Assignment4

Saturday, 25 March, 2023 12:31 PM

Objective:

•Design an application for student management system that includes all the functionalities and concepts learned in this course

•Follow the instructions to create solutions according to the concepts taught in each module

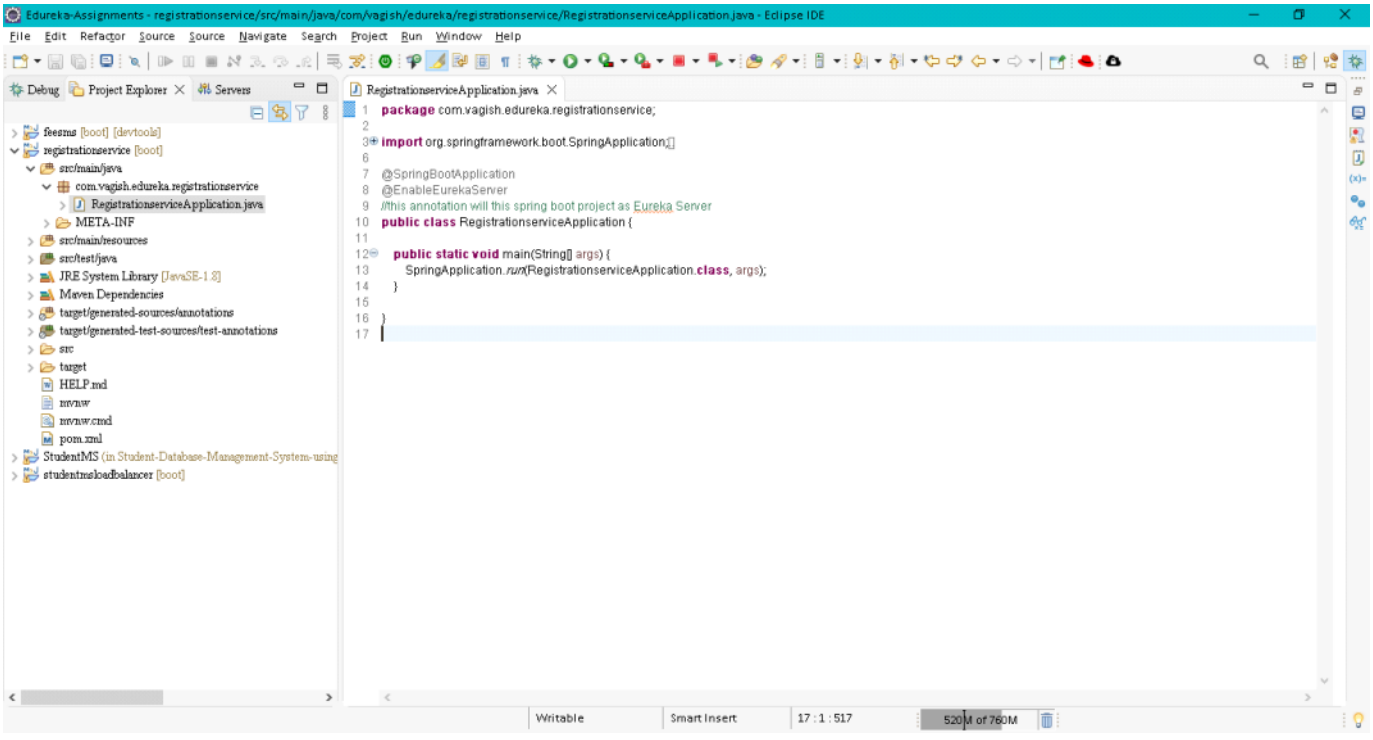
Problem Statement:

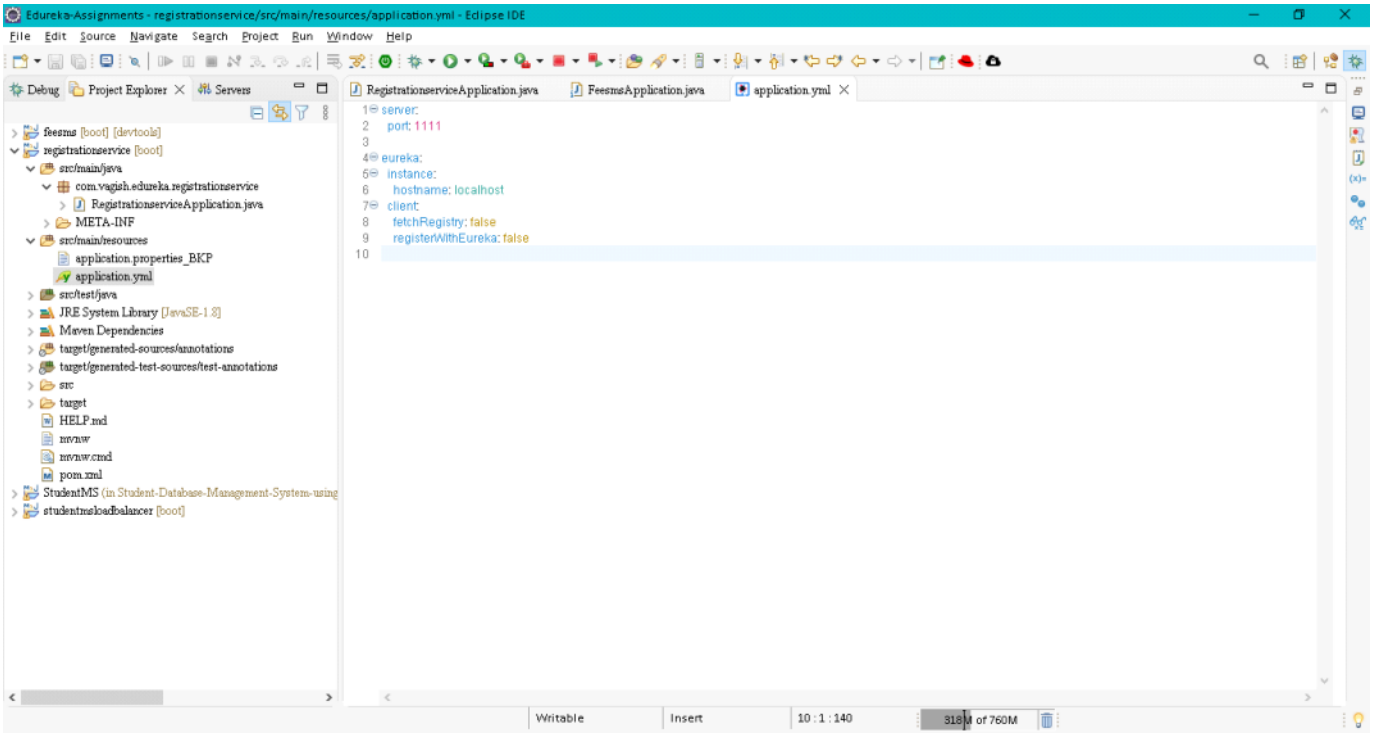
A popular education society "EduExcellence" is setting up a new school in the city. They want a "School Management Software" that should match the high standards of the school that they aspire to build. They have hired you to provide them with the software that should enable them to manage all aspects of running a school successfully, but they would want the application to grow gradually with the growth of the new school.

Implement a microservice(fees ms) to manage the following functions.
Use similar tools as used in student ms.
1.An API to fetch all fees paid by a student
2.An API to pay fees for a student
Also, implement a couple of APIs in student ms to call these new APIs of fee sms.

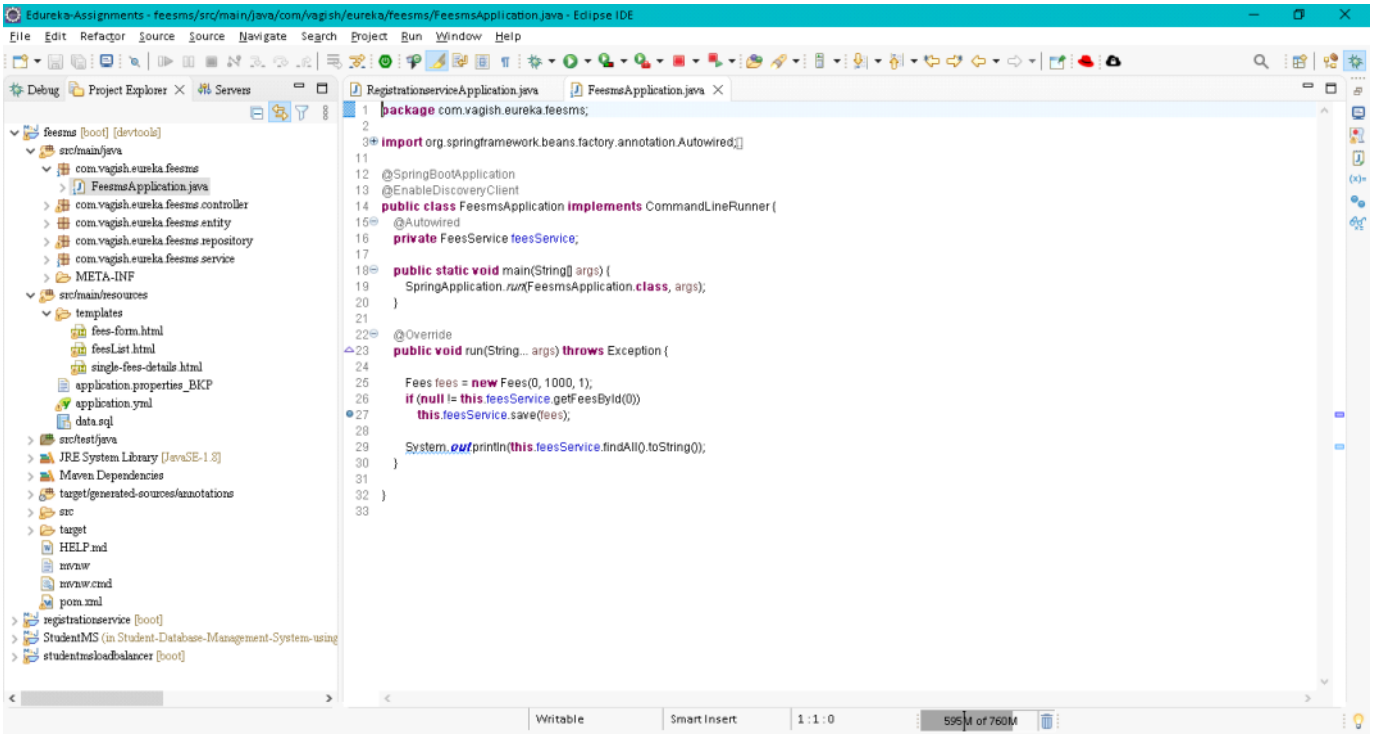
Solution:

