<https://github.com/koushikkothagal/spring-boot-microservices-workshop>

Machine generated alternative text:
Agenda 
• Creating a few microservices 
• Having them communicate with each other 

Machine generated alternative text:
MicRo ICES 
ano 

**Service Discovery : is pattern and Eureka is server which allow pattern**

Machine generated alternative text:
Managing this complexity 
• Patterns 
• Technologies 

Machine generated alternative text:
In this course 
We'll start from scratch 
Pick one "set" Of technologies - Spring Cloud 
Use only what's required 

Machine generated alternative text:


One ms--call data and second ms- call data and third ms- combine use these data

Api we need to build :-'

Machine generated alternative text:
c."o.l.s . / qt // 

Machine generated alternative text:
Movce— 
S e.ov 
Use-r2D 
Se-O.Jt2.— 
Input: VS.-CID 

Machine generated alternative text:
• Create 3 Spring Boot projects 
Build movie catalog service 
API 
• Build movie info service API 
• Build ratings data service API 
• Have movie catalog service 
call the other two services (th 
naive way) 
Implement a better way 
(Service discovery) 

Machine generated alternative text:
Configure the three microservices to run On a different port each - 8081, 8082 and 8083 
Starting state commit: 03. Added Movie Info API 
Completion state commit: 04. Set application properties 
Github Repo: https:/&ithubcornAoushikkothagaUspring-boot-microservices-workshop 

Machine generated alternative text:
What we have now 

**Requirement :**  need to call movie info service into movie catalog service

Machine generated alternative text:
How to make a REST call from your 
code? 
• Calling REST APIs programatically 
• Using a REST client library 
• Spring Boot comes with a client already in your 
classpath- RestTemplate 

**Three service**

Rating Data service

Movie Info service

Movie Catalog Service

**Step 1 :-** Get all rated Movie id

**Step 2:-** for each movie id , call movie info service and get details

**Step 3 :-** put them all together

Note : we are doing hardcoded rating in **movie catalog service**

**Note:**

**What is RestTemplate?**

RestTemplate. is the central class within the Spring framework for executing synchronous HTTP requests on the client side. Like Spring JdbcTemplate, RestTemplate. is also a high-level API, which in turn is based on an HTTP client

**Why RestTemplate is deprecated?**

RestTemplate provides a synchronous way of consuming Rest services, which means it will block the thread until it receives a response. RestTemplate is **deprecated since Spring 5** which means it's not really that future proo

**Imp :- restTemplate.GetForObject ----- take 2 arg;**

1 arg :- it will be url

2 arg :- it will be string which need to be unmarshal into object

**Note : while unmarshalling java will create a instance**

Machine generated alternative text:
L6 @RestContr011er 
18 
23 
24 
28 
29 
31 
32 
33 
34 
35 
36 
38 
39 
public class CatalogResource 
public List'Cata10gIten1) getCata10g(@PathVariab1e("userId") String userld) 
- Get all rated Movie id 
Step 1 
Step 2:- for each movie id 
, call movie info service and get details 
put them all together 
Step 3 
new Rest Template(); 
Rest Template rest Template 
List'Rating) ratings Arrays.asList(new Rating("1234", 3), new Rating("5678", 4)); 
return ratings. stream() . map(rating 
Movie 
movie 
rest Template . getForObject("http://localhost:8882/ • 
movies 
+ rating. getMovieId(), 
return new Catalogltem(movie . getName(), "Description" , 
rating. getRating( ) ) ; 
. collect (Collectors. toList() ) ; 
Movie. class); 

**Note: how to create a single object in spring** : create a bean which is singleton

Machine generated alternative text:
MovieCatalogServiceAppl 
avabrains.io 
Spr 
Ja'K 
args); 
13 
package javabrains.moviecatalogservice; 
öSpr ication 
ic class MovieCataIogServiceAppIication 
3Bean 
public Rest Template 
args) 
SpringAppI icat ion. run(uov ieCataIogServiceAppI ica t ion. ass , 
2019-01-26 
2019-01-26 
INFO 
INFO 
13537 - 

Above method will execute only one. Any class will autowired RestTemplate will get same instance from here.

