**Answer Booklet**



M251-M257: Object Oriented Programming Using Java

|  |  |
| --- | --- |
| **Personal Information** | |
| **Student Name** | mohamed fathi abdelraouf |
| **Student ID** | 1851711489 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Registration Information** | | | |
| **Branch** |  | | |
| **Section Number** |  | **Assessment type** | Final-THE |
| **Academic Year** | 2020/2021 | **Semester** | Spring |

|  |
| --- |
| **General Note** |
|  |

Below, you can start your answer…

**Tracing:**

value of integer details when the return 19 is 199 and the explain the method is received integer number and check if this number is < 10 this check number that received is % 2 not equals 0 it means this number is odd it returns this number

but if the result equal 0 it means this number is even and return 0

so that if this number is >= 10

this check again but with the same scenario but the return is different

this check number that received is % 2 not equals 0 it means this number is odd it returns this number % 10 multiply the recursive method and repeat these instructions again

but if the result equals 0 it means this number is even and return

the recursive method and repeat these instructions again

**Java Project:**

**Implementation:**

import java.util.ArrayList;

import java.util.List;

public class WorkShop {

private String name;

private List<Attendee> appendables = new ArrayList<>();

public WorkShop() {

}

public WorkShop(String name) {

this.name = name;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public List<Attendee> getAppendables() {

return appendables;

}

public void setAppendables(List<Attendee> appendables) {

this.appendables = appendables;

}

}

public class Attendee {

private long passportNumber;

private String fullName;

private String number;

private String age;

public Attendee() {

}

public Attendee(long passportNumber, String fullName, String number, String age) {

this.passportNumber = passportNumber;

this.fullName = fullName;

this.number = number;

this.age = age;

}

public String getAge() {

return age;

}

public void setAge(String age) {

this.age = age;

}

public long getPassportNumber() {

return passportNumber;

}

public void setPassportNumber(long passportNumber) {

this.passportNumber = passportNumber;

}

public String getFullName() {

return fullName;

}

public void setFullName(String fullName) {

this.fullName = fullName;

}

public String getNumber() {

return number;

}

public void setNumber(String number) {

this.number = number;

}

}

import java.io.File;

import java.io.FileNotFoundException;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class AttendeeController {

private List<WorkShop> appendables = new ArrayList<>();

private List<String> namsWorkShop = new ArrayList<>();

private List<Long> ids = new ArrayList<>();

public AttendeeController() {

}

public List<String> getWorkShops() {

try {

File myObj = new File("workshop.txt");

Scanner myReader = new Scanner(myObj);

while (myReader.hasNextLine()) {

String data = myReader.nextLine();

namsWorkShop.add(data);

}

myReader.close();

} catch (FileNotFoundException e) {

e.printStackTrace();

}

return namsWorkShop;

}

public List<WorkShop> showAllAppendables(){

return appendables;

}

public void addAttendee(WorkShop workShop) {

int flag = 0;

for (int i = 0; i < workShop.getAppendables().size(); i++) {

if(!ids.contains(workShop.getAppendables().get(i).getPassportNumber())){

ids.add(workShop.getAppendables().get(i).getPassportNumber());

flag = 1;

} else {

flag = 0;

break;

}

}

if (flag == 0) {

System.out.println("Pass Id Is Exist");

} else {

appendables.add(workShop);

}

}

public void removeAttendee(Long id) {

int flag = 0;

for (int i = 0; i < appendables.size(); i++) {

for (int z = 0; z < appendables.get(i).getAppendables().size(); z++) {

Attendee attendee = appendables.get(i).getAppendables().get(z);

if (attendee.getPassportNumber() == id) {

appendables.get(i).getAppendables().remove(attendee);

flag = 1;

break;

}

}

if (flag == 1) {

break;

}

}

if(flag == 0){

System.out.println("Pass Id Is Exist");

}

}

public void diplayWorkShop() {

for (int i = 0; i < appendables.size(); i++) {

System.out.println(appendables.get(i).getName());

for (int z = 0; z < appendables.get(i).getAppendables().size(); z++) {

System.out.print("attendee " + (z + 1) + " ");

System.out.print(appendables.get(i).getAppendables().get(z).getPassportNumber() + " ");

System.out.print(appendables.get(i).getAppendables().get(z).getFullName() + " ");

System.out.print(appendables.get(i).getAppendables().get(z).getNumber() + " ");

System.out.print(appendables.get(i).getAppendables().get(z).getAge() + " ");

System.out.println();

}

}

}

}

**Testing:**

public class Test {

public static void main(String[] args) {

// TODO code application logic here

WorkShop workShop1 = new WorkShop();

workShop1.setName("WorkShop 1");

WorkShop workShop2 = new WorkShop();

workShop2.setName("WorkShop 2");

Attendee attendee1 = new Attendee(1000, "Ahmed", "20", "20");

workShop1.getAppendables().add(attendee1);

Attendee attendee2 = new Attendee(1001, "Ayman", "30", "25");

workShop1.getAppendables().add(attendee2);