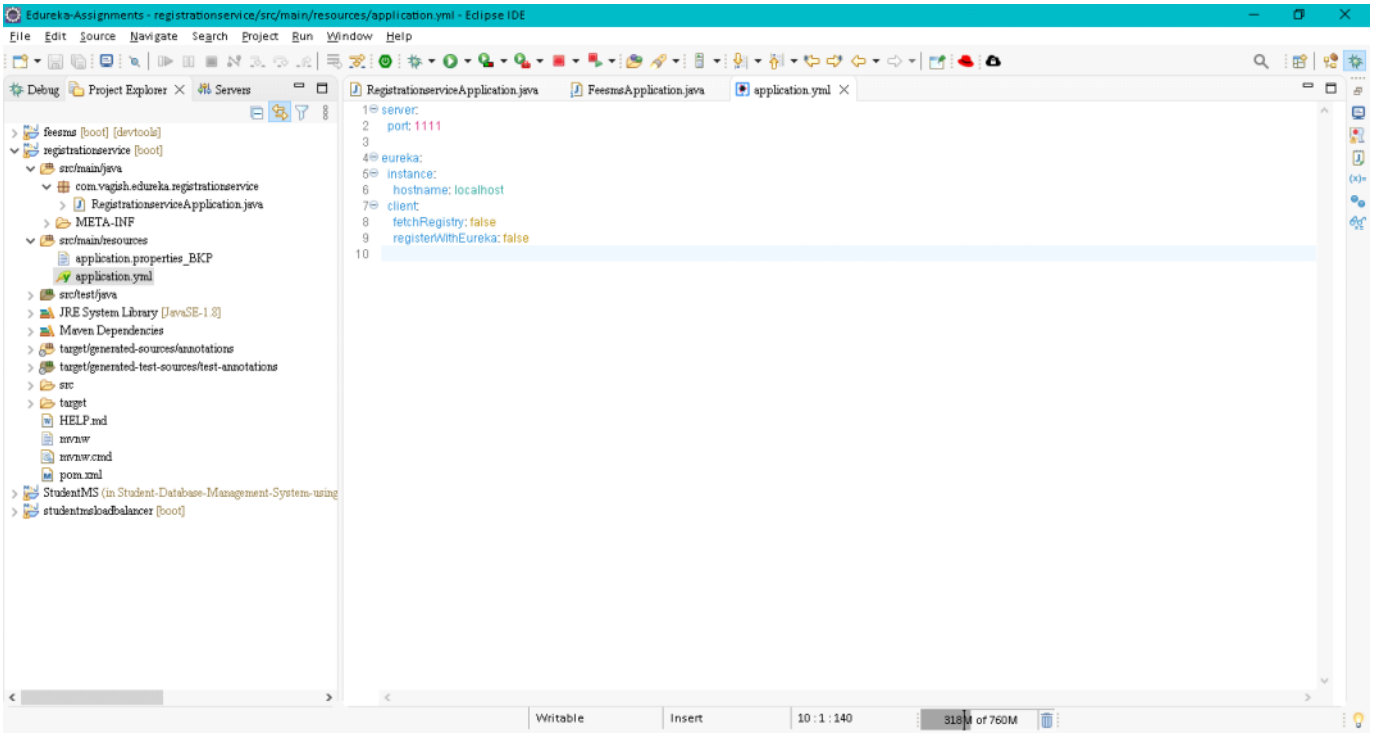
Registration Service

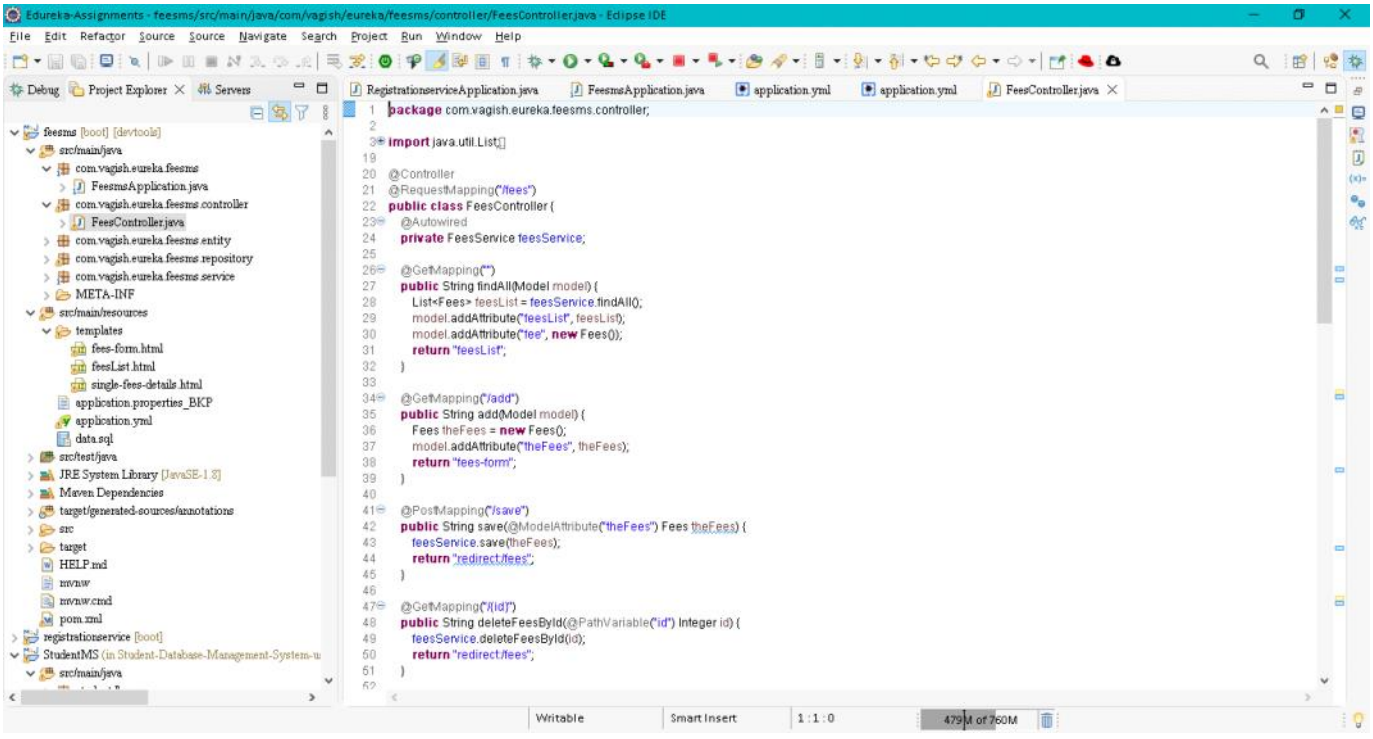


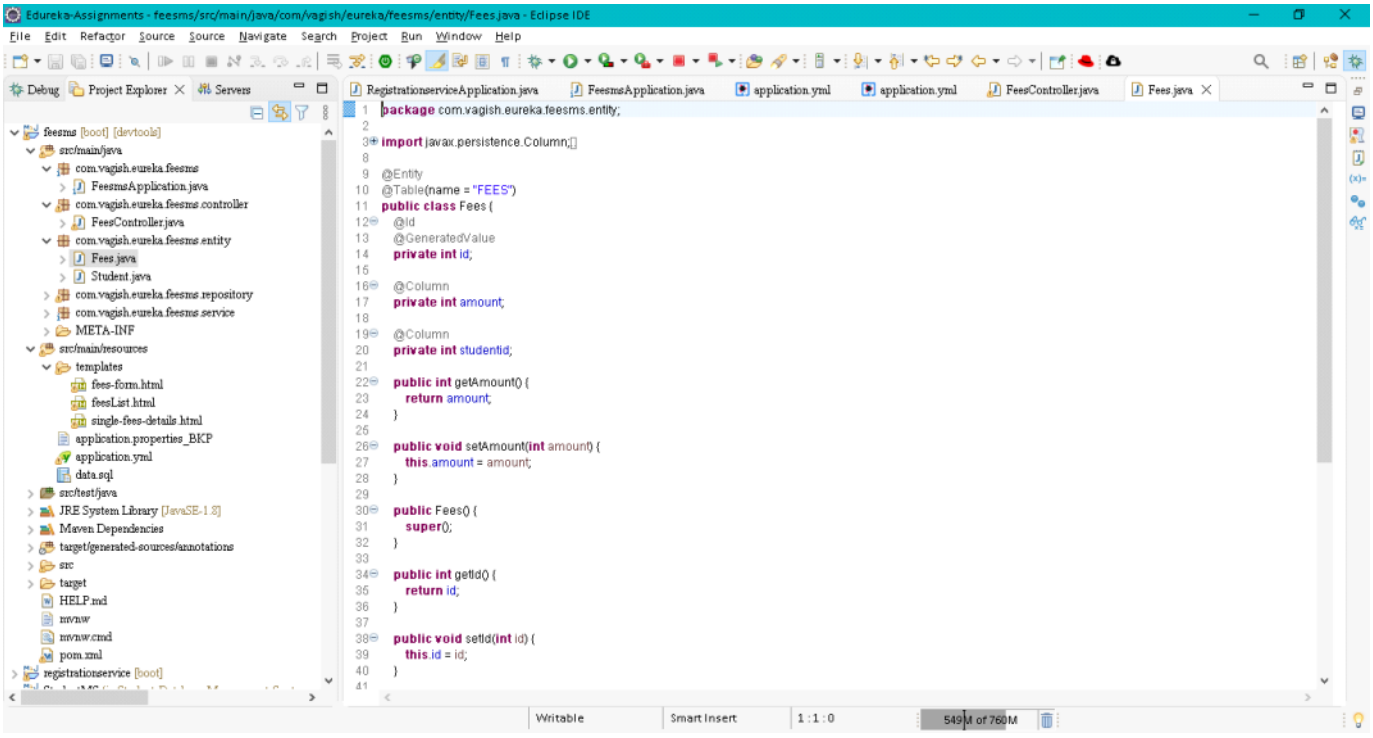

Edureka-Assignments - registrationservice/src/main/resources/application.yml - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
feesms [boot] [devtools]
registrationservice [boot]
src/main/java
com.vagish.edureka.registrationservice
RegistrationserviceApplication.java
META-INF
src/main/resources
application.properties_BKP
application.yml
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
target/generated-test-sources/test-annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
StudentMS (in Student-Database-Management-System-using
studentmsloadbalancer [boot]
RegistrationserviceApplication.java
FeesmsApplication.java
application.yml
server:
port: 1111
eureka:
instance:
hostname: localhost
client:
fetchRegistry: false
registerWithEureka: false
Writable
Insert
10 : 1 : 140
318M of 760M

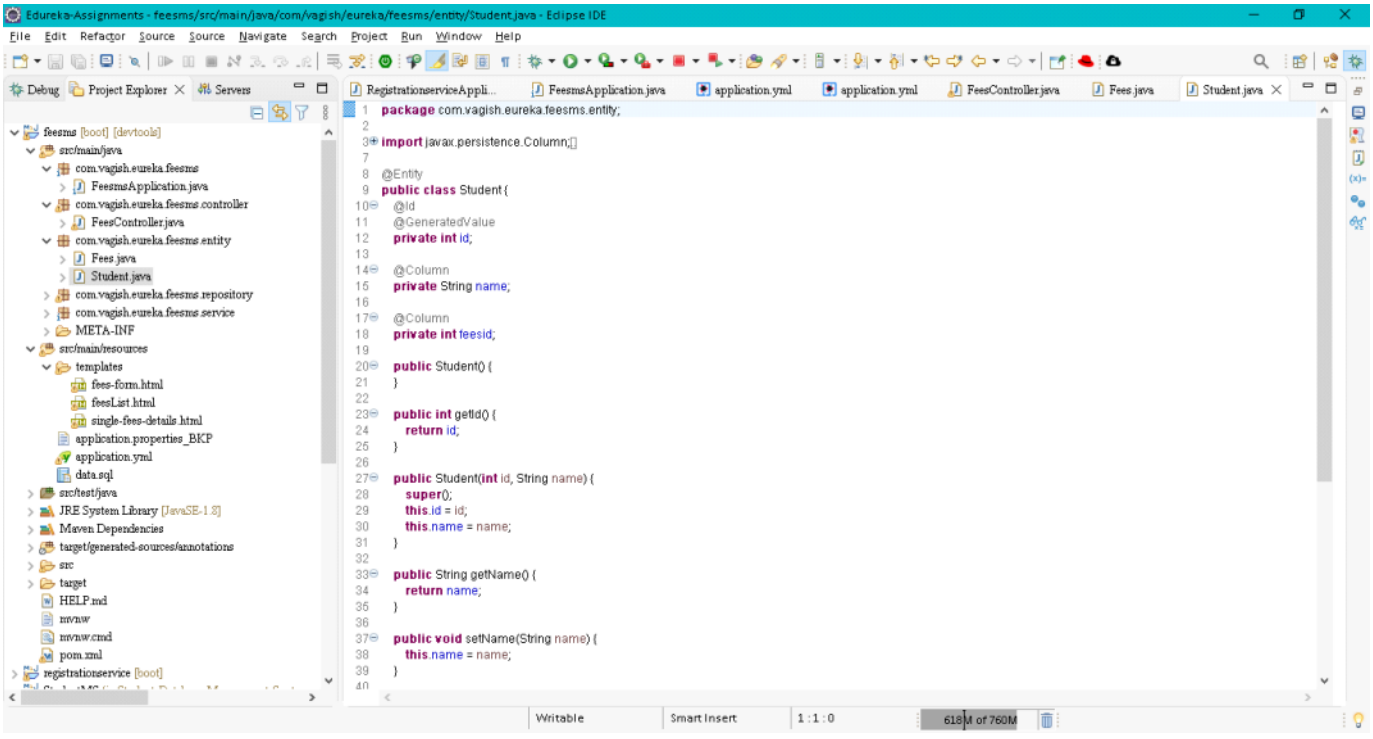
Fees MS

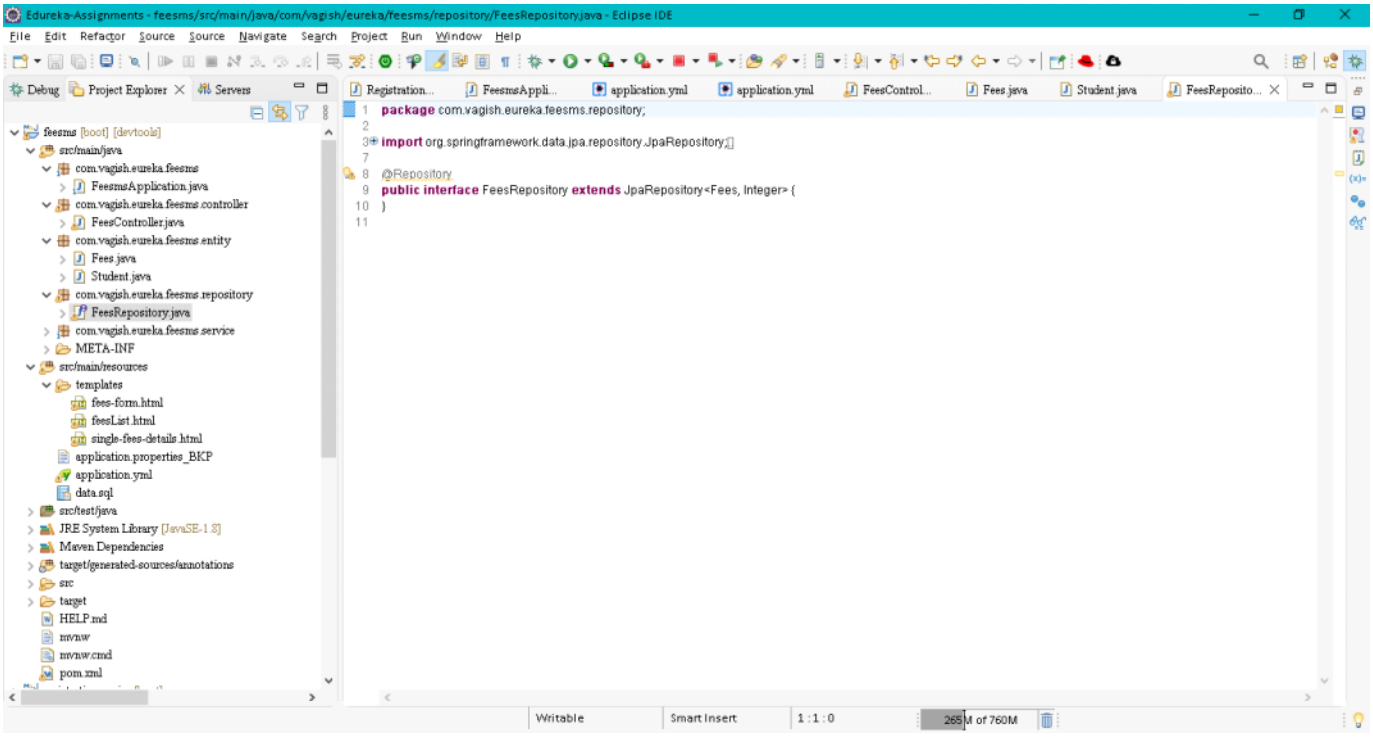

Edureka-Assignments - feesms/src/main/java/com/vagish/eureka/feesms/FeesmsApplication.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
feesms [boot] [devtools]
src/main/java
com.vagish.eureka.feesms
FeesmsApplication.java
com.vagish.eureka.feesms.controller
com.vagish.eureka.feesms.entity
com.vagish.eureka.feesms.repository
com.vagish.eureka.feesms.service
META-INF
src/main/resources
templates
fees-form.html
feesList.html
single-fees-details.html
application.properties_BKP
application.yml
data.sql
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
registrationservice [boot]
StudentMS (in Student-Database-Management-System-using
studentmsloadbalancer [boot]
RegistrationserviceApplication.java
FeesmsApplication.java
package com.vagish.eureka.feesms;
import org.springframework.beans.factory.annotation.Autowired;
@SpringBootApplication
@EnableDiscoveryClient
public class FeesmsApplication implements CommandLineRunner {
@Autowired
private FeesService feesService;
public static void main(String[] args) {
SpringApplication.run(FeesmsApplication.class, args);
}
@Override
public void run(String... args) throws Exception {
Fees fees = new Fees(0, 1000, 1);
if (null != this.feesService.getFeesById(0))
this.feesService.save(fees);
System.out.println(this.feesService.findAll().toString());
}
}
Writable
Smart Insert
1 : 1 : 0
595M of 760M

