Sem : MSc ICT 3rd Sem

Name :- Reshamwala Vivek Yogeshkumar

Roll No :- 52

Q.2. Do the following using Micro services architecture 30 marks

There are two services

1. A Customer Service which maintains the record of customers and interacts with a table customer\_master having field customer\_id, email, name, rating, address and phone number. The rating is between 1 to 10 It has a method with the following prototype

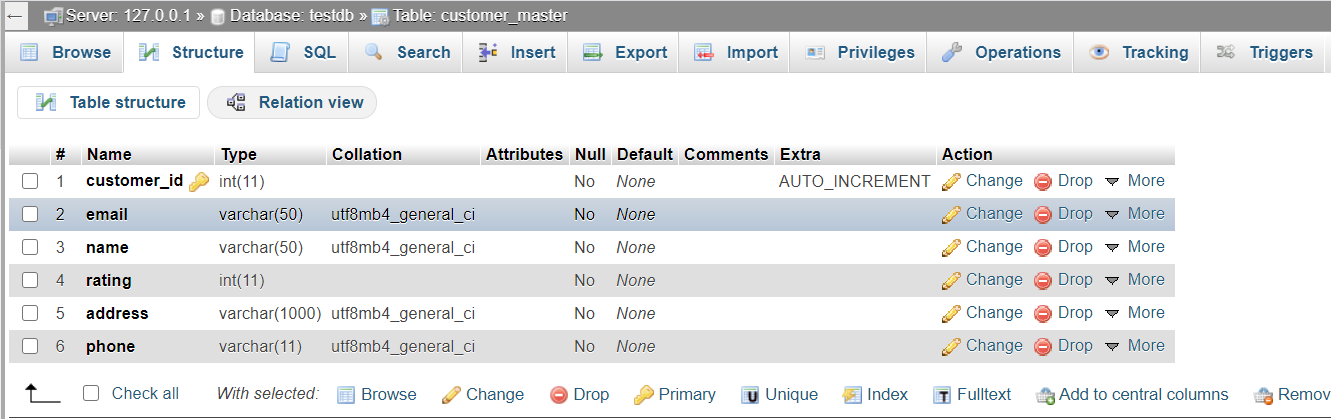
Collection<Customer> getCustomersWithRating(String condition, int rating);

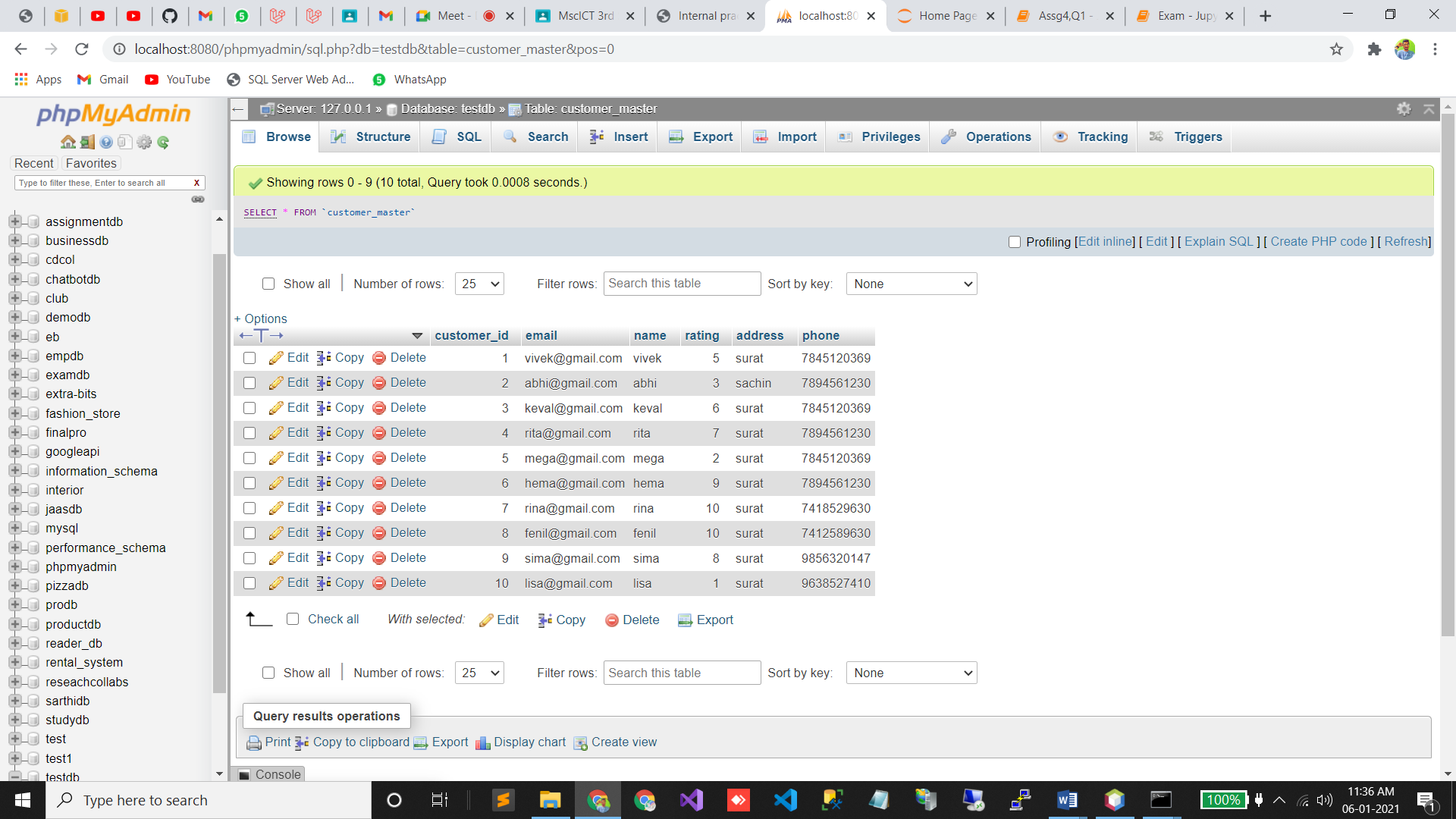
the possible values of condition can only be - less than, greater than, lesss than equal to, greater than equal to and Equal to

