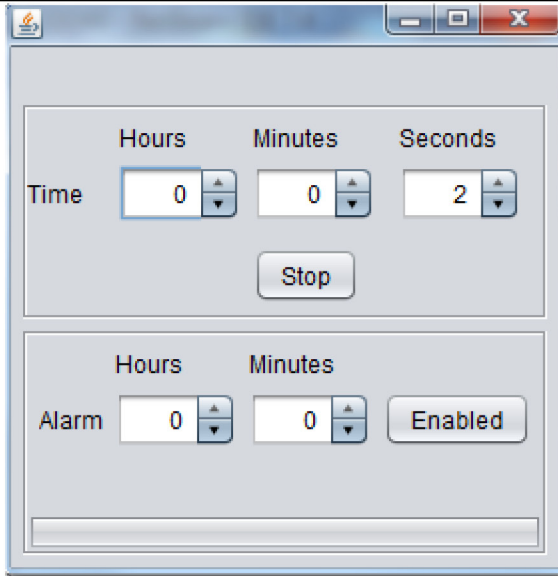


Fundamentals of Programming II Lab 12

0043



ITime	<pre> public interface ITime { int getMinutes(); int getHours(); void setHours(int h); void setMinutes(int m); long convert2Minutes(); } </pre>
-------	---

Clock	<pre> public class CClock implements ITime{ public int s,m,h; CClock() { s=0; m=0; h=0; } CClock(int h, int m, int s) { this.h=h%24; this.m=m%60; this.s=s%60; } public int getMinutes() { return m; } } </pre>
-------	---

```

    }
    public int getHours()
    {
        return h;
    }
    public void setHours(int h)
    {
        this.h=h;
    }
    public void setMinutes(int m)
    {
        this.m=m;
    }
    public long convert2Minutes()
    {
        long M;
        M=(h*60*60)+(m);
        return M;
    }
    void tick()
    {
        s++;
        m=m+(s/60);
        s=s%60;
        h=h+(m/60);
        m=m%60;
        h=h%24;
    }
    public int getSeconds()
    {
        return s;
    }
    public void setSeconds(int s)
    {
        this.s=s;
    }

```

		<pre> Boolean compareToAlarm(CAlarm ca) { long M1,M2; M1=ca.convert2Minutes(); M2=this.convert2Minutes(); if(M1==M2) return true; else return false; } </pre>
	CAalarm	<pre> public class CAlarm implements ITime{ public int m,h; CAlarm() { m=0; h=0; } CAlarm(int h, int m) { this.h=h%24; this.m=m%60; } public int getMinutes() { return m; } public int getHours() { return h; } public void setHours(int h) { this.h=h; } public void setMinutes(int m) { this.m=m; } } </pre>

$$\left. \begin{array}{l} \} \\ \rho \\ \{ \end{array} \right\}$$

```
public interface IBoard {
    void playPlayer1(int d);
    void playPlayer2(int d);
    int getPlayer1();
}
```

	<pre> int getPlayer2(); void setPlayer1(int p); void setPlayer2(int p); void checkExpectations() ; } </pre>
Board	<pre> public class Board implements IBoard { int player1,player2; Board() { player1 = 0; player2 = 0; } public void playPlayer1(int d) { player1 = (player1+d); if(player1>99) player1=99; } public void playPlayer2(int d) { player2 = (player2+d); if(player2>99) player2=99; } public int getPlayer1() { return player1; } public int getPlayer2() { return player2; } public void setPlayer1(int p) { player1=p; } public void setPlayer2(int p) { player2=p; } } </pre>

```
public void checkExpectations() {
    if (player1 == 7) {
        player1=(30);
    }
    if (player2 == 7) {
        player2=(30);
    }
    if (player1 == 14) {
        player1=(96);
    }
    if (player2 == 14) {
        player2=(96);
    }
    if (player1 == 41) {
        player1=(80);
    }
    if (player2 == 41) {
        player2=(80);
    }
    if (player1 == 65) {
        player1=(86);
    }
    if (player2 == 65) {
        player2=(86);
    }
    if (player1 == 23) {
        player1=(0);
    }
    if (player2 == 23) {
        player2=(0);
    }
    if (player1 == 54) {
        player1=(12);
    }
    if (player2 == 54) {
        player2=(12);
    }
}
```