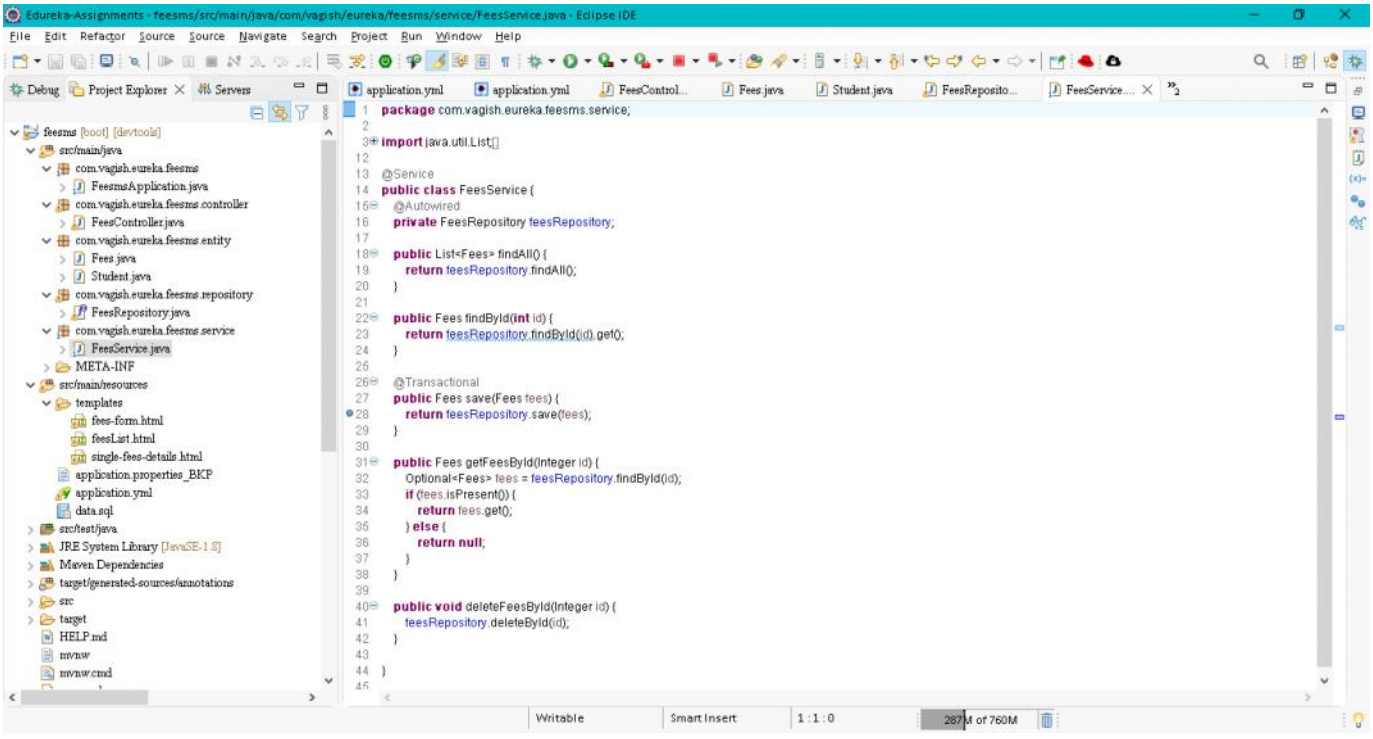

Edureka-Assignments - registrationservice/src/main/resources/application.yml - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
feesms [boot] [devtools]
registrationservice [boot]
src/main/java
com.vagish.edureka.registrationservice
RegistrationserviceApplication.java
META-INF
src/main/resources
application.properties_BKP
application.yml
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
target/generated-test-sources/test-annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
StudentMS (in Student-Database-Management-System-using
studentmsloadbalancer [boot]
RegistrationserviceApplication.java
FeesmsApplication.java
application.yml
server:
port: 1111
eureka:
instance:
hostname: localhost
client:
fetchRegistry: false
registerWithEureka: false
Writable
Insert
10 : 1 : 140
318M of 760M

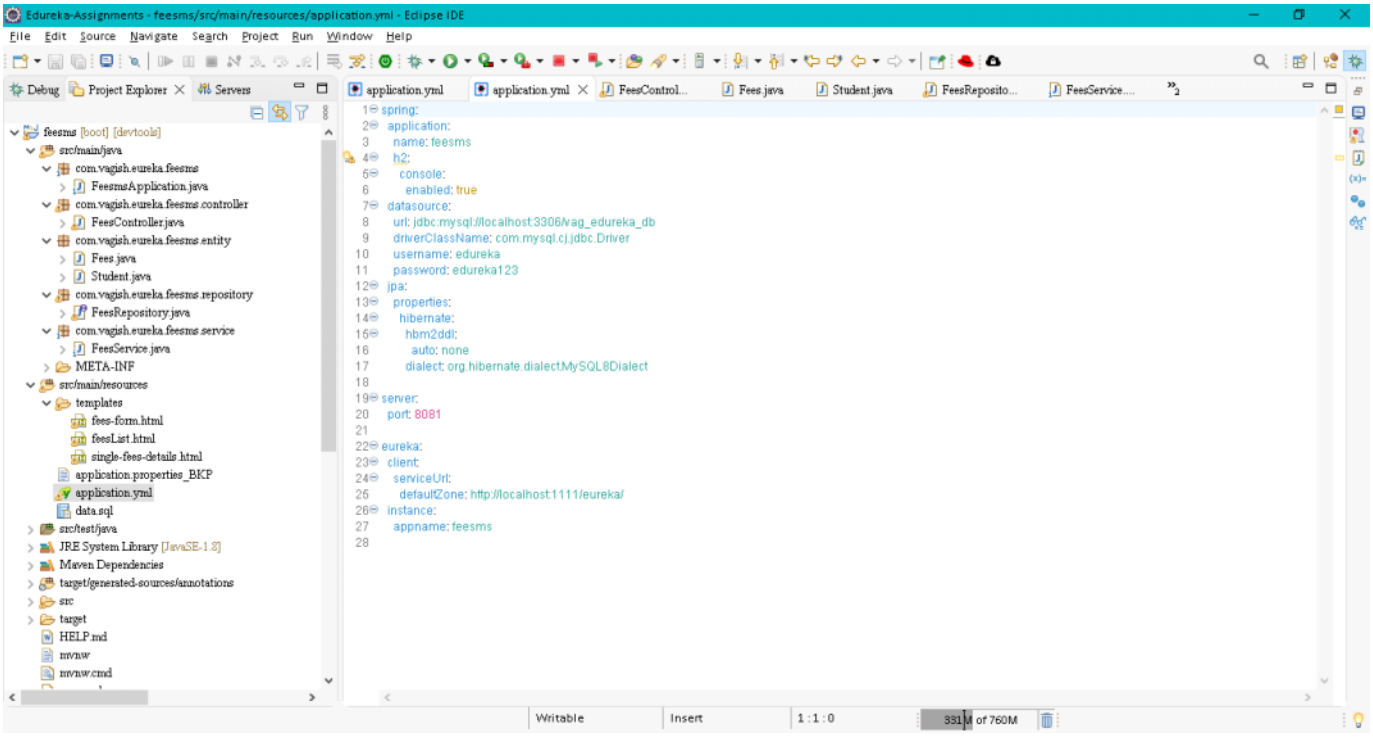

Edureka-Assignments - feesms/src/main/java/com/vagish/eureka/feesms/controller/FeesController.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
feesms [boot] [devtools]
src/main/java
com.vagish.eureka.feesms
FeesmsApplication.java
com.vagish.eureka.feesms.controller
FeesController.java
com.vagish.eureka.feesms.entity
com.vagish.eureka.feesms.repository
com.vagish.eureka.feesms.service
META-INF
src/main/resources
templates
fees-form.html
feesList.html
single-fees-details.html
application.properties_BKP
application.yml
data.sql
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
registrationservice [boot]
StudentMS (in Student-Database-Management-System-u
src/main/java
RegistrationserviceApplication.java
FeesmsApplication.java
application.yml
application.yml
FeesController.java
package com.vagish.eureka.feesms.controller;
import java.util.List;
@Controller
@RequestMapping("/fees")
public class FeesController {
@Autowired
private FeesService feesService;
@GetMapping("")
public String findAll(Model model) {
List<Fees> feesList = feesService.findAll();
model.addAttribute("feesList", feesList);
model.addAttribute("fee", new Fees());
return "feesList";
}
@GetMapping("/add")
public String add(Model model) {
Fees theFees = new Fees();
model.addAttribute("theFees", theFees);
return "fees-form";
}
@PostMapping("/save")
public String save(@ModelAttribute("theFees") Fees theFees) {
feesService.save(theFees);
return "redirect:/fees";
}
@GetMapping("/{id}")
public String deleteFeesById(@PathVariable("id") Integer id) {
feesService.deleteFeesById(id);
return "redirect:/fees";
}
Writable
Smart Insert
1 : 1 : 0
479M of 760M


