**COMPS203F: Assignment Answer Book (2021)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name:** | **Jiawei Wang** | **OUID *(8-digit)*:** | **S1239587** |

*Instructor’s Use Only*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Q1** | **Q2** | **Q3** | **Q4** | **Total** |
|  |  |  |  |  |

**Question 1**

(a)

(i)

// DialogBox.java

import javax.swing.\*;

public class DialogBox {

// JOptionPane

JFrame f;

int SGDAmount;

boolean ST = true;

// CST means the Cancel or No status

boolean CST = true;

public DialogBox() {

// Just a simple parent component (wont be displayed)

f = new JFrame("DialogBox");

f.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}

public int inputSGDAmount() {

int SGD;

// Show the Input Dialog

String StrSGD = JOptionPane.showInputDialog(f, "Input the SGD Amount:");

try {

SGD = Integer.parseInt(StrSGD);

} catch (NumberFormatException e) {

SGD = -1;

}

// Ask to Confirm

int c = JOptionPane.showConfirmDialog(f, "The input is: " + StrSGD + ", is it corret?");

if (c == JOptionPane.YES\_OPTION) {

return SGD;

}

else if (c == JOptionPane.CLOSED\_OPTION) {

System.exit(0);

}

// if No

CST = false;

return 0;

}

}

(ii)

public void checkSGDAmount() {

while(ST) {

SGDAmount = inputSGDAmount();

if (SGDAmount > 0) {

ST = false;

}

else if (CST && SGDAmount <= 0) {

JOptionPane.showMessageDialog(f, "The SGD amount should be greater than 0");

}

}

System.exit(0);

}

(iii)

public static void main(String[] args) {

DialogBox DB = new DialogBox();

DB.checkSGDAmount();

}

(b)

(i)

// Conversion.java

public class Conversion extends JFrame implements ActionListener {

GridBagConstraints gbc = new GridBagConstraints();

JTextField InputSGD = new JTextField(20);

JLabel OutputHKD = new JLabel(" ");

JButton Calculate = new JButton("Calculate");

public Conversion() {

setLayout(new GridBagLayout());

Calculate.addActionListener(this);

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.gridx = 0;

gbc.gridy = 0;

add(InputSGD, gbc);

gbc.gridx = 1;

gbc.gridy = 0;

add(OutputHKD, gbc);

gbc.gridx = 0;

gbc.gridy = 2;

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.gridwidth = 2;

add(Calculate, gbc);

pack();

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

setVisible(true);

}

}

(ii)

public void actionPerformed(ActionEvent ae) {

double SGD = Double.parseDouble(InputSGD.getText());

double HKD = SGD \* 5.7;

String OHKD = String.valueOf(HKD);

OutputHKD.setText(OHKD);

}

(iii)

public static void main(String[] args) {

Conversion CV = new Conversion();

}

**Question 2**

(a)

// Router.java

import java.util.\*;

public class Router {

private String brand;

private String modelNumber;

private double price;

public Router(String b, String m, double p) {

brand = b;

modelNumber = m;

price = p;

}

public String getBrand() {

return this.brand;

}

public String getModelNumber() {

return this.modelNumber;

}

public double getPrice() {

return this.price;

}

@Override

public String toString() {

return "brand: " + this.brand + ", model number: " + this.modelNumber + ", price: " + this.price;

}

}

(b)

// ComputerShop.java

import java.util.HashMap;

public class ComputerShop {

HashMap<String, Router> routerMap = new HashMap<String, Router>();

public void addRouter(Router oneRouter) {

String key = oneRouter.getBrand() + ": " + oneRouter.getModelNumber();

routerMap.put(key, oneRouter);

}

}

(c)

// TestComputerShop.java

public class TestComputerShop {

public static void main(String[] args) {

ComputerShop CS = new ComputerShop();

CS.addRouter(new Router("Linksys", "RVS4000", 1080));

CS.addRouter(new Router("Planet", "VRT-311S", 510));

}

}

(d)

// ComputerShop.java

public void showRouter() {

for(String k : routerMap.keySet()) {

System.out.println(routerMap.get(k));

}

}

// TestComputerShop.java

aShop.showRouter();

(e)

// ComputerShop.java

public Set<String> modelNumberSet() {

Set<String> returnSet = new HashSet<String>();

for (Router r : routerMap.values()) {

returnSet.add(r.getModelNumber());

}

return returnSet;

}

// TestComputerShop.java

Set<String> modelNumber = aShop.modelNumberSet();

for (String m : modelNumber) {

System.out.println(m);

}

(f)

// ComputerShop.java

public List<Double> priceList() {

List<Double> returnList = new ArrayList<Double>();

for (Router r : routerMap.values()) {

returnList.add(r.getPrice());