		<pre> } if (player1 == 70) { player1=(28); } if (player2 == 70) { player2=(28); } if (player1 == 87) { player1=(66); } if (player2 == 87) { player2=(66); } if (player1 == 98) { player1=(5); } if (player2 == 98) { player2=(5); } } } </pre>
	ADice	<pre> abstract class ADice { int rank; } </pre>
	Dice	<pre> public class Dice extends ADice{ void genRank() { rank=(((int)(Math.random()*10))%6)+1; } int getRank() { return rank; } } </pre>

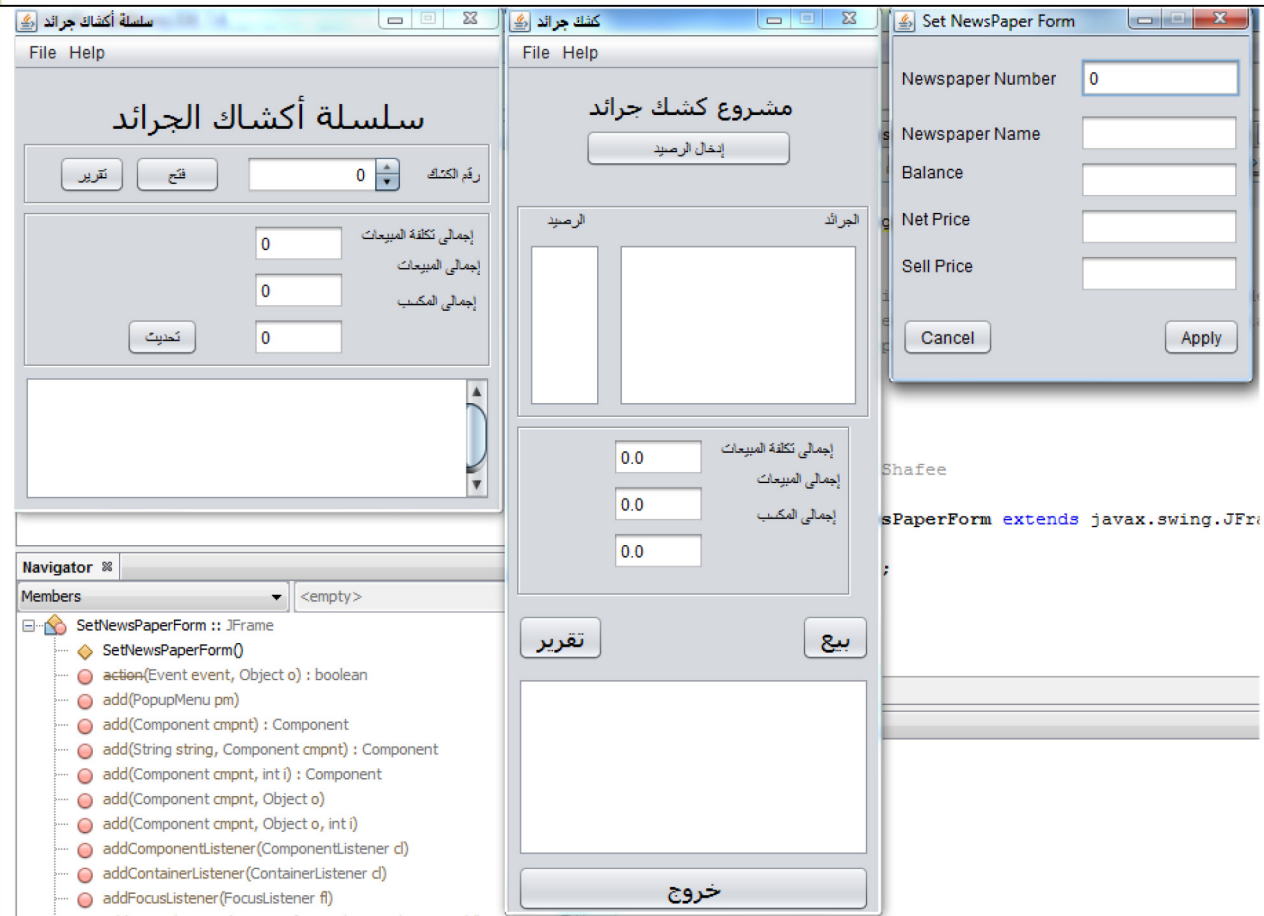
0045	ITime	<pre> public interface ITime { int getMinutes(); int getHours(); void setHours(int h); void setMinutes(int m); long convert2Minutes(); } </pre>
	ATime	<pre> abstract public class ATime implements ITime{ public int m,h; } </pre>
	CAalarm	<pre> public class CAalarm extends ATime{ CAalarm() { m=0; h=0; } CAalarm(int h, int m) { this.h=h%24; this.m=m%60; } public int getMinutes() { return m; } public int getHours() { return h; } public void setHours(int h) { this.h=h; } public void setMinutes(int m) { </pre>

