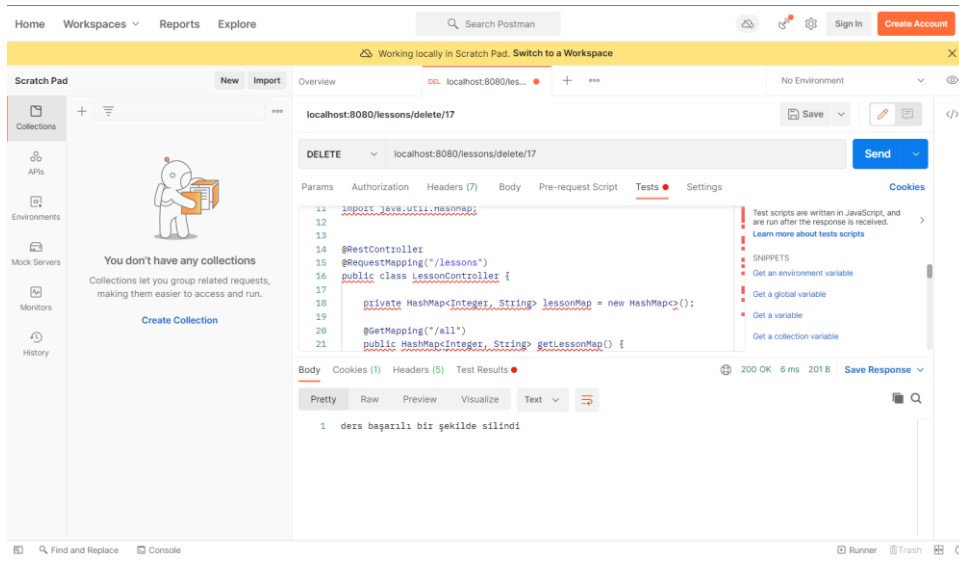
Postman den projemize istek atıp derslerin geldiğini gördüm.



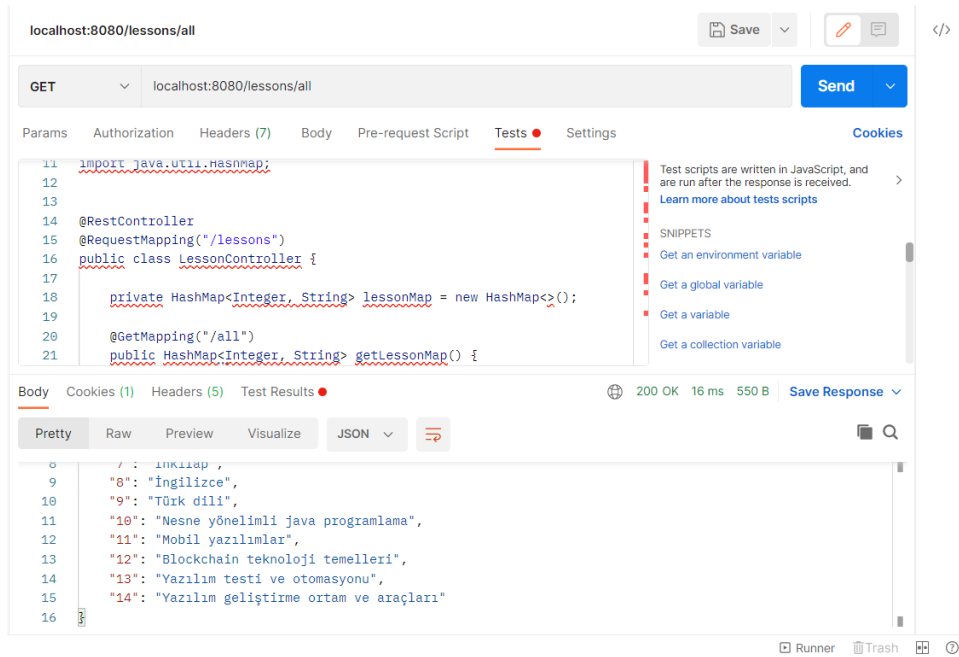
Postman de listelediğimiz derslere yeni bir ders ekliyoruz.



Eklediğimiz dersin çıktısını almış olup istediğimiz sonuca ulaşmış oluyoruz.