@Autowired is a consumer :- give me something

@Bean :- producer tell spring I have something that I have to give.

**Note : two bean of same type then we have to use qualifier**

**Example : -** if we are using two RestTemplate then use Qualifier if, we will use two bean of same name then will get error

**Note :-**

**RestTemplate is deprecated since it is synchronous**

Machine generated alternative text:
movie-catalog-service (bootl (spring-boot-micrc 
Spring Elements 
src/main/java 
io.javabrains.moviecatalogservice 
MovieCatalogServiceApplication.java 
MovieCatalogServiceApplication 
io.javabrains.moviecatalogservice.models 
Catalogltem.java 
Movie.java 
v Rating.java 
e Rating 
io.javabrains.moviecatalogservice.resource 
CatalogResource.java 
CatalogResource 
src/main/resources 
application.properties 
src/test/java 
JRE Svstem Librarv 
IJavaSE-1.81 
Boot Dashboard X 
Type tags, projects, or working set names to match (incl. * 
local 
package io.javabrains . moviecatalogservice; 
1 
2 
39 import org. springframework . boot . SpringApp1ication; 
4 import org. springframework. boot . autoconfigure . SpringBootApp1ication; 
5 import org. springframework. context . annotation . Bean; 
6 import org. springframework.web . client . Rest Template; 
7 
8 @SpringBootApp1ication 
public class MovieCata10gServiceApp1ication 
9 
args); 
le 
11 
129 
13 
14 
15 
169 
17 
18 
19 
2e 
21 
22 
23 
public static void main(StringCl args) 
SpringApp1ication . run (MovieCata10gServiceApp1ication . class , 
@Bean 
public Rest Template getRestTemp1ate() 
return new Rest Template(); 

Machine generated alternative text:
Catalog Resource.java 21 Catalog Item.java 
MovieCatalogSewiceApplication.java 
org. springframework . beans . factory. annotation . Autowired; 
org. springframework . web . bind . annotation . PathVariab1e; 
org. springframework . web . bind . annotation . RequestMapping; 
org. springframework . web . bind . annotation . RestContr011er ; 
org. springframework . web . client . Rest Template; 
application.properties 
Rating.java 
Movie.java 
6 
7 
8 
9 
11 
12 
13 
14 
15 
16 
17 
18 
19 
2e 
2 Is 
22 
23 
249 
25 
26 
27 
29 
3e 
31 
32 
33 
34 
35 
36 
37 
38 
39 
41 
import 
import 
import 
import 
import 
import 
import 
import 
Movie . class) ; 
io. javabrains . moviecatalogservice . models . Catalogltem; 
io. javabrains . moviecatalogservice . models . Movie; 
io. javabrains . moviecatalogservice . models . Rating; 
@RestContr011er 
@RequestMapping("/catalog" ) 
public class CatalogResource 
@Autowired 
private Rest Template rest Template; 
@RequestMapping("/fuserIdY' ) 
public List'Cata10gIten1) getCata10g(@PathVariab1e("userId") String userld) 
- Get all rated Movie id 
Step 1 
Step 2:- for each movie id 
, call movie info service and get details 
put them all together 
Step 3 
List'Rating) ratings Arrays.asList(new Rating("1234", 3), new Rating("5678", 4)); 
return ratings. stream() . map(rating 
Movie movie 
rest Template . getForObject("http://10ca1host:8B82/movies/" + rating. getMovieId(), 
return new Catalogltem(movie . getName(), "Description" , 
rating. getRating() ) ; 
. collect (Collectors. toList() ) ; 

RestTemplate uses Java Servlet API and is therefore **synchronous and blocking**. Contrarily, WebClient is asynchronous and will not block the executing thread while waiting for the response to come back. Only when the response is ready will the notification be produced. RestTemplate will still be used.