		<pre> this.m=m; } public long convert2Minutes() { long M; M=(h*60*60)+(m); return M; } } </pre>
	CClock	<pre> public class CClock extends ATime{ public int s; CClock() { s=0; m=0; h=0; } CClock(int h, int m, int s) { this.h=h%24; this.m=m%60; this.s=s%60; } public int getMinutes() { return m; } public int getHours() { return h; } public void setHours(int h) { this.h=h; } public void setMinutes(int m) </pre>

```

{
    this.m=m;
}
public long convert2Minutes()
{
    long M;
    M=(h*60*60)+(m);
    return M;
}
void tick()
{
    s++;
    m=m+(s/60);
    s=s%60;
    h=h+(m/60);
    m=m%60;
    h=h%24;
}
int getSeconds()
{
    return s;
}
void setSeconds(int s)
{
    this.s=s;
}
Boolean compareToAlarm(CAlarm ca)
{
    long M1,M2;
    M1=ca.convert2Minutes();
    M2=this.convert2Minutes();
    if(M1==M2) return true;
    else return false;
}
}

```



NewsPepersBooth
Series

public class NewsPepersBoothSeries implements
Serializable {

```

    double totalCostInHand = 0;
    double totalSellInHand = 0;
    double totalProfit = 0;
    int boothes=0;
    NewsPapersBooth[] npb = new
    NewsPapersBooth[10];
    NewsPepersBoothSeries(int boothes) {
        this.boothes=boothes;
        for (int n = 0; n < boothes; n++) {
            npb[n] = new NewsPapersBooth(100);
        }
    }
    double getTotalCostInHand() {
        totalCostInHand=0;
        for(int n=0;n<boothes;n++)
    
```

	<pre> totalCostInHand=totalCostInHand+npb[n].getCostInHand(); return totalCostInHand; } double getTotalSellInHand() { totalSellInHand=0; for(int n=0;n<booths;n++) totalSellInHand=totalSellInHand+npb[n].getSellInHand(); return totalSellInHand; } double gettotalProfit() { totalProfit=0; for(int n=0;n<booths;n++) totalProfit=totalProfit+npb[n].getProfit(); return totalProfit; } } </pre>
NewsPapersBoothForm	
Form declaration	<pre> final int totalItems=100; int curretnBooth=0; NewsPapersBooth npb = new NewsPapersBooth(totalItems); SetNewsPaperForm snpf = new SetNewsPaperForm(); String press="-"; </pre>
خروج	<pre> press="exit"; this.setVisible(false); </pre>
formWindowActivated	<pre> if (snpf.press.equalsIgnoreCase("apply")) { this.setEnabled(true); npb.addNewsPaper(snpf.name, Integer.parseInt(snpf.balance), Double.parseDouble(snpf.netPrice), </pre>