}

return returnList;

}

// TestComputerShop.java

List<Double> price = aShop.priceList();

for (Double p : price) {

System.out.println(p);

}

**Question 3**

(a)

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*; // Resolve class ActionListener

public class HexEditor extends JFrame {

private JMenuBar menuBar = new JMenuBar();

private JMenu File = new JMenu("File");

private JMenuItem Load = new JMenuItem("Load");

private JTextArea TextArea = new JTextArea(5, 10);

public HexEditor() {

setJMenuBar(menuBar);

menuBar.add(File);

File.add(Load);

setSize(5, 10);

add(TextArea);

pack();

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

}

}

(b)

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*; // Resolve class ActionListener

public class TestHexEditor {

public static void main(String[] args) {

HexEditor anEditor = new HexEditor();

anEditor.setVisible(true);

}

}

(c)

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*; // Resolve class ActionListener

import java.io.\*;

public class HexEditor extends JFrame implements ActionListener {

private JMenuBar menuBar = new JMenuBar();

private JMenu File = new JMenu("File");

private JMenuItem Load = new JMenuItem("Load");

private JTextArea TextArea = new JTextArea(5, 10);

public HexEditor() {

setJMenuBar(menuBar);

menuBar.add(File);

Load.addActionListener(this);

File.add(Load);

setSize(5, 10);

add(TextArea);

pack();

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

}

public void actionPerformed(ActionEvent ae) {

Object Source = ae.getSource();

if (Source == Load) {

String File = JOptionPane.showInputDialog("Please input the file name: ");

try {

FileInputStream in = new FileInputStream(File);

FileOutputStream out = new FileOutputStream(File);

} catch (IOException e) {

System.out.println(e.getMessage());

}

}

}

}

(d)

// HexEditor.java

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*; // Resolve class ActionListener

import java.io.\*;

public class HexEditor extends JFrame implements ActionListener {

private JMenuBar menuBar = new JMenuBar();

private JMenu File = new JMenu("File");

private JMenuItem Load = new JMenuItem("Load");

private JPanel PanelTextArea = new JPanel();

private JTextArea TextArea = new JTextArea(5, 10);

private JLabel Blank = new JLabel();

private JPanel PanelDisplayArea = new JPanel();

private JTextArea DisplayArea = new JTextArea(5, 20);

private JButton Update = new JButton("Update hex");

private JScrollPane spt = new JScrollPane(TextArea);

private JScrollPane spd = new JScrollPane(DisplayArea);

public HexEditor() {

setLayout(new BorderLayout());

setJMenuBar(menuBar);

File.add(Load);

menuBar.add(File);

Load.addActionListener(this);

Update.addActionListener(this);

PanelTextArea.add(spt);

PanelDisplayArea.add(spd);

add(PanelTextArea, BorderLayout.WEST);

add(Blank, BorderLayout.CENTER);

add(PanelDisplayArea, BorderLayout.EAST);

add(Update, BorderLayout.SOUTH);

pack();

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

}

public void actionPerformed(ActionEvent ae) {

Object Source = ae.getSource();

if (Source == Load) {

String File = JOptionPane.showInputDialog("Please input the file name: ");

try {

FileInputStream in = new FileInputStream(File);

FileOutputStream out = new FileOutputStream("Output.txt");

String FileContent = "";

int inputByte;

while ((inputByte = in.read()) != -1) {

FileContent += (char) inputByte;

}

TextArea.append(FileContent);

in.close();

out.close();

} catch (IOException e) {

System.out.println(e.getMessage());

}

}

if (Source == Update) {

}

}

}

(e)

// HexEditor.java

if (Source == Update) {

String FileContent1 = TextArea.getText();

for (int i = 0; i < FileContent1.length(); i++) {

char ch = FileContent1.charAt(i);

DisplayArea.append(Integer.toHexString(ch));

DisplayArea.append(" ");

}

}

(f)

// HexEditor.java

private JMenuItem Save = new JMenuItem("Save");

Save.addActionListener(this);

if (Source == Save) {

if (FileName == "") {

FileName = JOptionPane.showInputDialog("Please input the file you want to save: ");

}

try {

FileOutputStream out = new FileOutputStream(FileName);

String SaveContent = TextArea.getText();

for (int i = 0; i < SaveContent.length(); i++) {

char ch = SaveContent.charAt(i);

out.write(ch);

}

out.close();

} catch (IOException e) {

System.out.println(e.getMessage());

}

}

**Question 4**

(a)

(i)

CREATE TABLE item {

number char(10) primary key,

price decimal(8, 2)

};

(ii)

INSERT INTO item(numbebr, price) values ('Milk-01', 13.8);

(iii)

SELECT \* FROM item WHERE price > 10;

(b)