**Should I use WebClient?**

Spring 5 documentation suggests that WebClient is now the **preferred way to make HTTP requests**. WebClient is part of Spring WebFlux and is intended to replace the classic RestTemplate. Compared to RestTemplate , WebClient has a more functional feel and is fully reactive.

**Microservice call using web client**

.bodyToMono :- is eventually a type of promise that u will get it

.block ---- we are blocking execution till mono is return ( that is movie object)

Machine generated alternative text:
rains.10 
args) ; 
package io. javabrains.moviecatalogservice; 
import 
OSpringBootApp1 ication 
public class MovieCatalogServiceApplication 
public Rest Template getRestvempVave() 
return new RestT+p1ate(); 
i)Bean 
public WebClient . Builder 
return Webclient.buitder(); 
14 
public 
static void main(stringt) args) SpringApplication.run(MovieCata10gServiceApp1ication.class, 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
4163 - 
4163 - 
4163 - 
4163 - 
4163 
4163 - 
4163 
main) 
mainl 
main) 
mainl 
main) 
mainl 
tnio-8081 -exec-Il 
o. a. catalina.core .AprLifecycleListener 
o. a. c.c.c. (Tomcat) 
o. s.web. context. ContextLoader 
0. s. s. concurrent . ThreadP001 TaskExecutor 
o. s. b .w. embedded. tomcat. TomcatWebServer 
i j.m.MovieCatalogServiceApplication 
o. s. servlet. Dispatcherservlet 
o. s .web.servlet.Dispatcherse 
i 
2019-ø2-ø9 14:19:01.347 
-01.426 
2019-ø2-09 14:19. 
2019-ø2-09 14:19. 
2019-ø2-09 
2019-ø2-09 .84" 
2019-ø2-09 
2019-132-09 14:19:29. 155 
The APR based Apache Tomcat Natii. 
Initializing Spring embedded Web' 
Root WebApp1icationContex 
Initializing ExecutorSer•.• 
Tomcat started on 
Started MovieCatalogServiceApplie 
Initializing Servlet 'dispatchers 
t ion n 

Machine generated alternative text:
Mutowired 
private WebClient.Buitder webC1ientBui1der; 
public ListeCata10gItem) "userld") String userld) ( 
ListeRating» ratings : Arrays. asList( 
new Rating( movield: "1234" , 
rating: 4), 
new Rating( movield: "5678", rating: 3) 
return ratings. stream().map(rating 
// Movie movie rest Template.getForObject( "http://localhost:8ø82/movies/" + ra 
Movie movie - webCtientBui1der. build() WebClient 
. get( ) of 
.uri( s; -http://10cathost:8Ø82/movies/" + rating.getMovieId()) captureof' 
retrieve() ResponseSpec 
. bodyToMono(Movie. class) MonoeMovie» 
4163 
"Desc" 
, rating.getRating()); 
. block( ) 
return new Catatogltem(movie .ÉetName(), desc; 
main) 
main) 
main) 
main) 
main) 
INFO 
INFO 
INFO 
INFO 
INFO 
4163 
4163 - 
4163 - 
4163 
o. a. catatina. core.AprLifecycleListener 
o.a.c.c.C. (Tomcat). (localhost).t/) 
o. s.web. context . Context Loader 
o. s. s . concurrent. ThreadPootTaskExecutor 
i. j lication 
Initializi 
Started MO 

Machine generated alternative text:
@RestContr011er 
public class CatalogResource 
@Autowired 
private Rest Template rest Tem late; 
@Autowired 
WebC1ient . Builder webC1ientBui1der; 
public List'Cata10gIten1) getCata10g(@PathVariab1e("userId") String userld) 
- Get all rated Movie id 
Step 1 
Step 2:- for each movie id 
, call movie info service and get details 
put them all together 
Step 3 
List'Rating) ratings Arrays.asList(new Rating("1234", 3), new , 
return ratings. stream() . map(rating 
// Movie movie - 
rest Template . getForObject ( "http : / / localhost : 8B82/movies/" 
// rating. getMovieId(), Movie. class); 
// Alternative WebC1ient way 
Movie 
movie webC1ientBui1der . build() 
. get() 
. uri("http://10ca1host:8B82/movies/" + rating. getMovieId()) 
4)); 
. retrieve( ) . bodyToMono(Movie . class) 
l. block(); 
return new Catalogltem(movie . getName(), "Description" , 
. collect (Collectors . toList() ) ; 
rating. getRating() ) ; 