2. A client service which has a front end to get List of customers with certain rating. This displays a query page with a form having drop down box containing choices (less than, greater than, lesss than equal to, greater than equal to and Equal to) and a text box to write rating we want. On Submiting this form one gets the list of customer details in form of a table which satisfies the criteria given in the page.

Note : No Security is required . You can use JPA, Servlets, JSF etc.

Database:-



0

Services :-

package com.mycompany.mavenproject1.service;

import com.mycompany.mavenproject1.entity.CustomerMaster;

import java.util.Collection;

import javax.persistence.EntityManager;

import javax.persistence.Persistence;

import javax.ws.rs.GET;

import javax.ws.rs.Path;

import javax.ws.rs.PathParam;

import javax.ws.rs.Produces;

import javax.ws.rs.core.MediaType;

@Path("/test")

public class ExampleService {

EntityManager em=Persistence.createEntityManagerFactory("examPU").createEntityManager();

@GET

@Path("getCustomerMaster")

@Produces(MediaType.APPLICATION\_JSON)

public Collection<CustomerMaster> getCustomerMaster() {

return em.createNamedQuery("CustomerMaster.findAll").getResultList();

}

@GET

@Produces(MediaType.APPLICATION\_JSON)

@Path("getCustomersWithRating/{condition}/{rating}")

public Collection<CustomerMaster> getCustomersWithRating(@PathParam("condition")String condition,@PathParam("rating") int rating){

Collection<CustomerMaster> customerMasters;

if(condition=="Greater")

{

customerMasters=em.createNamedQuery("CustomerMaster.findByRatingGreater").setParameter("rating",rating).getResultList();

}

else if(condition == "Less")

{

customerMasters=em.createNamedQuery("CustomerMaster.findByRatingLess").setParameter("rating",rating).getResultList();

}

else if(condition == "Greater Equal"){

customerMasters=em.createNamedQuery("CustomerMaster.findByRatingGreaterThanEqual").setParameter("rating",rating).getResultList();

}

else if(condition == "Less Equal"){

customerMasters=em.createNamedQuery("CustomerMaster.findByRatingLessThanEqual").setParameter("rating",rating).getResultList();

}

else{

customerMasters=em.createNamedQuery("CustomerMaster.findByRating").setParameter("rating",rating).getResultList();

}

return customerMasters;