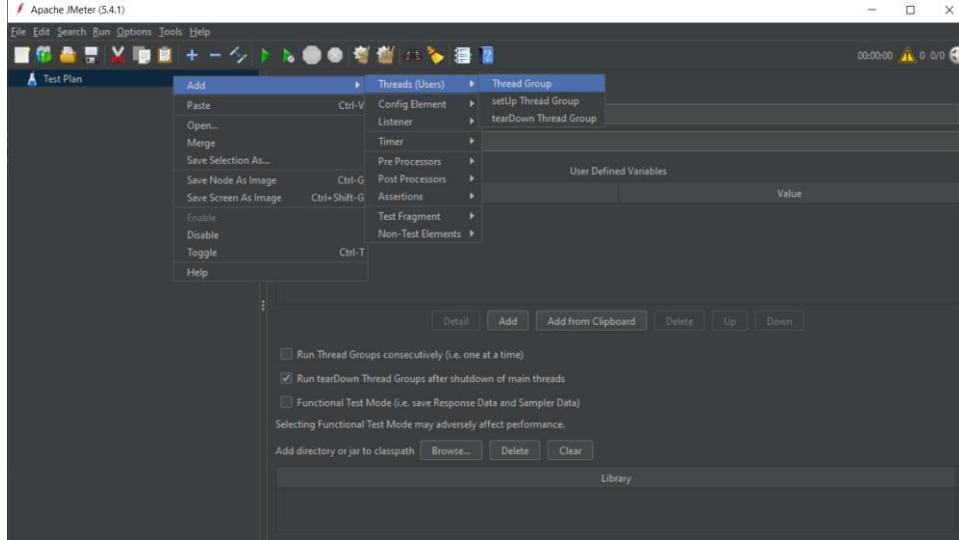


Daha sonra eklediğimiz dersi postman de siliyoruz.

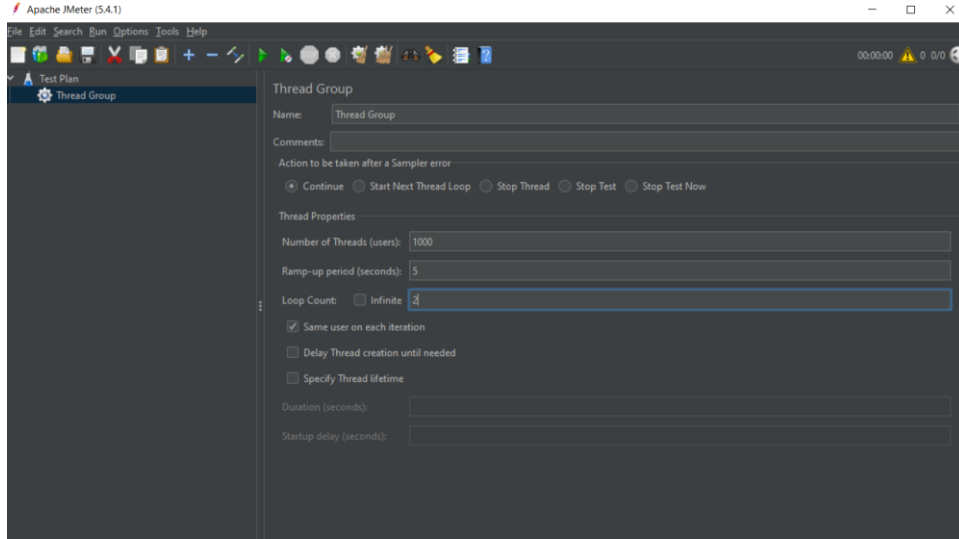


Silmiř olduėumuz dersin silindiėinden emin oluyoruz.