Machine generated alternative text:
4 movie-catalog-service/pom. xml 
@@ -23,6 +23, le @@ 
23 
24 
25 
26 
27 
28 
23 
24 
25 
27 
31 
32 
(groupld»org.springframework.bootegrouplcb 
'/dependencp 
(dependencp 
(groupld»org.springframework.bootegrouplcb 
( artifactld»spri ng-boot-starter -webfl / artifactlcb 
'/dependencp 
(dependencp 
(groupld»org.springframework.bootegrouplcb 

**Calling rating api**

Machine generated alternative text:
21 Catalogltem.java 
application.properties 
Rating.java 
Movie.java 
movie-catalog-service/pom.xml 
Quick Acces 
UserRating.java 
MovieCatalogSewiceApplication.java 
Zl Catalog Resource.java 
18 
19 @RestContr011er 
2B 
21 
22 
24 
25 
269 
27 
28 
299 
3e 
31 
35 
36 
37 
38 
39 
4e 
1 
2 
3 
47 
8 
9 
51 
52 
53 
public class CatalogResource 
@Autowired 
private Rest Template rest Template; 
@Autowired 
WebC1ient . Builder webC1ientBui1der; 
public List'Cata10gIten1) getCata10g(@PathVariab1e("userId") String userld) 
- Get all rated Movie id 
Step 1 
Step 2:- for each movie id 
, call movie info service and get details 
put them all together 
Step 3 
UserRating userRating rest Template. getForObject("http://10ca1host:8B83/ratingsdata/users/" 
return userRating. getUserRatings() . stream() . map(rating - ) 
Movie movie 
rest Template . getForObject( "http://10ca1host:8B82/movies/" + 
rating. getMovieId(), Movie. class); 
// Alternative WebC1ient way 
Movie movie webC1ientBui1der . build() 
. get() 
.uri("http://10ca1host:8B82/movies/" + rating. getMovieId()) 
UserRating. class); 
+ userld, 
. retrieve( ) . bodyToMono(Movie . class) 
. block(); 
return new Catalogltem(movie. _ 
( ) , "Description" , 
getName 
. collect (Collectors. toList() ) ; 
rating. getRating()) ; 

Machine generated alternative text:
MovielnfoSen.'iceApplication.java 
Movie.java 
MovieResource.java application.properties 
package io.javabrains. movieinfoservice. resources; 
1 
2 
39 import org. springframework . web . bind . annotation . PathVariab1e; 
4 import org. springframework.web . bind . annotation . RequestMapping; 
5 import org. springframework.web . bind. annotation . RestContr011er; 
6 
7 import io.javabrains. movieinfoservice . models . Movie; 
8 
9 @RestContr011er 
1B 
11 public class MovieResource 
12 
19 
public Movie getMovieInfo(@PathVariab1e("movieId") String movield) 
return new Movie(movield, "Test Name"); 
14 
15 
16 
17 
18 

Machine generated alternative text:
RatingsDataSewiceApplication.java 
RatingsResource.java 21 Rating.java 
application.properties 
1 package io.javabrains. ratingsdataservice. resources; 
UserRating.java 
2 
39 import 
java . util . Arrays; 
java . util. List; 
org. springframework . web . bind . annotation . PathVariab1e; 
org. springframework . web . bind . annotation . RequestMapping; 
org. springframework . web . bind . annotation . RestContr011er ; 
10. javabrains. ratingsdataservice . model . Rating; 
io. javabrains . ratingsdataservice . model . UserRating; 
4 
5 
6 
7 
8 
9 
11 
12 
13 
14 
15 
16 
17 
18 
19 
2e 
21 
229 
23 
24 
25 
26 
27 
28 
29 
3e 
31 
32 
import 
import 
import 
import 
import 
import 
4)); 
@RestContr011er 
public class RatingsResource 
public Rating getRating(@PathVariab1e("movieId") String movield) 
return new Rating(movield, 4); 
public UserRating getUserRatings(@PathVariab1e("userId") String userld) 
List'Rating) ratings Arrays.asList(new 3), new Rating("5678", 
UserRating userRating new UserRating(); 
userRating. setUserRatings ( ratings) ; 
return userRating; 