}

}

Client App :-

customerClient

package com.mycompany.client\_app1.client;

import com.mycompany.client\_app1.entity.CustomerMaster;

import java.util.Collection;

import javax.ws.rs.GET;

import javax.ws.rs.Path;

import javax.ws.rs.PathParam;

import javax.ws.rs.Produces;

import javax.ws.rs.core.MediaType;

import org.eclipse.microprofile.rest.client.inject.RegisterRestClient;

/\*\*

\*

\* @author LENOVO

\*/

@RegisterRestClient(baseUri = "http://localhost:9090/servicesApp1/rest/test")

public interface customerClient {

@GET

@Path("getCustomerMaster")

@Produces(MediaType.APPLICATION\_JSON)

public Collection<CustomerMaster> getCustomerMaster();

@GET

@Produces(MediaType.APPLICATION\_JSON)

@Path("getCustomersWithRating/{condition}/{rating}")

public Collection<CustomerMaster> getCustomersWithRating(@PathParam("condition")String condition,@PathParam("rating") int rating);

}

Bean:-

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.mycompany.client\_app1.bean;

import com.mycompany.client\_app1.client.customerClient;

import com.mycompany.client\_app1.entity.CustomerMaster;

import java.util.Collection;

import javax.inject.Named;

import javax.enterprise.context.RequestScoped;

import javax.inject.Inject;

import org.eclipse.microprofile.rest.client.inject.RestClient;

/\*\*

\*

\* @author LENOVO

\*/

@Named(value = "customerBean")

@RequestScoped

public class CustomerBean {

@Inject @RestClient customerClient cc;

Collection<CustomerMaster> customerMasters;

String condition,rating;

public String getCondition() {

return condition;

}

public void setCondition(String condition) {

this.condition = condition;

}

public String getRating() {

return rating;

}

public void setRating(String rating) {

this.rating = rating;

}

public Collection<CustomerMaster> getCustomerMasters() {

return customerMasters;

}

public void setCustomerMasters(Collection<CustomerMaster> customerMasters) {

this.customerMasters = customerMasters;

}

/\*\*

\* Creates a new instance of CustomerBean

\*/

public CustomerBean() {

try{

customerMasters=cc.getCustomerMaster();

}catch(Exception e){ e.printStackTrace(); }

}

public void customerMasterRating()

{

customerMasters=null;

customerMasters=cc.getCustomersWithRating(condition,Integer.parseInt(rating));

//return "index.xhtml";

}

}

Index.xhtml

<?xml version='1.0' encoding='UTF-8' ?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"

xmlns:h="http://xmlns.jcp.org/jsf/html"

xmlns:f="http://xmlns.jcp.org/jsf/core">

<h:head>

<title>Facelet Title</title>

</h:head>

<h:body>

<h:form>

Select condition

<h:selectOneMenu value="#{ customerBean.condition }" required="">

<f:selectItem itemLabel="Greater" itemValue="Greater" />

<f:selectItem itemLabel="Less" itemValue="Less" />

<f:selectItem itemLabel="Greater Equal" itemValue="Greater Equal" />

<f:selectItem itemLabel="Less Equal" itemValue="Less Equal" />

<f:selectItem itemLabel="Equal" itemValue="Equal" />

</h:selectOneMenu>

Rating :- <h:inputText id="rating" value="${customerBean.rating}" required=""/>

<h:commandButton action="#{customerBean.customerMasterRating()}" value="Submit"/>

</h:form>

<f:view>

<h:form>

<h1><h:outputText value="List"/></h1>

<h:dataTable value="#{customerBean.customerMasters}" var="item">

<h:column>

<f:facet name="header">

<h:outputText value="CustomerId"/>

</f:facet>

<h:outputText value="#{item.customerId}"/>

</h:column>

<h:column>

<f:facet name="header">

<h:outputText value="Email"/>

</f:facet>

<h:outputText value="#{item.email}"/>

</h:column>

<h:column>

<f:facet name="header">

<h:outputText value="Name"/>

</f:facet>

<h:outputText value="#{item.name}"/>

</h:column>

<h:column>

<f:facet name="header">

<h:outputText value="Rating"/>

</f:facet>

<h:outputText value="#{item.rating}"/>

</h:column>

<h:column>

<f:facet name="header">

<h:outputText value="Address"/>

</f:facet>

<h:outputText value="#{item.address}"/>

</h:column>

<h:column>

<f:facet name="header">

<h:outputText value="Phone"/>

</f:facet>

<h:outputText value="#{item.phone}"/>

</h:column>

</h:dataTable>

</h:form>

</f:view>

</h:body>

</html>