Attendee attendee3 = new Attendee(1002, "Ali", "40", "30");

workShop1.getAppendables().add(attendee3);

Attendee attendee4 = new Attendee(1003, "Atef", "45", "25");

workShop2.getAppendables().add(attendee4);

Attendee attendee5 = new Attendee(1004, "Samir", "35", "40");

workShop2.getAppendables().add(attendee5);

AttendeeController ac = new AttendeeController();

Attendee attendee6 = new Attendee(1005, "Samir", "35", "40");

workShop2.getAppendables().add(attendee6);

ac.addAttendee(workShop1);

ac.addAttendee(workShop2);

ac.diplayWorkShop();

long id = 1006;

ac.removeAttendee(id);

}

}

WorkShop 1

attendee 1 1000 Ahmed 20 20

attendee 2 1001 Ayman 30 25

attendee 3 1002 Ali 40 30

WorkShop 2

attendee 1 1003 Atef 45 25

attendee 2 1004 Samir 35 40

attendee 3 1005 Samir 35 40

Pass Id Is Exist

**GUI:**

import java.util.ArrayList;

import java.util.List;

import java.util.Observable;

import javafx.collections.FXCollections;

import javafx.collections.ObservableList;

import javafx.event.ActionEvent;

import javafx.fxml.FXML;

import javafx.scene.control.Button;

import javafx.scene.control.TableColumn;

import javafx.scene.control.TableView;

import javafx.scene.control.TextField;

import javafx.scene.control.cell.PropertyValueFactory;

/\*

\* 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.

\*/

/\*\*

\*

\* @author MBR

\*/

public class WorkShopController {

@FXML

private Button del;

@FXML

private TableColumn<WorkShop, String> workname;

@FXML

private TableView<WorkShop> tableWork;

@FXML

private TableColumn<Attendee, String> namePer;

@FXML

private TableView<Attendee> persons;

@FXML

private TableColumn<Attendee, String> number;

@FXML

private TableColumn<Attendee, Integer> id;

@FXML

private TableColumn<Attendee, String> age;

@FXML

private TextField namea;

@FXML

private TextField ida;

@FXML

private TextField agea;

@FXML

private TextField deID;

@FXML

private TextField numbera;

@FXML

private Button btnDo;

private List<Attendee> l = new ArrayList<>();

@FXML

void show(ActionEvent event) {

workname.setCellValueFactory(

new PropertyValueFactory<WorkShop, String>("name")

);

id.setCellValueFactory(

new PropertyValueFactory<Attendee, Integer>("passportNumber")

);

namePer.setCellValueFactory(

new PropertyValueFactory<Attendee, String>("fullName")

);

number.setCellValueFactory(

new PropertyValueFactory<Attendee, String>("number")

);

age.setCellValueFactory(

new PropertyValueFactory<Attendee, String>("age")

);

tableWork.setItems(getObs());

persons.setItems(getAtt());

tableWork.setVisible(false);

persons.setVisible(false);

ida.setVisible(false);

namea.setVisible(false);

numbera.setVisible(false);

agea.setVisible(false);

agea.setVisible(false);

btnDo.setVisible(false);

deID.setVisible(false);

btnDo.setVisible(false);

del.setVisible(false);

tableWork.setVisible(true);

persons.setVisible(true);

}

public ObservableList<WorkShop> getObs() {

AttendeeController getWorkShops = new AttendeeController();

List<String> works = getWorkShops.getWorkShops();

List<WorkShop> l = new ArrayList<>();

for (int i = 0; i < works.size(); i++) {

l.add(new WorkShop(works.get(i)));

}

ObservableList<WorkShop> w = FXCollections.observableArrayList(

l

);

return w;

}

public ObservableList<Attendee> getAtt() {

AttendeeController getWorkShops = new AttendeeController();

List<WorkShop> works = getWorkShops.showAllAppendables();

for (int i = 0; i < works.size(); i++) {

l.addAll(works.get(i).getAppendables());

}

ObservableList<Attendee> w = FXCollections.observableArrayList(

l

);

return w;

}

@FXML

void doDelete(ActionEvent event) {

int id = Integer.parseInt(deID.getText());

for(int i=0;i<l.size();i++){

if(l.get(i).getPassportNumber() == id){

l.remove(l.get(i));

}

}

}

@FXML

void add(ActionEvent event) {

tableWork.setVisible(false);

persons.setVisible(false);

ida.setVisible(false);

namea.setVisible(false);

numbera.setVisible(false);

agea.setVisible(false);

agea.setVisible(false);

btnDo.setVisible(false);

deID.setVisible(false);

btnDo.setVisible(false);

agea.setVisible(true);

btnDo.setVisible(true);

ida.setVisible(true);

namea.setVisible(true);

numbera.setVisible(true);