		<pre> Double.parseDouble(snpf.sellPrice)); DefaultListModel dlm1 = new DefaultListModel(); DefaultListModel dlm2 = new DefaultListModel(); for (int n = 0; n < npb.items; n++) { dlm1.addElement(npb.np[n].getName()); } for (int n = 0; n < npb.items; n++) { dlm2.addElement(npb.np[n].getBalance()); } jList1.setModel(dlm1); jList2.setModel(dlm2); } else if (snpf.press.equalsIgnoreCase("cancel")) { this.setEnabled(true); } else if (snpf.press.equalsIgnoreCase("1")) { snpf.setVisible(true); } } snpf.press = "0"; </pre>
		OO49
	Form declaration	<pre> NewsPapersBoothForm npbf=new NewsPapersBoothForm(); NewsPepersBoothSeries npbs=new NewsPepersBoothSeries(10); </pre>
	فتح	<pre> npbf.press=""; npbf.curretnBooth=Integer.parseInt(jSpinner1.getValu e().toString()); npbf.npb=npbs.npb[Integer.parseInt(jSpinner1.getValu e().toString())]; npbf.update(); </pre>

		<pre> npbf.setVisible(true); this.setEnabled(false); </pre>
	تقرير	<pre> jTextArea1.setText(""); for(int n=0;n<npbs.booths;n++) { jTextArea1.append("***** Booth no : "+n+"*****\n"); for(int m=0;m<npbs.npb[n].items;m++) { jTextArea1.append(npbs.npb[m].getNewsPaperName(n)+ " : " + npbs.npb[m].getNewsPeperBalance(n) + "\n"); } jTextArea1.append(jLabel8.getText() + " : " + npbs.npb[n].getCostInHand() + "\n"); jTextArea1.append(jLabel9.getText() + " : " + npbs.npb[n].getSellInHand() + "\n"); jTextArea1.append(jLabel10.getText() + " : " + npbs.npb[n].getProfit() + "\n"); } jTextArea1.append("***** *****\n"); jTextArea1.append(jLabel8.getText() + " : " + npbs.getTotalCostInHand() + "\n"); jTextArea1.append(jLabel9.getText() + " : " + npbs.getTotalSellInHand() + "\n"); jTextArea1.append(jLabel10.getText() + " : " + npbs.gettotalProfit()+ "\n"); </pre>
	تحديث	<pre> jTextField1.setText(Double.toString(npbs.getTotalSellIn Hand())); jTextField2.setText(Double.toString(npbs.getTotalCostI nHand())); </pre>

		<pre> jTextField3.setText(Double.toString(npbs.gettotalProfit())); </pre>
	formWindowActivated	<pre> if(npbf.press.equalsIgnoreCase("exit")) { this.setEnabled(true); npbs.npb[Integer.parseInt(jSpinner1.getValue().toString())]=npbf.npb; jButton2ActionPerformed(null); } else if(npbf.press.equalsIgnoreCase("1")) { npbf.setVisible(true); } npbf.press="0"; </pre>
	Load	<pre> if (jFileChooser1.showOpenDialog(this) == 0) { try { FileInputStream fis = new FileInputStream(jFileChooser1.getSelectedFile().toString()); ObjectInputStream ois = new ObjectInputStream(fis); npbs = (NewsPepersBoothSeries) ois.readObject(); ois.close(); fis.close(); jButton2ActionPerformed(null); } catch (Exception e) { JOptionPane.showMessageDialog(this, e.getMessage(), "About", JOptionPane.ERROR_MESSAGE); } } </pre>

	save	<pre> if (jFileChooser1.showSaveDialog(this) == 0) { try { FileOutputStream fos = new FileOutputStream(jFileChooser1.getSelectedFile().toStr ing()); ObjectOutputStream oos = new ObjectOutputStream(fos); oos.writeObject(npbs); oos.close(); fos.close(); } catch (Exception e) { JOptionPane.showMessageDialog(this, e.getMessage(), "About", JOptionPane.ERROR_MESSAGE); } } </pre>
--	------	---