RestTemplate : to external call mayor may not be rest service.

**Understanding service Discovery**

**What we are doing here :** hard coding url

Machine generated alternative text:
Why hard coded URLs are bad? 
Changes require code updates 
Dynamic URLs in the cloud 
Load balancing 
Multiple environments 

Multiple environment :- localhost , in qa----qa host

**How to do service discovery**

1. **Client side service discovery**

Machine generated alternative text:
Can I get 
Service 2 please? 
CLIENT 
Sure. here is Where 
SOS 
D tscovny 
SERVER 
SERV 

1. **Server side service discovery**

Machine generated alternative text:
Can you tell 
Service 2 1 said 
SERVER 
Hey Service 2! That 
Client guy Says hi' 

Machine generated alternative text:
Spring Cloud uses 
client side discovery 

**Pattern : client side service discovery**

**Technology : Eureka that spring cloud uses to talk for microservice**

**Netflix oss has created bunch of libraries**

Machine generated alternative text:
Eureka 
Ribbon 
Hysterix 
zuul 

Machine generated alternative text:
Spring Boot / 
Spring Cloud 
abstraction layer 

**======================================**

**Demo :-**

**Eureka client :**Individual microservice

**which is registered in eureka server**

**Eureka client talk to eureka server, hey I need this service**

Machine generated alternative text:
Steps to making this work 
• Start up a Eureka server 
• Have microservices register (publish) using 
Eureka client 
Have microservices locate (consume) using 
Eureka client 

**Starting a Eureka server**

**Create a Eureka server means :** create spring boot application

Machine generated alternative text:
SPRING INITIALIZR 
bootstrap your application now 
Generate a 
with 
Mave' project 
• and Spring Boot 
Dependencies 
Add Spring Starters to application 
Search for 
Project Metadata 
Artifact coordinates 
'o. javabrain• 
'cove ry — server 
Don't know what to 100k for? more 
Generate Project + 
Switch to o 

Machine generated alternative text:
package io. javabrains.discoveryserver; 
- import 
*pringgootAppIication 
aEnabI eE Server 
public class 
public static void args) ( SpringApptication. run(DiscoveryServerAppIication.ctass, 
rains. io 
args); 

**@EnableEurekaServer** : tells spring boot application that this is eureka server

JAXB **allows Java developers to access and process XML data without having to know XML**

**JAXB has been completely removed from Java SE 11**

Machine generated alternative text:
C O localhost:8761 
spring 
Eureka 
System Status 
Environment 
Data center 
DS Replicas 
test 
default 
Current time 
Uptime 
Lease expiration enabled 
Renews threshold 
Renews (last min) 
Availability Zones 
HOME 
Paused 
LAST 1000 SINCE STARTUP 
+0530 
00:00 
false 
1 
Instances currently registered with Eureka 
Application 
No instances available 
General Info 
AMIs 

**NOTE:** spring Eureka is ui which default run in port 8761

**No instance available :**

why we have microservice but they don’t have any eureka client , we need to register eureka client . Those microservice have the functionality to tell eureka server that they exist.