Edureka-Assignments - feesms/src/main/java/com/vagish/eureka/feesms/entity/Fees.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
RegistrationserviceApplication.java FeesmsApplication.java application.yml application.yml FeesController.java Fees.java
feesms [boot] [devtools]
src/main/java
com.vagish.eureka.feesms
FeesmsApplication.java
com.vagish.eureka.feesms.controller
FeesController.java
com.vagish.eureka.feesms.entity
Fees.java
Student.java
com.vagish.eureka.feesms.repository
com.vagish.eureka.feesms.service
META-INF
src/main/resources
templates
fees-form.html
feesList.html
single-fees-details.html
application.properties_BKP
application.yml
data.sql
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
registrationservice [boot]
1 package com.vagish.eureka.feesms.entity;
3 import javax.persistence.Column;
8 @Entity
9 @Table(name = "FEES")
10 public class Fees {
11 @Id
12 @GeneratedValue
13 private int id;
15 @Column
16 private int amount;
18 @Column
19 private int studentid;
21 public int getAmount() {
22 return amount;
23 }
25 public void setAmount(int amount) {
26 this.amount = amount;
27 }
29 public Fees() {
30 super();
31 }
33 public int getId() {
34 return id;
35 }
37 public void setId(int id) {
38 this.id = id;
39 }
Writable
Smart Insert
1 : 1 : 0
549M of 760M

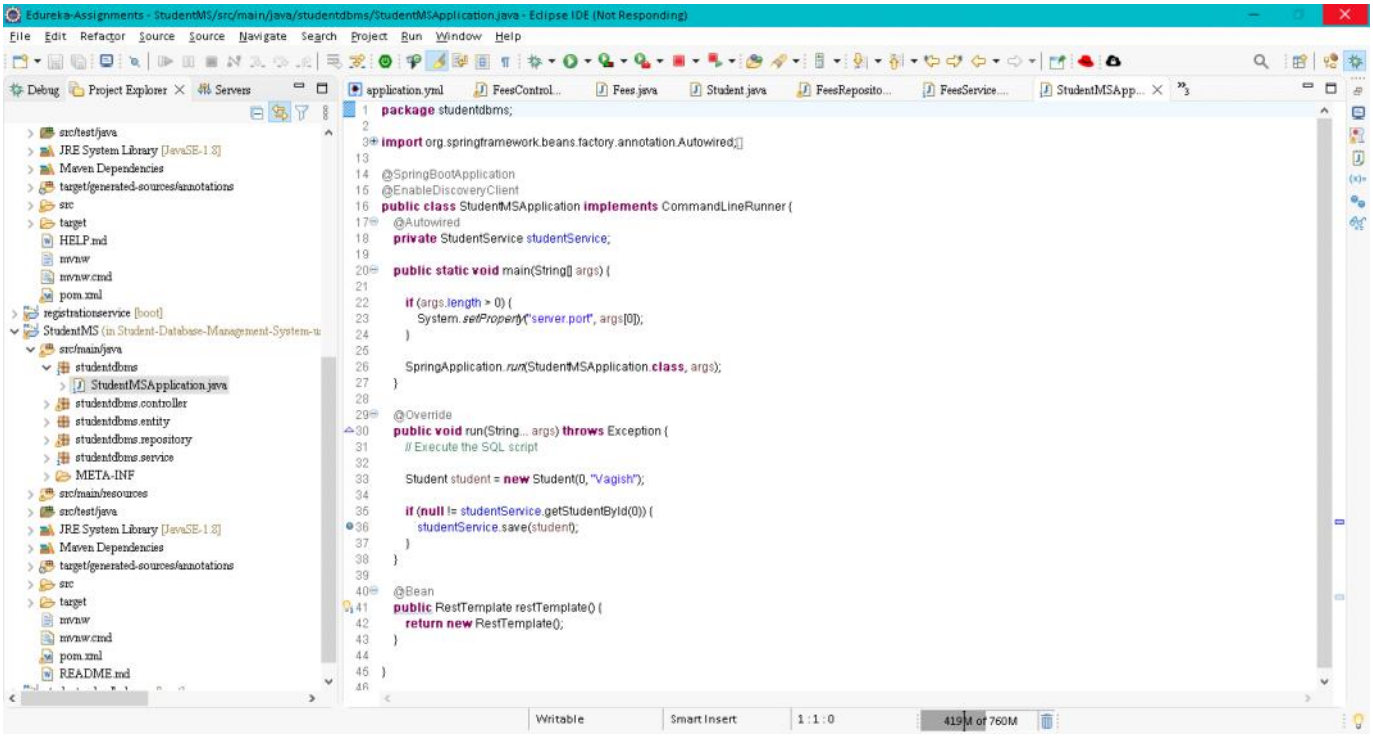

Edureka-Assignments - feesms/src/main/java/com/vagish/eureka/feesms/entity/Student.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
RegistrationserviceAppli... FeesmsApplication.java application.yml application.yml FeesController.java Fees.java Student.java
feesms [boot] [devtools]
src/main/java
com.vagish.eureka.feesms
FeesmsApplication.java
com.vagish.eureka.feesms.controller
FeesController.java
com.vagish.eureka.feesms.entity
Fees.java
Student.java
com.vagish.eureka.feesms.repository
com.vagish.eureka.feesms.service
META-INF
src/main/resources
templates
fees-form.html
feesList.html
single-fees-details.html
application.properties_BKP
application.yml
data.sql
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
registrationservice [boot]
1 package com.vagish.eureka.feesms.entity;
3 import javax.persistence.Column;
8 @Entity
9 public class Student {
10 @Id
11 @GeneratedValue
12 private int id;
14 @Column
15 private String name;
17 @Column
18 private int feesid;
20 public Student() {
21 }
23 public int getId() {
24 return id;
25 }
27 public Student(int id, String name) {
28 super();
29 this.id = id;
30 this.name = name;
31 }
33 public String getName() {
34 return name;
35 }
37 public void setName(String name) {
38 this.name = name;
39 }
Writable
Smart Insert
1 : 1 : 0
618M of 760M

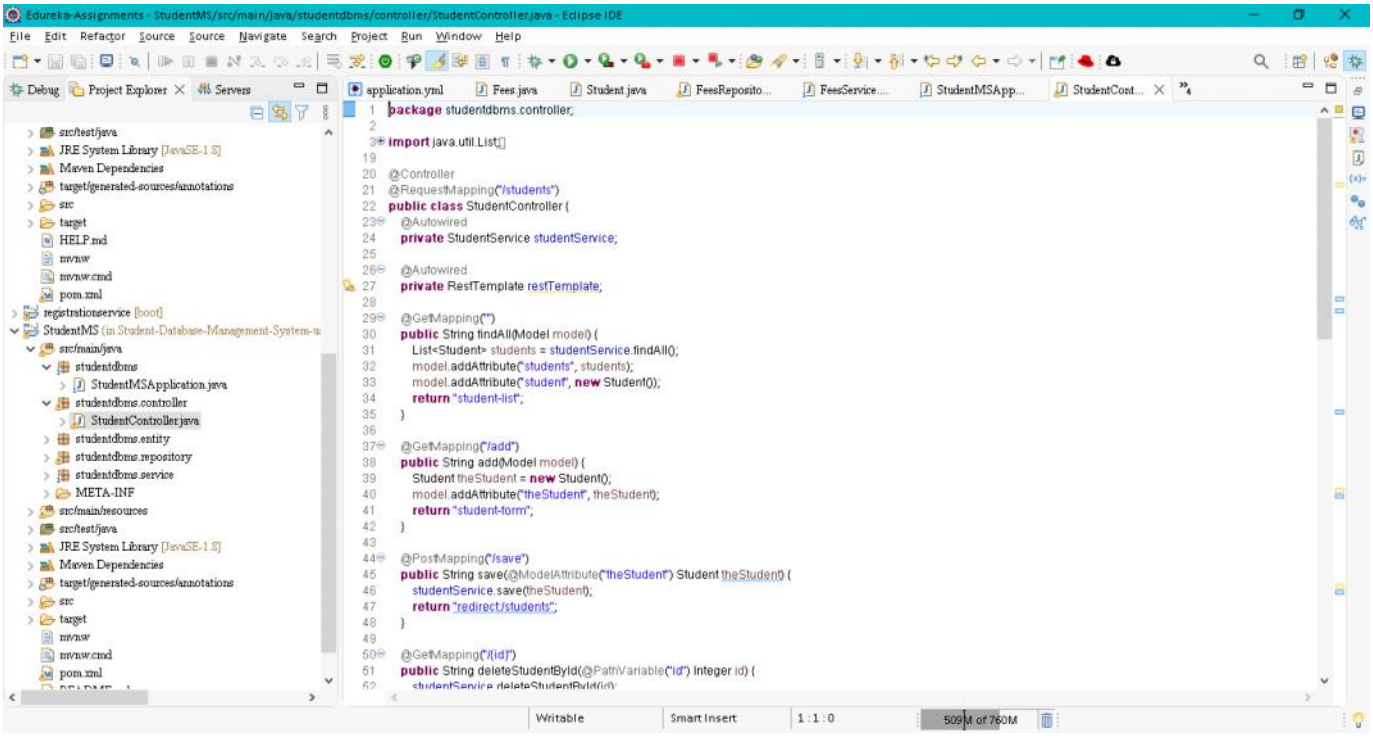

Edureka-Assignments - feesms/src/main/java/com/vagish/eureka/feesms/repository/FeesRepository.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
feesms [boot] [devtools]
src/main/java
com.vagish.eureka.feesms
FeesmsApplication.java
com.vagish.eureka.feesms.controller
FeesController.java
com.vagish.eureka.feesms.entity
Fees.java
Student.java
com.vagish.eureka.feesms.repository
FeesRepository.java
com.vagish.eureka.feesms.service
META-INF
src/main/resources
templates
fees-form.html
feesList.html
single-fees-details.html
application.properties_BKP
application.yml
data.sql
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
Registration...
FeesmsAppli...
application.yml
application.yml
FeesControl...
Fees.java
Student.java
FeesReposito...
1 package com.vagish.eureka.feesms.repository;
2
3 import org.springframework.data.jpa.repository.JpaRepository;
7
8 @Repository
9 public interface FeesRepository extends JpaRepository<Fees, Integer> {
10 }
11
Writable
Smart Insert
1 : 1 : 0
265M of 760M

