JAVA - OOP PROJECT

DOCUMENTATION

21SW027

BY :

APSARA ZAUR BALOCH

Subject: Object Oriented Programming.

Teacher: Sir Mohsin.

ABOUT THE PROJECT:

This project represents a digital clock with the use of GUI as it runs with each thread lasting for 1000 milliseconds. The clock displays Pakistan Standard Time Zone as it works in accordance with the time zone set on the device.

OBJECTIVE OF THIS PROJECT

The purpose of this project is to get a better understanding of the concepts like threads, aggregation, composition, and inheritance. This project contains the following concepts in a primary manner to get a clear idea of their implementation without making it too complex.

Import Statements.

CODE SNIPPET

```
import java.awt.*;
import java.text.SimpleDateFormat;
import java.util.Date;
import javax.swing.*;
```

EXPLANATION

1. java.awt.* - Provides the classes necessary to create an applet.
2. Java.text.SimpleDateFormat - Provides methods to format and parse date and time in java.
3. Java.util.Date - Represents a particular moment in time.
4. Javax.swing.*- Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.

class Aclock extends JFrame{//body}

CODE SNIPPET

```
private JLabel heading;
private JLabel ClockLabel;
private Font font=new Font("",Font.ITALIC,20);
```

EXPLANATION

Following statements are used for initializing:

1. Jheading
2. JLabel
3. Font- which is set to default, italic, size 20.

class Aclock extends JFrame{//body}

CODE SNIPPET

```
Aclock(){
   super.setTitle("A clock");
   super.setSize(400,400);
   super.setLocation(300,50);
   this.creatGUI();
   this.startClock();
   super.setVisible(true);}
public void startClock() {

 }
```

EXPLANATION

In the code above values are set and the keyword super refers to the parent class jframe but without using it the same result could have be obtained. The specifications of the interface are being set and the visibility is set to true.

class Aclock extends JFrame{//body}

CODE SNIPPET

```
public void creatGUI(){
    //GUI
    //heading=new JLabel("It's clock");
    ClockLabel= new JLabel("clock");
    //heading.setFont(font);
    ClockLabel.setFont(font);
    this.setLayout(new GridLayout(2,1));
    //this.add(heading);
    this.add(ClockLabel);
    this.setDefaultCloseOperation(0);}
```

EXPLANATION

In the code above a method, named creatGUI is created in which the Interface is defined and the values for the JLabel, Jheading etc. are set. The creatGUI method was introduced above.

class Aclock extends JFrame{//body}

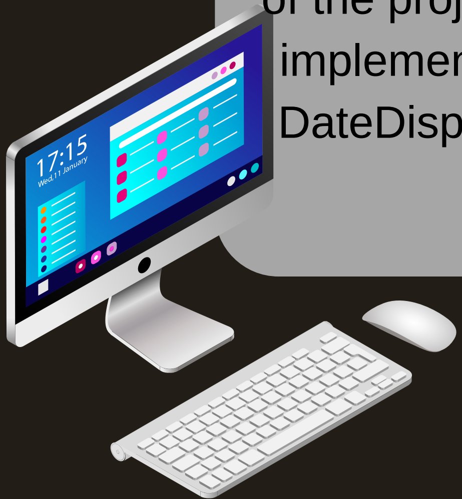
CODE SNIPPET

```
//composition
class DateDisplay{
    public void display_date() {
        try{
            while(true){
                Date d=new Date();
                SimpleDateFormat sdf=new SimpleDateFormat("hh : mm : ss a");
//hh:mm:ss are specifiers
                sdf.format(d);
                String dateTime=sdf.format(d);
                ClockLabel.setText(dateTime);
                Thread.sleep(1000);
            }

        } catch (InterruptedException e){
            e.printStackTrace();
        }
    }
```

EXPLANATION

The DateDisplay class is made in which the functionality of the project is defined and the exception handling is implemented since it is within the class "Aclock" the DateDisplay represents the concept of composition.



class Aclock extends JFrame{//body}

CODE SNIPPET

```
public void display(){
    creatGUI();
    DateDisplay dateDisplay= new DateDisplay();
    dateDisplay.display_date();

}
```

EXPLANATION

In the code above a method, named display is created in which the creatGUI method is called and the object for the class DateDisplay is made. The body of Class Aclock ends here.

class Clock extends Thread{//body}

CODE SNIPPET

```
class Clock extends Thread{

    //aggregation
    //threads implemented
    Aclock clock;
    Clock(){
        clock= new Aclock();
    }

    public void run() {
        clock.display();
    }
}

}
```

EXPLANATION

The use of thread is implemented and the concept of aggregation is defined in the code above as the Clock class initialize the object of the class Aclock,

Main Method.

CODE SNIPPET

```
public class Main {
    public static void main(String[] args) {
        System.out.println("testing...");
        Clock clock = new Clock();
        clock.start();
    }
}
```

EXPLANATION

The "Main Method" consists of the following components:

- A printing statement
- An object of the class Clock
- Start of the thread "clock".

SOME OTHER VARIATIONS (SimpleDateFormat):

Letter	Date or Time Component	Presentation	Examples
G	Era designator	Text	AD
y	Year	Year	1996; 96
Y	Week year	Year	2009; 09
M	Month in year (context sensitive)	Month	July; Jul; 07
L	Month in year (standalone form)	Month	July; Jul; 07
w	Week in year	Number	27
W	Week in month	Number	2
D	Day in year	Number	189
d	Day in month	Number	10
F	Day of week in month	Number	2
E	Day name in week	Text	Tuesday; Tue
u	Day number of week (1 = Monday, ..., 7 = Sunday)	Number	1
a	Am/pm marker	Text	PM
H	Hour in day (0-23)	Number	0
k	Hour in day (1-24)	Number	24
K	Hour in am/pm (0-11)	Number	0
h	Hour in am/pm (1-12)	Number	12
m	Minute in hour	Number	30
s	Second in minute	Number	55
S	Millisecond	Number	978
z	Time zone	General time zone	Pacific Standard Time; PST; GMT-08:00
Z	Time zone	RFC 822 time zone	-0800
X	Time zone	ISO 8601 time zone	-08; -0800; -08:00

OCTOBER 2022

THE END. 21SW027

OBJECT-ORIENTED PROGRAMMIG PROJECT.

PREPARED FOR
Sir Mohsin

PREPARED BY
Apsara Zaur Baloch