Machine generated alternative text:
server. 
port:8ffål 
abrains.io 
eureka. client . fetch-registrvzfalse 

**Need to add this in properties file.**

**Like multiple microservice run in same way multiple eureka server run**

**When application run (eureka server) not only it provide registry it also try to register other eureka server, so if one eureka server fails others are there to registry**

**Demo**

Machine generated alternative text:
Viewer Exercise 
Add the Eureka client dependency to have the 3 microservices register with Eureka 
server 
Starting state commit: 09. Added Eureka server 
SUBSCRIBE 
u 
mp euon state commit: 
services to 

**Spring cloud is a technology for building microservice**

NOTE : in microservice pom.xml we need to add

1.) eureka client

2.) spring cloud version

Machine generated alternative text:
MovielnfoSewiceApplication.java 21 Movie.java 
MovieResource.java 
application.properties 
movie-info-service/pom.xml 
17S 
229 
279 
31 S 
verslon» 
(relativePathD ( ! 
lookup parent repository - 
e/ parenb 
(artifactld)movie- catalog- service(/arti+actld» 
i e- catalog- service(/narne» 
(narne)mov 
(description»Demo project for Spring aoot(/description» 
(prope 
ava . version»l .84/ java . version» 
spring- cloud . version»Greenwi ch . REL EASE(/ spri ng- cloud . versio» 
( / opertie» 
(dependencie» 
(dependenc» 
(groupld»org.springframework.boot / 
(artifactld»spri ng- boot- starter- web(/arti+actld» 
( / dependenc» 
(dependenc» 
(groupld»org.springframework.boot / 
(a rti+actld)spri ng- boot- starter- web-FI ux(/arti+actld» 
e,' dependen 
epen ency) 
(artifact I d) spri ng- cloud- starter- net+l ix- eureka- cl artifactld» 
(dependency) 
(groupld»org.springframework.boot / groupld» 
(artifactld»spring- boot- starter-test U artifactld» 
( 'dependenc» 
('dependencie» 
anagernenb 
(dependencies 
(dependenc» 
(artifactld»spring- cloud-dependencies(/arti+actld» 
(versi ng- cloud . version) ( / versi 
( / dependenc» 
('dependencie» 
e/ dependencyManagernenO 
(builcD 
(plugin" 
(plugi» 
(groupld»org.springframework.boot / groupld» 
(artifactld»spri ng- bcnzt-ma ven- pl ugin(/arti+actld» 
e/ plugi» 
4/ plugin" 
(/buiIcD 
pom.xml 
64 
Overview Dependencies Dependency Hierarchy Effective POM 

NOTE:

once we will add client in pom.xml of microservice , it will register in eureka server by

Name unknown

Machine generated alternative text:
C 
O localhost:8761 
System Status 
Environment 
test 
default 
Data center 
Current time 
Uptime 
Lease expiration enabled 
Renews threshold 
Renews (last min) 
Paused 
+0530 
01:14 
false 
3 
EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN 
THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE. 
DS Replicas 
urrently registered with Eureka 
localhost:8082 
UP (1) - 
Application 
UNKNOWN 
Genera 
AMIs 
n/a (1) 
Info 
Availability Zones 
(1) 