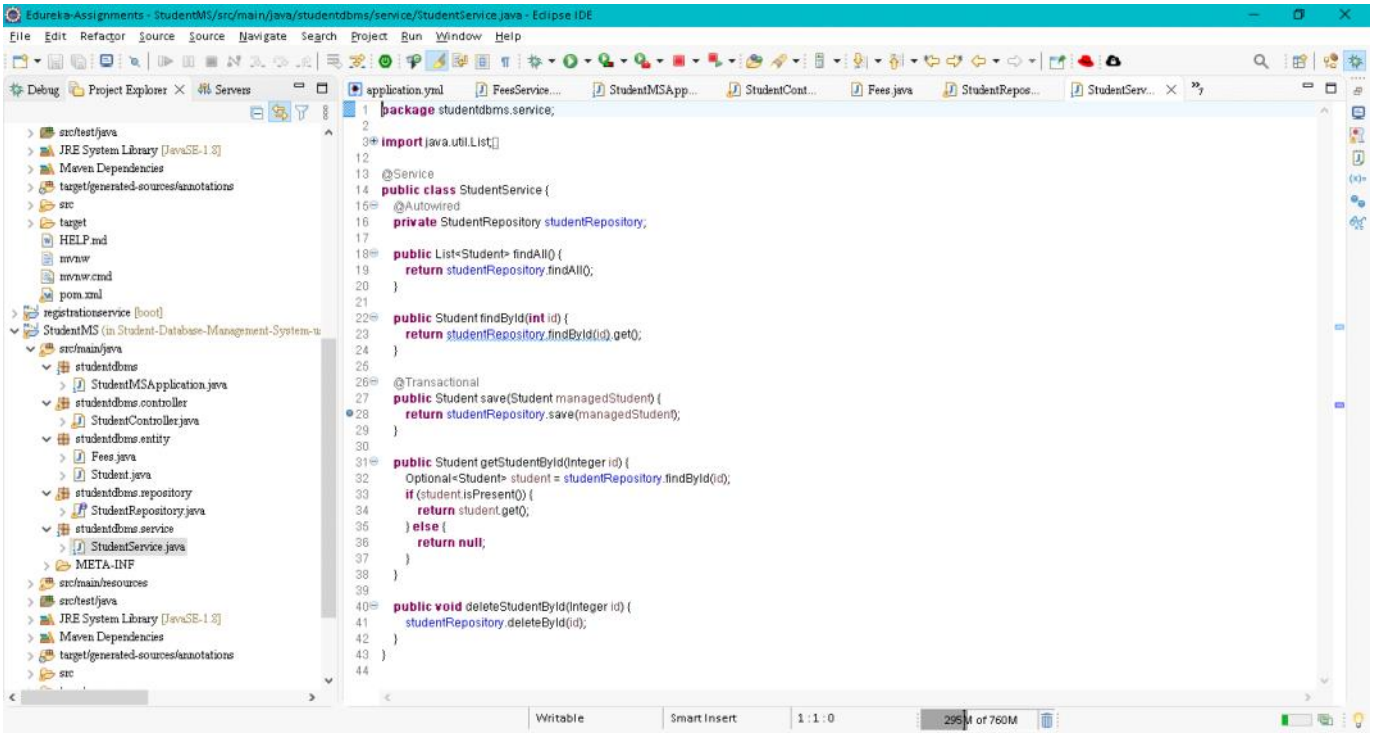

Edureka-Assignments - feesms/src/main/java/com/vagish/eureka/feesms/service/FeesService.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
feesms [boot] [devtools]
src/main/java
com.vagish.eureka.feesms
FeesmsApplication.java
com.vagish.eureka.feesms.controller
FeesController.java
com.vagish.eureka.feesms.entity
Fees.java
Student.java
com.vagish.eureka.feesms.repository
FeesRepository.java
com.vagish.eureka.feesms.service
FeesService.java
META-INF
src/main/resources
templates
fees-form.html
feesList.html
single-fees-details.html
application.properties_BKP
application.yml
data.sql
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
application.yml
application.yml
FeesControl...
Fees.java
Student.java
FeesReposito...
FeesService....
1 package com.vagish.eureka.feesms.service;
2
3 import java.util.List;
12
13 @Service
14 public class FeesService {
15 @Autowired
16 private FeesRepository feesRepository;
17
18 public List<Fees> findAll() {
19 return feesRepository.findAll();
20 }
21
22 public Fees findById(int id) {
23 return feesRepository.findById(id).get();
24 }
25
26 @Transactional
27 public Fees save(Fees fees) {
28 return feesRepository.save(fees);
29 }
30
31 public Fees getFeesById(Integer id) {
32 Optional<Fees> fees = feesRepository.findById(id);
33 if (fees.isPresent()) {
34 return fees.get();
35 } else {
36 return null;
37 }
38 }
39
40 public void deleteFeesById(Integer id) {
41 feesRepository.deleteById(id);
42 }
43
44 }
Writable
Smart Insert
1 : 1 : 0
287M of 760M

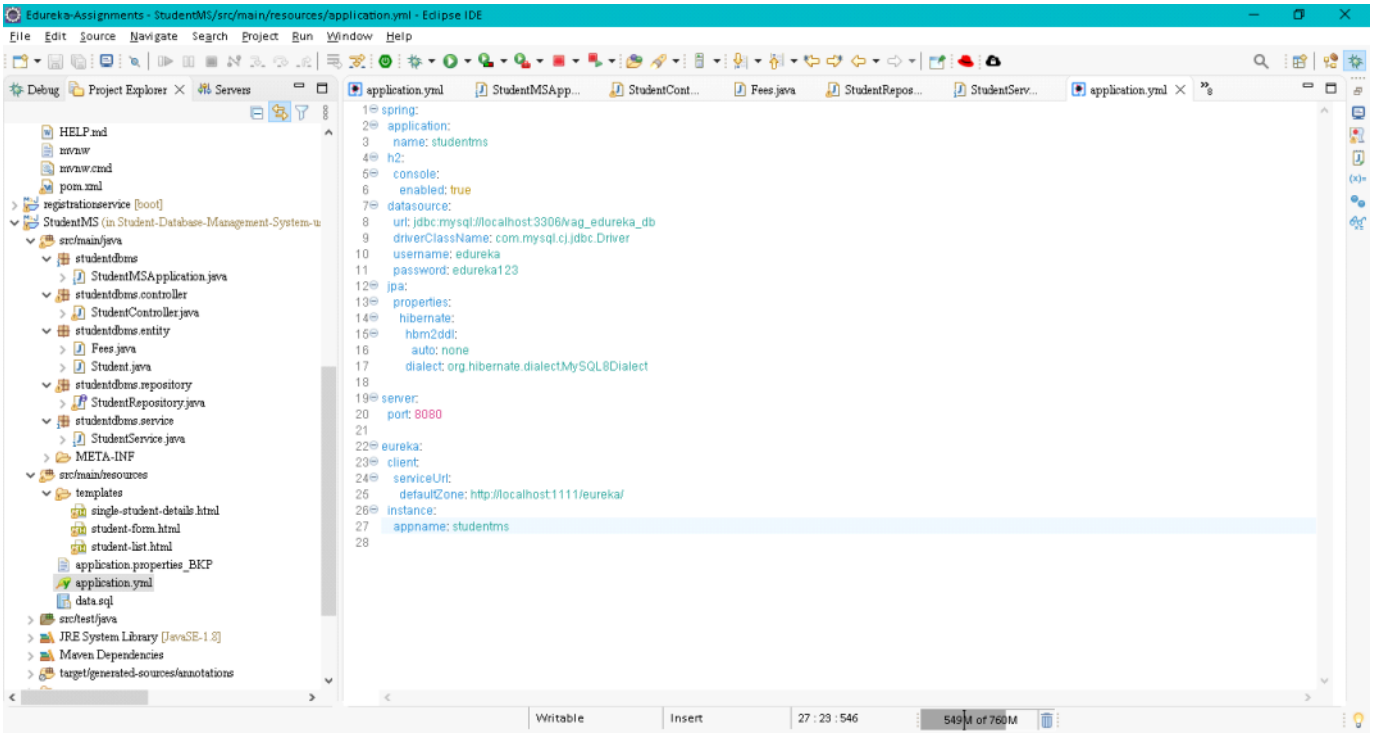

Edureka-Assignments - feesms/src/main/resources/application.yml - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
feesms [boot] [devtools]
src/main/java
com.vagish.eureka.feesms
FeesmsApplication.java
com.vagish.eureka.feesms.controller
FeesController.java
com.vagish.eureka.feesms.entity
Fees.java
Student.java
com.vagish.eureka.feesms.repository
FeesRepository.java
com.vagish.eureka.feesms.service
FeesService.java
META-INF
src/main/resources
templates
fees-form.html
feesList.html
single-fees-details.html
application.properties_BKP
application.yml
data.sql
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
application.yml
FeesControl...
Fees.java
Student.java
FeesReposito...
FeesService....
spring:
application:
name: feesms
h2:
console:
enabled: true
datasource:
url: jdbc:mysql://localhost:3306/vag_edureka_db
driverClassName: com.mysql.cj.jdbc.Driver
username: edureka
password: edureka123
jpa:
properties:
hibernate:
hbm2ddl:
auto: none
dialect: org.hibernate.dialect.MySQL8Dialect
server:
port: 8081
eureka:
client:
serviceUrl:
defaultZone: http://localhost:1111/eureka/
instance:
appname: feesms
Writable
Insert
1 : 1 : 0
331M of 760M