## Jmeter

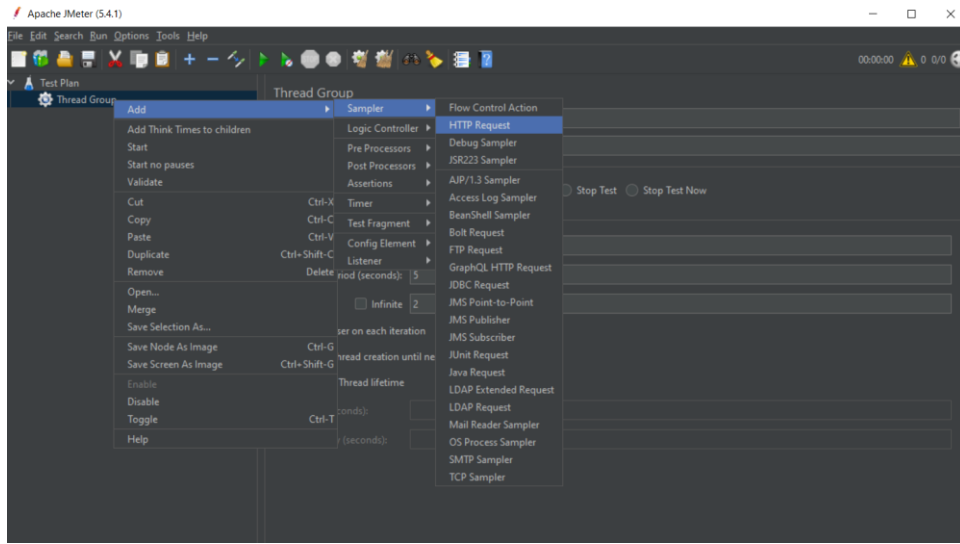


Jmeter’ın projemizi test etmesi iin thread grubunun seilmesi.

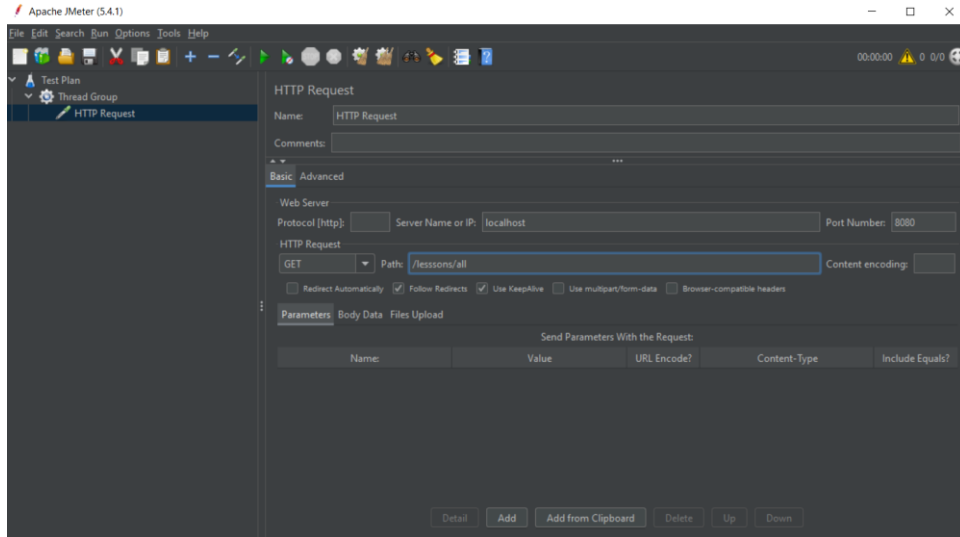


Thread Grubunun parametrelerinin girilmesi. Kullanıcı sayısının girilmesi, saniyenin girilmesi ve dngnn girilmesi.