Note: **we can provide name by giving name in properties file**

Machine generated alternative text:
Package Explorer 
oSewiceAp... 
J owe Java 
2e22-e3-e3 
:44.7e3 
2e22-e3-e3 
:44.7es 
2822-83-83 
:44.7e8 
2822-83-83 
:44.711 
2822-83-83 
:44.713 
2822-83-83 
:44.714 
2822-83-83 
:44. 717 
2822-83-83 
:44.768 
2e22-e3-e3 
:44.778 
2e22-e3-e3 
:44. 772 
2822-83-83 
:44.788 
MovieResource.java 
application.propert... 
movie-info-service/p... 
Outline 
An outline is not available. 
Spring Explorer 
movie-info-service 
Quick Access 
spring. application. name-movie- info- service 
1 
server . port:8B82 
Console 
*b Progress Problems 
MovielnfoServiceApplication IJava Application) C:XProgram FilesXJavaXJre1.8.0 191XbinXJavaw.exe (Mar 3, 2022, 4:06:32 PM) 
movie-info-service (bootl (spring-boot-microservice 
src/main/java 
io.javabrains.movieinfoservice 
us MovielnfoServiceApplication.java 
io.javabrains.movieinfoservice.models 
Movie.java 
io.javabrains.movieinfoservice.resources 
MovieResource.java 
src/main/resources 
application.properties 
src/test/java 
JRE System Library 
1.81 
Maven Dependencies 
src 
target 
mvnw 
mvnw.cmd 
pom.xml 
Servers (spring-boot-microservices-workshop b710 
@ Boot Dashboard 
Sewers 
Pivotal tc Server Developer Edition v4.0 
(Stopped) 
16. 
16. 
16. 
16. 
16. 
16. 
16. 
16. 
16. 
16. 
16. 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
INFO 
le2€e - 
le2€e - 
le2€e - 
le2€e - 
18268 - 
18268 - 
le2€e - 
le2€e - 
le2€e - 
le2€e - 
le2€e - 
main) 
main) 
main) 
main) 
main) 
main) 
EnfoRep1icator-81 
main) 
main) 
main) 
EnfoRep1icator-81 
com.netflix.discovery.DiscoveryC1ient 
com.netflix . discovery. DiscoveryC1ient 
c . n . discovery. InstanceInfoRep1icator 
com.netflix . discovery. DiscoveryC1ient 
o. s. c. n. e. s. EurekaServiceRegistry 
com.netflix . discovery. DiscoveryC1ient 
com.netflix . discovery. DiscoveryC1ient 
o. s. b . w. embedded . tomcat . TomcatWebSer'•, 
. s. c. n . e. s . EurekaAutoServiceRegistrat 
i . j . m. MovieInfoServiceApp1ication 
com.netflix . discovery. DiscoveryC1ient 

Machine generated alternative text:
Adding input structure fib X 
20 Creating Eureka clients 
Java Brains - YouTube 
C O localhost:8761 
System Status 
Environment 
test 
default 
Data center 
x 
Eu reka 
Current time 
Uptime 
x 
spring-boot-microservice 
+0530 
01:17 
false 
5 
2 
x 
Lease expiration enabled 
Renews threshold 
Renews (last min) 
EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN 
THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE. 
DS Replicas 
Instances currently registered with Eureka 
Status 
UP (1) - 
UP (1) - 
localhost:movie- info-service:8082 
localhost:8082 
Value 
Application 
MOVIE-INFO-SERVICE 
UNKNOWN 
General Info 
Name 
AMIs 
n/a (1) 
n/a (1) 
Availability Zones 
(1) 
(1) 

**NOTE:** the way to register microservice in eureka server is to add eureka client in pom.xml of microservice.no need to add extra line

**Note:**

we can add @EnableEurekaClient upto certain version is mandatory but now it is not required.

**NOTE:**

right now eureka server is running on default port(8761) so eureka client able to register it

**Note:**

**if we are running eureka server on different port then we need to add it in properties file of microservice**

Machine generated alternative text:
C 
O localhost:8761 
System Status 
Environment 
test 
default 
Data center 
Current time 
Uptime 
Lease expiration enabled 
Renews threshold 
Renews (last min) 
+0530 
01:35 
false 
8 
4 
EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN 
THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE. 
DS Replicas 
Instances currently registered with Eureka 
UP 
UP 
UP 
localhost:movie-catalog-service:8081 
localhost:movie-info-service:8082 
localhost:ratings-data-service:8083 
Application 
MOVIE-CATALOG-SERVICE 
MOVIE-INFO-SERVICE 
RATINGS-DATA-SERVICE 
AMIS 
n/a (1) 
n/a (1) 
n/a (1) 
Availability Zones 
(1) 
(1) 
(1) 

**NOTE:**

**If we are changing any code in microservice no need to restart eureka server , microservice will register in eureka server automatically at run time**

**=============registered the services now how to consume it===================**

**The lib knows eureka, it will call eureka and get service back**

**@LoadBalanced :** it does service discovery in load balanced way

**Basically it telling RestTemplate** : loadBalanced giving resttemplate a url which is hint to call actual url.

restTemplate will call eureka instance but not actual call

Machine generated alternative text:
package io.javabrains.moviecatalogservice; 
import . 
aspringBootApplication 
iEnableEurekaCl ient 
public class MovieCatalogServiceApplication 
13 
iBean 
aLoadBalanced 
public RestTemplate 
return new RestTempIate(); 
iBean 
public WebClient. Builder getWebCIientBuilder() 
return WebClient . builder(); 
public static void main(StringLJ args) SpringApplication . run(MovieCata1 
brains. 

