**Assignment Q1(1203)**

#include <iostream>

using namespace std;

class employee

{

public:

    string name;

    int age, salary, code;

    int code2 = 0;

public:

    void addemployee()

    {

        cout << "Enter your Name :";

        cin >> name;

        cout << "Enter your age,Salary and code";

        cin >> age >> salary >> code;

    }

    void display()

    {

        cout << "Name : " << name << endl;

        cout << "Salary : " << salary << endl;

        cout << "age : " << age << endl;

    }

};

class manager : public employee

{

public:

    employee \*ref[10];

    int k = 0;

public:

    void addrefemp(employee m[], int s, int sup)

    {

        cout << "Which employee you want to supervise?\nEnter Employee code : ";

        int c;

        cin >> c;

        for (int i = 1; i <= s; i++)

        {

            if (c == m[i].code)

            {

                k = k + 1;

                m[i].code2 = sup;

                ref[k] = &m[i];

                break;

            }

        }

    }

    void delrefemp()

    {

        cout << "Which reference you want to delete?\nEnter Employee code : ";

        int c;

        cin >> c;

        for (int i = 1; i <= k; i++)

        {

            if (c == ref[i]->code)

            {

                ref[i]->code2 = 0;

                ref[i] = NULL;

                break;

            }

        }

    }

    void display()

    {

        employee::display();

        if (k != 0)

        {

            cout << "References :" << endl;

            for (int i = 1; i <= k; i++)

            {

                if (ref[i] != NULL)

                    cout << i << " : " << ref[i]->name << endl;

            }

        }

    }

};

void menu()

{

    cout << "Employee Management System\n1. add manager\n2. add employee\n3. display manager\n4. display employee\n5. add employref to manager\n6. Delete reference\n7.exit" << endl;

}

int main()

{

    int m = 0, n = 0, n2 = 0;

    employee person[20];

    manager person2[20];

    while (m != 7)

    {

        menu();

        cin >> m;

        switch (m)

        {

        case 1:

            n2 += 1;

            cout << "Manager : " << n2 << endl;

            person2[n2].addemployee();

            break;

        case 2:

            n += 1;

            cout << "Employee : " << n << endl;

            person[n].addemployee();

            break;

        case 3:

            int c;

            cout << "Which manager code :";

            cin >> c;

            for (int i = 1; i <= n2; i++)

            {

                if (c == person2[i].code)

                {

                    person2[i].display();

                    break;

                }

            }

            break;

        case 4:

            int c2;

            cout << "Which employee code :";

            cin >> c2;

            for (int i = 1; i <= n; i++)

            {

                if (c2 == person[i].code)

                {

                    person[i].display();

                    cout << "supervisor code : " << person[i].code2 << endl;

                    break;

                }

            }

            break;

        case 5:

            cout << "Which manager ?" << endl;

            int cc;

            cin >> cc;

            for (int i = 1; i <= n2; i++)

            {

                if (cc == person2[i].code)

                {

                    person2[i].addrefemp(person, n, cc);

                    break;

                }

            }

            break;

        case 6:

            cout << "Which manager ?" << endl;

            int ccw;

            cin >> ccw;

            for (int i = 1; i <= n2; i++)

            {

                if (cc == person2[i].code)

                {

                    person2[i].delrefemp();

                    break;

                }

            }

            break;

        case 7:

            break;

        }

    }

}

**Assignment Q2(1203)**

package com.tonmoy.moneyconverter;

import java.awt.Color;

import java.awt.event.ActionListener; myframe.java

import java.awt.event.ActionEvent;

import java.text.DecimalFormat;

import javax.swing.\*;

public class myframe extends JFrame implements ActionListener{

    JTextField txf =new JTextField();

    private static final DecimalFormat df = new DecimalFormat("0.00");

    JTextField txf2 =new JTextField();

    JLabel label = new JLabel("INPUT MONEY");

    JLabel label2 = new JLabel("OUTPUT MONEY");

    JLabel label3= new JLabel();

    String[] choice={"USD"};

    String[] choice2={"BDT"};

    JComboBox<String> combo = new JComboBox<>(choice);

    JComboBox<String> combo2 = new JComboBox<>(choice2);

    JButton b=new JButton("Convert");

    JButton b2=new JButton("Clear");

    myframe(){

        setSize(420,420);

        setTitle("MoneyConverter");

        setVisible(true);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        setLocation(650,250);

        setResizable(false);

        getContentPane().setBackground(new Color(255, 213, 128));

        txf.setBounds(30,150,100,30);

        txf2.setBounds(270,150,120,30);

        txf.setBackground(Color.cyan);

        txf2.setBackground(Color.cyan);

        label.setBounds(30,20,100,100);

        label2.setBounds(270,20,100,100);

        label.setForeground(Color.BLUE);

        label2.setForeground(Color.BLUE);

        combo.setBounds(30,90,80,30);

        combo.setBackground(Color.GREEN);

        combo2.setBackground(Color.GREEN);

        combo2.setBounds(270,90,80,30);

        b.setBounds(140, 230, 100, 30);

        b.setBackground(Color.GREEN);

        b.setForeground(Color.red);

        b2.setBounds(140, 270, 100, 30);

        b2.setBackground(Color.GREEN);

        b2.setForeground(Color.red);

        label3.setBounds(140, 20, 150, 50);

        getContentPane().setLayout(null);

        add(txf);

        add(txf2);

        add(combo);

        add(combo2);

        add(label);

        add(b);

        add(b2);

        add(label2);

        add(label3);

        b.addActionListener(this);

        b2.addActionListener(this);

        b.setActionCommand("button");

        b2.setActionCommand("Clear");

    }

    @Override

    public void actionPerformed(ActionEvent e){

        if(e.getActionCommand().equals("button")){

        String s0 = (String) combo.getSelectedItem();

        String s = (String) combo2.getSelectedItem();

        String s1=txf.getText();

            try {

                double a=Double.parseDouble(s1);

                if(s1.isEmpty()){

                    label3.setText("Empty text-field !");

                    txf2.setText(null);

                }

                else {

                    label3.setText(null);

                    switch (s) {

                        case "BDT":

                            a = a \* 105.5085;

                            String s2 = df.format(a);

                            txf2.setText(s2 + " TAKA");

                            break;

                    }

                }

            } catch (NumberFormatException jj) {

                label3.setText("Not a valid double value !");

            }

        }

        else if(e.getActionCommand().equals("Clear")){

            txf.setText(null);

            txf2.setText(null);

        }

    }

}

package com.tonmoy.moneyconverter;

Moneyconverter.java

 \* @author tonmoy

public class Moneyconverter {

    public static void main(String[] args) {

        myframe test=new myframe();

    }

}