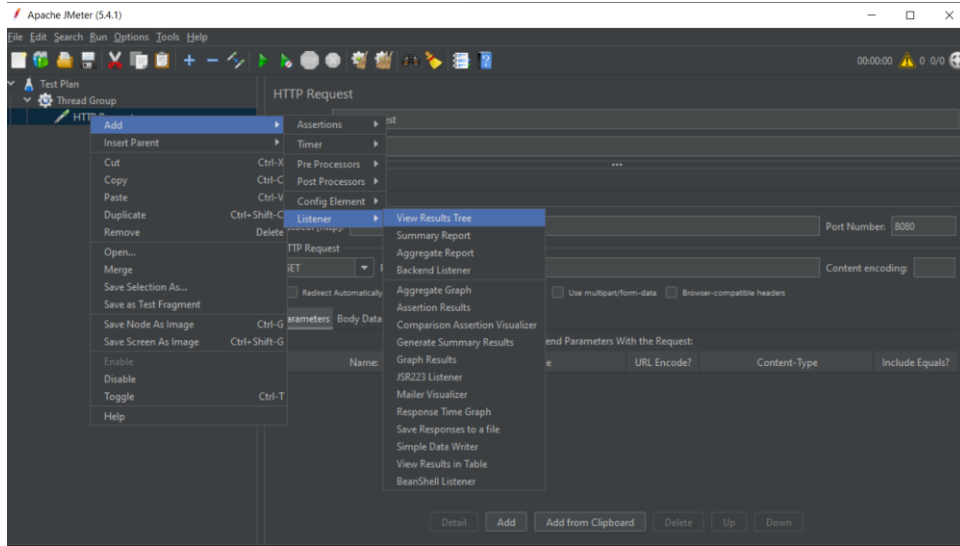




Jmeter ile apimize gidecek olan isteğin türünün seçilmesi.



Jmeterın istek atacağı endointin url ve port numarisının girilmesi.

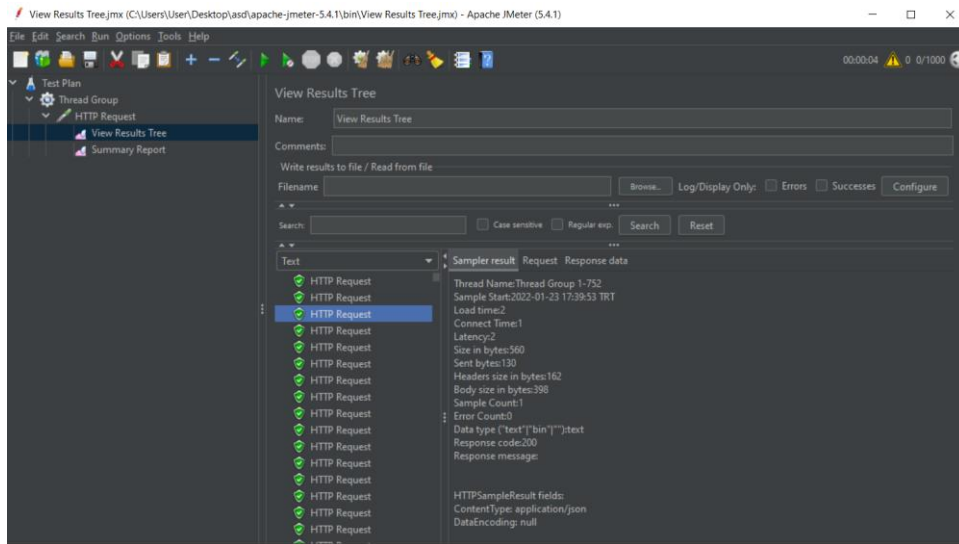


Testin sonucunu görebilmemiz için şablon seçimi yapılması.

The screenshot shows the 'View Results Tree.jmx' file open in Apache JMeter (5.4.1). The 'Summary Report' listener is selected in the 'Test Plan' tree. The main panel displays the 'Summary Report' configuration and a table of test results.

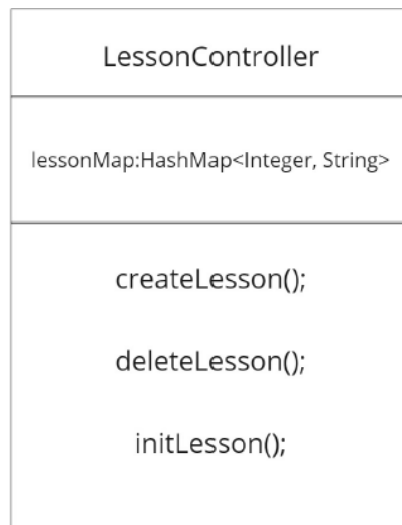
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
HTTP Req...	2000	0	0	5	0.77	0.00%	402.1/sec	219.89	51.05	560.0
TOTAL	2000	0	0	5	0.77	0.00%	402.1/sec	219.89	51.05	560.0

4 saniyede 2000 istek ile endpoint test edildi ve hiçbir hata ile karşılaşılmadı.



Testimizin çıktısı. Hiçbir sorun ile karşılaşılmadı.

## UML Diagram



Projemizin UML diagram çıktısı.