Machine generated alternative text:
Zl Catalog Resource.java movie-catalog-service/pom.xml 
application.properties 
MovieCatalogSewiceApplication.java 
18 
19 
21 
22 
24 
25 
269 
27 
28 
299 
3e 
31 
35 
36 
37 
38 
39 
1 
2 
44 
7 
48 
9 
51 
52 
53 
@RestContr011er 
@RequestMapping("/catalog" ) 
public class CatalogResource 
@Autowired 
private Rest Template rest Template; 
@Autowired 
WebC1ient . Builder webC1ientBui1der; 
public List'Cata10gIten1) getCata10g(@PathVariab1e("userId") String userld) 
- Get all rated Movie id 
Step 1 
Step 2:- for each movie id 
, call movie info service and get details 
put them all together 
Step 3 
UserRating userRating rest Template. getForObject("http://ratings-data-service/ratingsdata/users/" 
return userRating. getUserRatings() . stream() . map(rating 
Movie 
movie 
rest Template . getForObject( "http://movie-info-service/movies/" + 
rating. getMovieId(), Movie. class); 
// Alternative WebC1ient way 
Movie movie webC1ientBui1der . build() 
. get() 
. uri( "http://10ca1host:8B82/movies/" + rating. getMovieId()) 
+ userld, 
UserRating. class); 
. retrieve( ) . bodyToMono(Movie . class) 
. block(); 
return new Catalogltem(movie . getName(), "Description" , 
. collect (Collectors. toList() ) ; 
rating. getRating() ) ; 

Machine generated alternative text:
C 
System 
Environment 
Data center 
O localhost:8761 
Status 
test 
default 
Current time 
Uptime 
Lease expiration enabled 
Renews threshold 
Renews (last min) 
+0530 
01:37 
false 
8 
6 
EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN 
THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE. 
DS Replicas 
Instances currently registered with Eureka 
UP 
UP 
UP 
localhost:movie-catalog-service:8081 
localhost:movie-info-service:8082 
localhost:ratings-data-service:8083 
Application 
MOVIE-CATALOG-SERVICE 
MOVIE-INFO-SERVICE 
R INGS-DATA-SERVICE 
AMIs 
n/a (1) 
n/a (1) 
n/a (1) 
Availability Zones 
(1) 
(1) 
(1) 

**LoadBalancing : doing service discovery**

Machine generated alternative text:
on another port. 
ushiks—MacBook—Pro:target javabrainsS java —Dserver.port:82ø1 —jar 

**Note one micorservice is running at port 8081 same microservice we run at different port again. Now 2 microservice instance is running and it will register in eureka server**

Machine generated alternative text:
Renews (last min) 
DS Replicas 
localhost 
Instances currently registered with Eureka 
—- 
10 
Application 
MOVIE-CATALOG-SERVICE 
MOVIE-INFO-SERVICE 
RATINGS-DATA-SERVICE 
General Info 
Name 
total-avail- 
AMIS 
n/a (1) 
n/a (2) 
n/a (1) 
4) 
Availability Zones Status 
(1) 
(2) 
(1) 
4:24 / 
UP (I) - 
up (2) - . 
up (1) - 
Value 
308mb 
SUBSCRIB 

Machine generated alternative text:
(I •nane• name" , , : •rating" | 

**But still it working so client(microservice) is doing load balancing**

**NOTE: LoadBalancing :-** everything will happen automatically at client side, service discovery

Note: service name will go in properties file