}

@FXML

void delete(ActionEvent event) {

tableWork.setVisible(false);

persons.setVisible(false);

ida.setVisible(false);

namea.setVisible(false);

numbera.setVisible(false);

agea.setVisible(false);

agea.setVisible(false);

btnDo.setVisible(false);

deID.setVisible(false);

btnDo.setVisible(false);

deID.setVisible(true);

del.setVisible(true);

}

@FXML

void clear(ActionEvent event) {

ida.setText("");

namea.setText("");

numbera.setText("");

agea.setText("");

deID.setText("");

}

@FXML

void initialize() {

tableWork.setVisible(false);

persons.setVisible(false);

ida.setVisible(false);

namea.setVisible(false);

numbera.setVisible(false);

agea.setVisible(false);

agea.setVisible(false);

btnDo.setVisible(false);

deID.setVisible(false);

btnDo.setVisible(false);

del.setVisible(false);

}

@FXML

void doneAdd(ActionEvent event) {

System.out.println(namea.getText());

l.add(new Attendee(Integer.parseInt(ida.getText()), namea.getText(), numbera.getText(), agea.getText()));

}

//

}

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

<?import javafx.scene.text.\*?>

<?import java.lang.\*?>

<?import java.util.\*?>

<?import javafx.scene.\*?>

<?import javafx.scene.control.\*?>

<?import javafx.scene.layout.\*?>

<AnchorPane id="AnchorPane" prefHeight="566.0" prefWidth="614.0" style="-fx-background-color: brown;" xmlns="http://javafx.com/javafx/8" xmlns:fx="http://javafx.com/fxml/1" fx:controller="WorkShopController">

<children>

<Button layoutX="158.0" layoutY="25.0" mnemonicParsing="false" onAction="#show" prefHeight="65.0" prefWidth="298.0" style="-fx-background-color: green;" text="Show WorkShop" textFill="WHITE">

<font>

<Font name="System Bold" size="30.0" />

</font>

</Button>

<Button layoutX="14.0" layoutY="121.0" mnemonicParsing="false" onAction="#add" prefHeight="65.0" prefWidth="194.0" style="-fx-background-color: green;" text="Add" textFill="WHITE">

<font>

<Font name="System Bold" size="30.0" />

</font>

</Button>

<Button layoutX="433.0" layoutY="121.0" mnemonicParsing="false" onAction="#delete" prefHeight="65.0" prefWidth="167.0" style="-fx-background-color: green;" text="Delete" textFill="WHITE">

<font>

<Font name="System Bold" size="30.0" />

</font>

</Button>

<Button layoutX="232.0" layoutY="121.0" mnemonicParsing="false" onAction="#clear" prefHeight="65.0" prefWidth="177.0" style="-fx-background-color: green;" text="Clear" textFill="WHITE">

<font>

<Font name="System Bold" size="30.0" />

</font>

</Button>

<TableView fx:id="tableWork" layoutX="14.0" layoutY="210.0" prefHeight="297.0" prefWidth="167.0" visible="false">

<columns>

<TableColumn fx:id="workname" prefWidth="168.0" text="WorkShop Name" />

</columns>

</TableView>

<TableView fx:id="persons" layoutX="216.0" layoutY="210.0" prefHeight="297.0" prefWidth="387.0" visible="false">

<columns>

<TableColumn fx:id="id" prefWidth="108.0" text="national id" />

<TableColumn fx:id="namePer" prefWidth="73.0" text="name" />

<TableColumn fx:id="number" prefWidth="91.0" text="number" />

<TableColumn fx:id="age" prefWidth="98.0" text="Age" />

</columns>

</TableView>

<TextField fx:id="namea" layoutX="382.0" layoutY="243.0" prefHeight="65.0" prefWidth="194.0" promptText="Name" visible="false" />

<TextField fx:id="ida" layoutX="84.0" layoutY="243.0" prefHeight="65.0" prefWidth="194.0" promptText="id" visible="false" />

<TextField fx:id="numbera" layoutX="84.0" layoutY="342.0" prefHeight="71.0" prefWidth="194.0" promptText="number" visible="false" />

<TextField fx:id="agea" layoutX="382.0" layoutY="342.0" prefHeight="71.0" prefWidth="194.0" promptText="age" visible="false" />

<Button fx:id="btnDo" layoutX="208.0" layoutY="449.0" mnemonicParsing="false" onAction="#doneAdd" prefHeight="96.0" prefWidth="233.0" style="-fx-background-color: green;" text="Done" textFill="WHITE" visible="false">

<font>

<Font name="System Bold" size="30.0" />

</font>

</Button>

<Button fx:id="del" layoutX="216.0" layoutY="449.0" mnemonicParsing="false" onAction="#doDelete" prefHeight="65.0" prefWidth="167.0" style="-fx-background-color: green;" text="Do" textFill="WHITE">

<font>

<Font name="System Bold" size="30.0" />

</font>

</Button>

<TextField fx:id="deID" layoutX="136.0" layoutY="261.0" prefHeight="81.0" prefWidth="327.0" promptText="ID" />

</children>

</AnchorPane>