Student MS

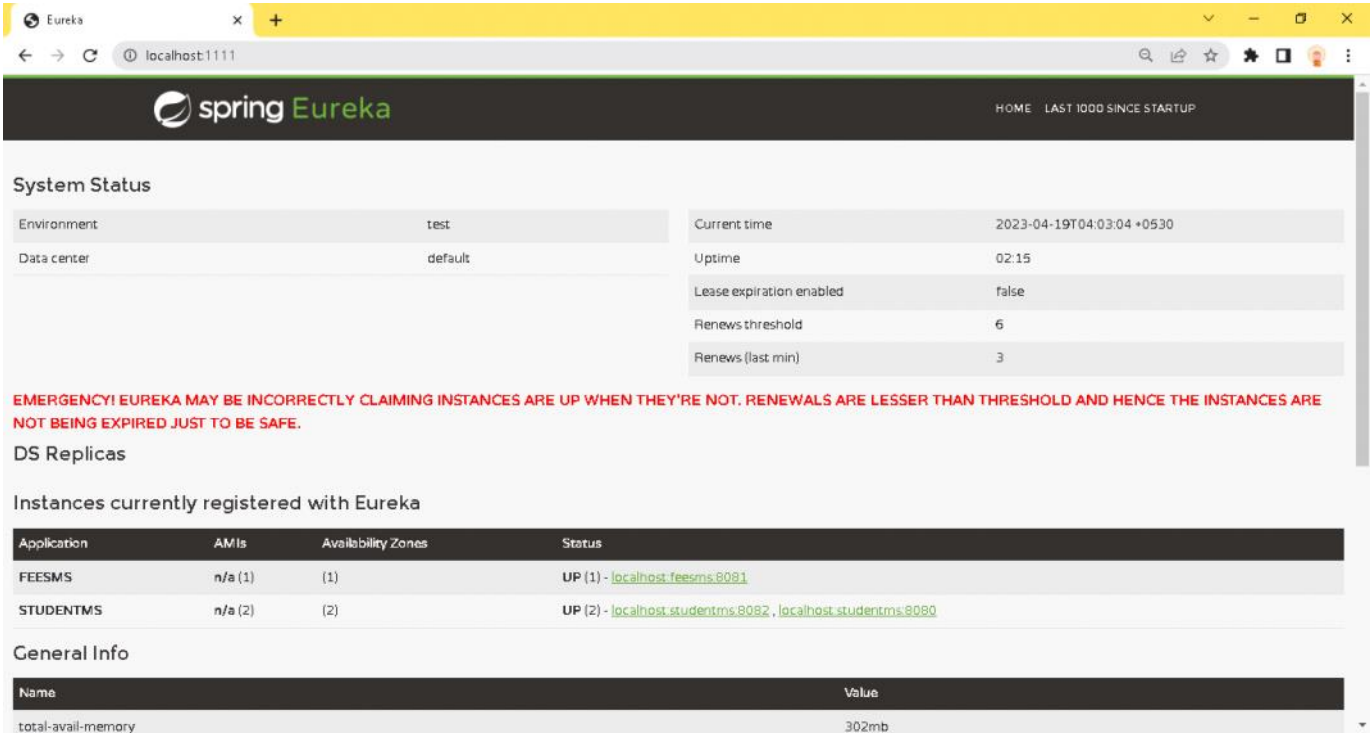

Edureka-Assignments - StudentMS/src/main/java/studentdbms/StudentMSApplication.java - Eclipse IDE (Not Responding)
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
registrationservice [boot]
StudentMS (in Student-Database-Management-System-u
src/main/java
studentdbms
StudentMSApplication.java
studentdbms.controller
studentdbms.entity
studentdbms.repository
studentdbms.service
META-INF
src/main/resources
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
mvnw
mvnw.cmd
pom.xml
README.md
application.yml
FeesControl...
Fees.java
Student.java
FeesReposito...
FeesService....
StudentMSApp...
package studentdbms;
import org.springframework.beans.factory.annotation.Autowired;
@SpringBootApplication
@EnableDiscoveryClient
public class StudentMSApplication implements CommandLineRunner {
@Autowired
private StudentService studentService;
public static void main(String[] args) {
if (args.length > 0) {
System.setProperty("server.port", args[0]);
}
SpringApplication.run(StudentMSApplication.class, args);
}
@Override
public void run(String... args) throws Exception {
// Execute the SQL script
Student student = new Student(0, "Vagish");
if (null != studentService.getStudentById(0)) {
studentService.save(student);
}
}
@Bean
public RestTemplate restTemplate() {
return new RestTemplate();
}
}
Writable
Smart Insert
1 : 1 : 0
419M of 760M


Edureka-Assignments - StudentMS/src/main/java/studentdbms/controller/StudentController.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
application.yml Fees.java Student.java FeesReposito... FeesService.... StudentMSApp... StudentCont...
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
registrationservice [boot]
StudentMS (in Student-Database-Management-System-u
src/main/java
studentdbms
StudentMSApplication.java
studentdbms.controller
StudentController.java
studentdbms.entity
studentdbms.repository
studentdbms.service
META-INF
src/main/resources
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
mvnw
mvnw.cmd
pom.xml
1 package studentdbms.controller;
2
3 import java.util.List;
19
20 @Controller
21 @RequestMapping("/students")
22 public class StudentController {
23 @Autowired
24 private StudentService studentService;
25
26 @Autowired
27 private RestTemplate restTemplate;
28
29 @GetMapping("")
30 public String findAll(Model model) {
31 List<Student> students = studentService.findAll();
32 model.addAttribute("students", students);
33 model.addAttribute("student", new Student());
34 return "student-list";
35 }
36
37 @GetMapping("/add")
38 public String add(Model model) {
39 Student theStudent = new Student();
40 model.addAttribute("theStudent", theStudent);
41 return "student-form";
42 }
43
44 @PostMapping("/save")
45 public String save(@ModelAttribute("theStudent") Student theStudent) {
46 studentService.save(theStudent);
47 return "redirect:/students";
48 }
49
50 @GetMapping("/{id}")
51 public String deleteStudentById(@PathVariable("id") Integer id) {
Writable
Smart Insert
1 : 1 : 0
509M of 760M

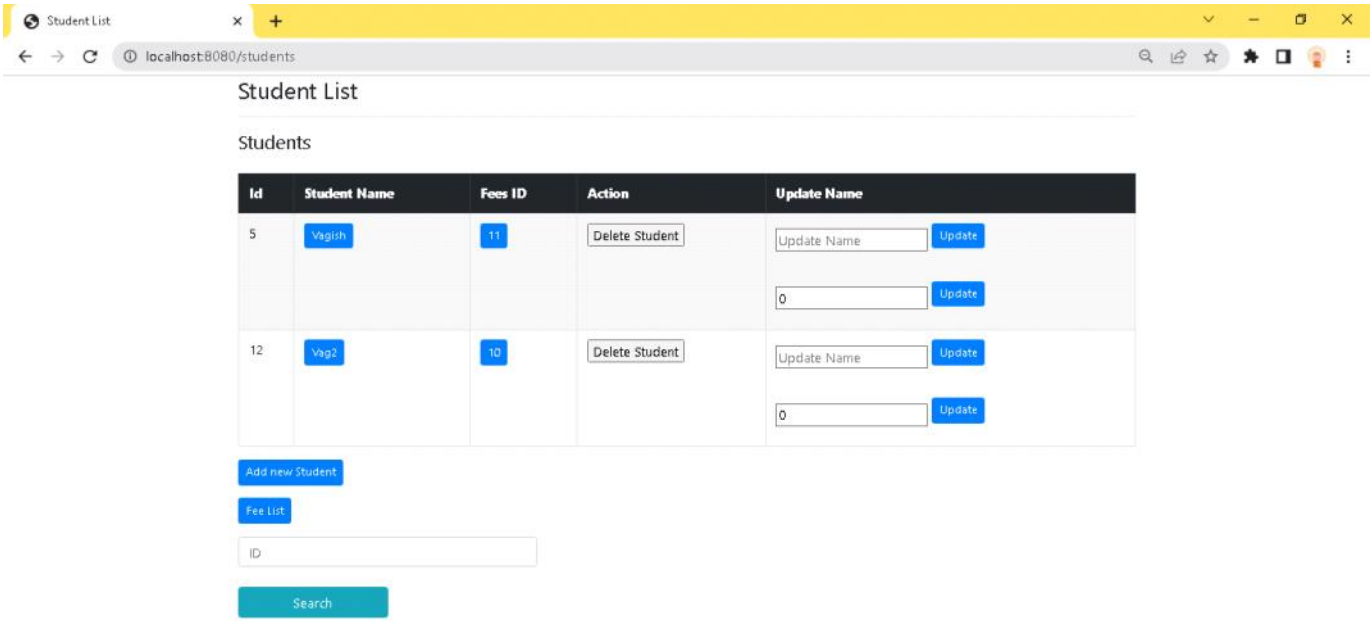

Edureka-Assignments - StudentMS/src/main/java/studentdbms/service/StudentService.java - Eclipse IDE
File Edit Refactor Source Source Navigate Search Project Run Window Help
Debug Project Explorer Servers
application.yml FeesService.... StudentMSApp... StudentCont... Fees.java StudentRepos... StudentServ...
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
registrationservice [boot]
StudentMS (in Student-Database-Management-System-u
src/main/java
studentdbms
StudentMSApplication.java
studentdbms.controller
StudentController.java
studentdbms.entity
Fees.java
Student.java
studentdbms.repository
StudentRepository.java
studentdbms.service
StudentService.java
META-INF
src/main/resources
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
src
1 package studentdbms.service;
2
3 import java.util.List;
12
13 @Service
14 public class StudentService {
15 @Autowired
16 private StudentRepository studentRepository;
17
18 public List<Student> findAll() {
19 return studentRepository.findAll();
20 }
21
22 public Student findById(int id) {
23 return studentRepository.findById(id).get();
24 }
25
26 @Transactional
27 public Student save(Student managedStudent) {
28 return studentRepository.save(managedStudent);
29 }
30
31 public Student getStudentById(Integer id) {
32 Optional<Student> student = studentRepository.findById(id);
33 if (student.isPresent()) {
34 return student.get();
35 } else {
36 return null;
37 }
38 }
39
40 public void deleteStudentById(Integer id) {
41 studentRepository.deleteById(id);
42 }
43 }
44
Writable
Smart Insert
1 : 1 : 0
295M of 760M

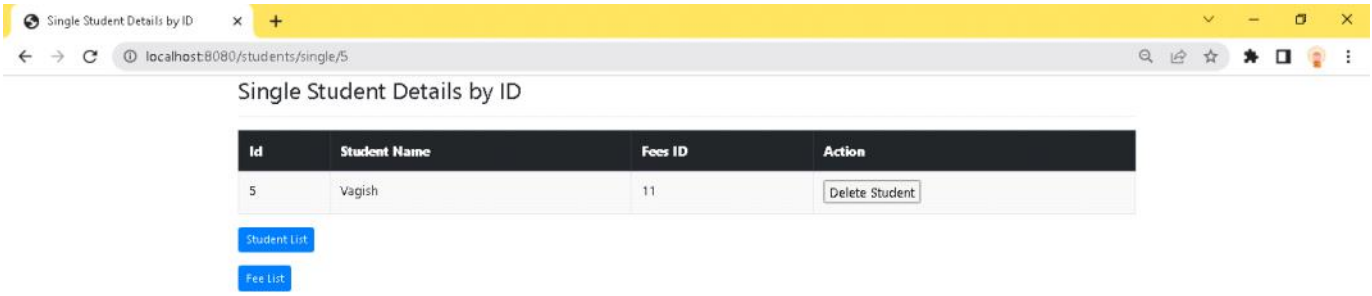


Eureka Server Output:

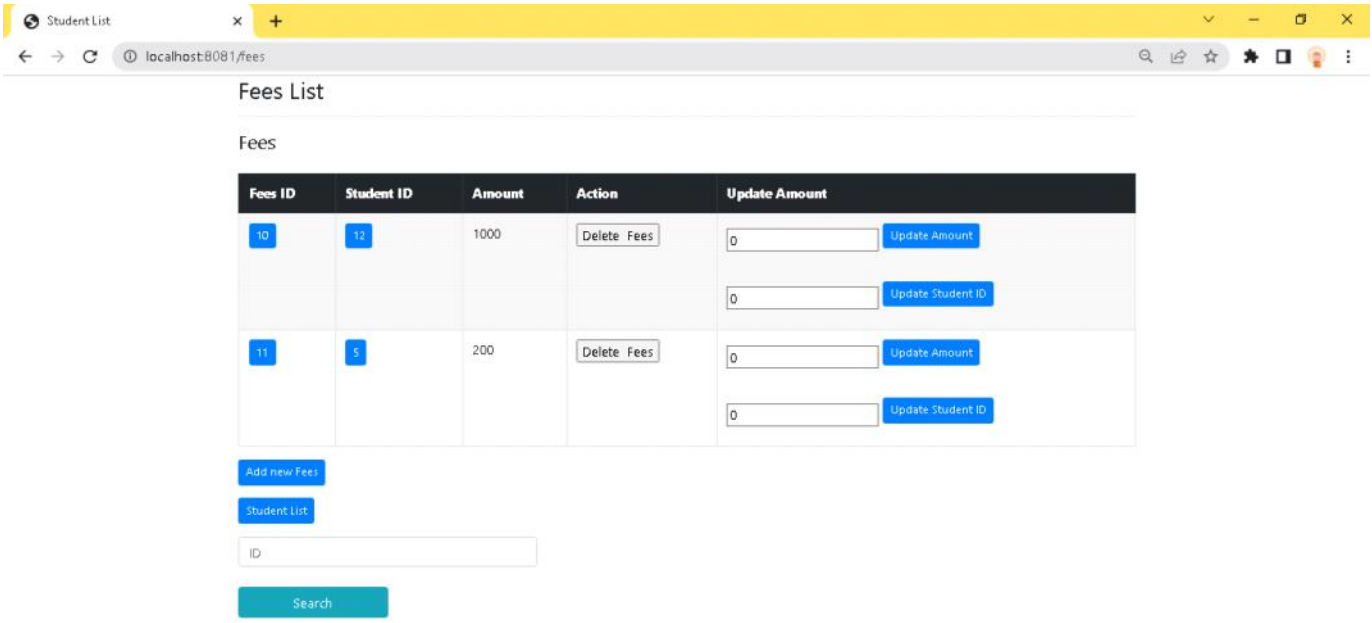


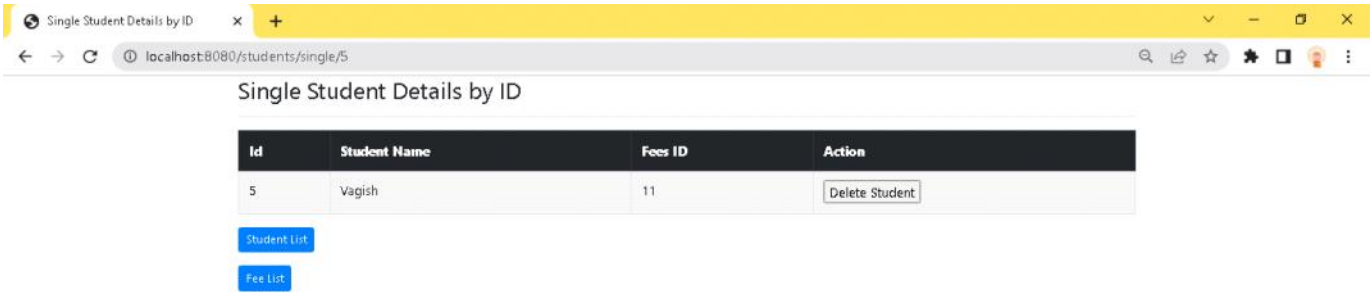
Student MS Output:



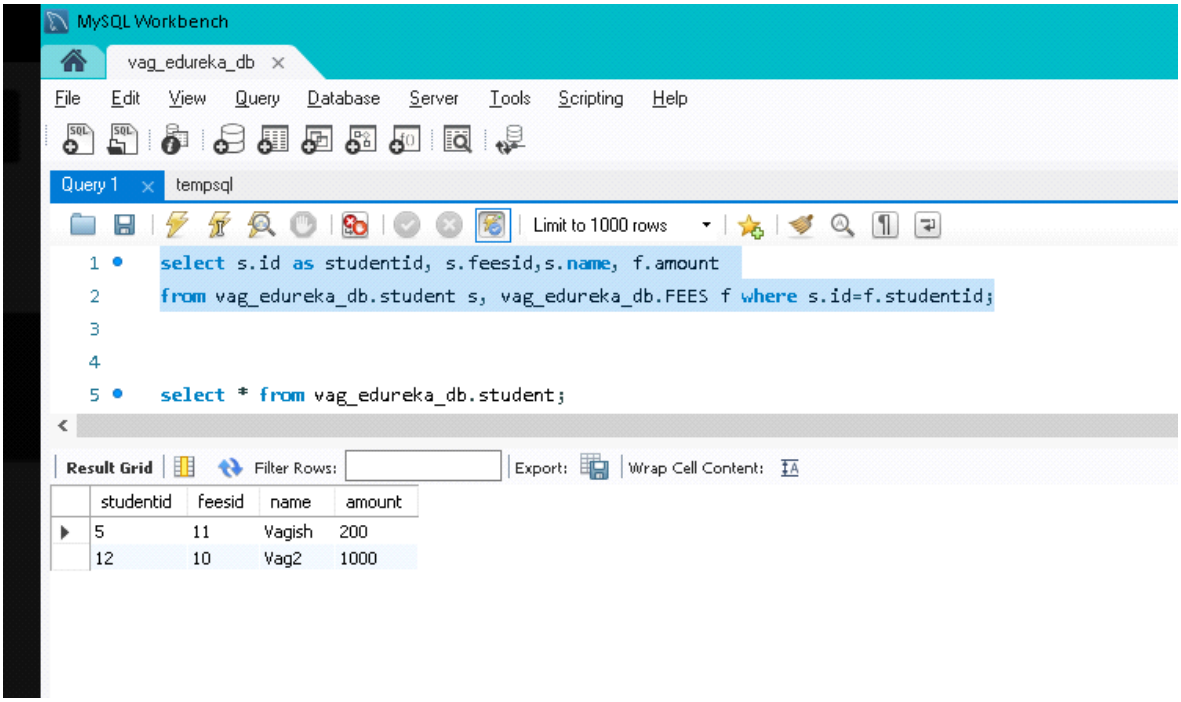


Fees MS Output





Database Output:



Uploaded PDF with consolidated Screenshots to create Student MS, Eureka Registry Service and Client Side Load Balancer to perform all the mentioned Tasks.

Also, all the related Maven Codes, result Screenshot PDF is saved on GIT Repo:
https://github.com/Vagish619/Edureka-Microservices.git

Branch Name:
Assignments_And_Projects