

APPLET:

* it use to web page development.

Packages:

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
```

awt → APPLET window Toolkit

APPLET ⇒ It is a class

keyword ⇒ extends
to inherit the one class
to another class

APPLET program Gr.f:

package;

public class classname extends Applet

{

statement;

}

compile → Java c classname.java

Run → applet viewer classname.java

APPLET Screen creation Gr.f:

```
<applet code="classname" height="width"
        width=""></applet>
```

Program :

```
import java.awt.*;
import java.applet.*;
public class cycle extends Applet
{
    public void start()
    {
        System.out.println("Applet Start . . . .");
    }
    public void stop()
    {
        System.out.println("Applet Stop . . . .");
    }
    public void init()
    {
        System.out.println("Applet Initialized . . . .");
    }
    public void paint(Graphics g)
    {
        System.out.println("Applet Paint . . . .");
        // repaint();
    }
    public void destroy()
    {
        System.out.println("Applet Destroy . . . .");
    }
}
```

cycle
"applet code = "first program" height = "500"
width = "500" > </applet>

Font :

it is a class

use: To set the text style and size

G.F:

```
Font objename = new Font("fontface", font.stylename,  
size);
```

fontface: Times New Roman, Arial.....

style: Font.BOLD, Font.ITALIC, Font.BOLD+Font.ITALIC

Set Font:

(G.F: Graphics object name . setFont(Font object name));

```
graphicsobjectname . SetFont(Font object name);  
(or)
```

```
graphicsobjectname . setFont(new Font ("Fontface",  
font.stylename, size));
```

```
import java.awt.*;  
import java.applet.*;  
import java.awt.event.*;  
public class fontdes extends Applet  
{  
    public void paint(Graphics g)  
    {  
        Font fo=new Font ("vinet Hand ITC", Font.BOLD, 30);  
        g.setFont(fo);  
        g.drawString ("welcom@", 100, 100);  
        g.setFont(new Font ("vivaldi", font.ITALIC, 40));  
        g.drawString ("CSC", 150, 150);  
    }  
}
```

```
<applet code="fontdes" height="500"  
width="500"></applet>
```

Color:

To change the text color and outline of shape color

Type \Rightarrow Direct color
 \Rightarrow Indirect color

object create with color

G.F:

color objectname = new Color(redValue, greenValue, blueValue);
color colorobj = new Color(color, grad);

set color:

G.F -

graphicsobjectname.setcolor[Color. colorname];

graphicsobjname.setcolor(colorobjectname); - Indirect

Set Background:

To change the background color of applet screen

G.F:

setBackground(Color.colorname);

setBackground(Colorobjectname);

set Foreground()

To change the text color

G.F:

graphicsobjectname.setForeground(Color.colorname);

Prog -

```
import java.awt.*;
import java.applet.*;
import java.event.*;
public class colorpgm extends Applet
{
    public void paint (Graphics g)
    {
        g.setColor (Color.red);
        setBackground (Color.yellow);
        g.drawString ("Welcome", 100, 100);
        //setForeground (Color.green);
        Color co = new Color (80,50,200);
        g.setColor (co);
        g.drawString ("CSC", 150, 150);
    }
}
//applet code="colorpgm" height="500"
//width="500" > </applet>
```

Shapes:

drawLine()
 drawRect()
 fillRect()
 HL → sc value change
 drawRoundRect()
 VL → s value change
 fillRoundRect()
 drawPolygon()
 fillPolygon()
 drawArc()
 fillArc()
 drawOval()
 fillOval()

drawLine();

graphics objectname.drawLine
(x1value, y1value, x2value, y2value);

```

import java.awt.*;
import java.applet.  

public class line extends Applet  

{
  public void paint(Graphics g)
  {
  }
  
```

```

    g.setColor(Color.red)
    g.drawLine(150, 350, 400, 350);
    g.setColor(Color.green)
    g.drawLine(800, 100, 200, 400);
    g.setColor(Color.blue);
    g.drawLine(400, 200, 800, 400);
  }  

  my  

  my
  
```

<applet="line" height="500"

width="500">

LW:

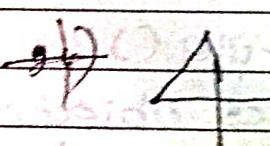
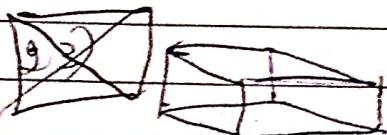
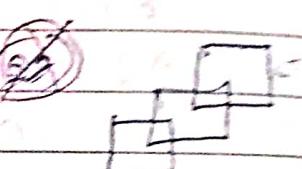
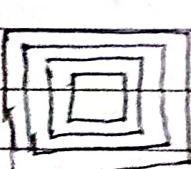
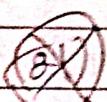
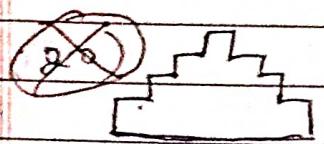
→ A → E → F → H

② I → R → L → M

③ N → X → T → V

④ W → Y → Z

⑤ V → M → F



1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16 → letters

10, 17, 18, 19, 24 → lines etc.

20, 21, 28 → red.

23 → rectangle

drawRect()

graphicsobj.methodname.drawRect(left value,
top value, width, height value)

Fill

graphics objname.fillRect(left
top value - m - m);

Ex-

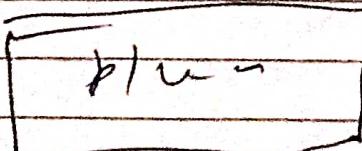
```
import java.awt.*;
import java.applet.*;
public class Rect extends Applet
{
    public void paint(Graphics g)
    {
        g.setColor(Color.red);
        setBackground(Color.yellow);
        g.drawRect(100, 250, 500, 250);
        g.setColor(Color.blue);
        g.fillRect(100, 250, 500, 250);
    }
}
```

Applet code = "rect".height = "600"

width = "600"></applet>

Output -

Output



drawRoundRect():

graphicsobjectname.drawRoundRect

(top value, left value, width value, height value,
xdiam, ydiam);

fillRoundRect()

graphicsobjectname.fillRoundRect

(top value, left value, width value, height value,
xdiam, ydiam);

```
import java.awt.*;
```

```
import java.applet.*;
```

```
public class shape extends Applet
```

```
{
```

```
g.setColor(Color.red);
g.drawRoundRect(150, 150, 300, 200, 20, 50);
g.fillRoundRect(150, 150, 300, 200, 20, 50);
```

```
//applet code = "shape" height="500" width="500"
```

```
<applet></applet>
```

~~→ x (150, 150) top-left corner~~

"(x, y) position value width, height

shape class name & file name

shape class name = "shape"

Polygon:

(25/09/2018)

To draw the Polygon Shape

GF:

`datatype x coordinate varname[] =
 value1, value2, value3, ..., value[n-1], y;`

`datatype y coordinate varname[] =
 value1, value2, value3, ..., value[n-1], y;`

Create: `new Polygon(x, y, n);`

`Graphics object name.drawPolygon`

`(X coordinate varname, Y coordinate varname`

Graphics: `(n (number of sides))`

Fill:

`Graphics object name.fillPolygon (x coordinate,
y coordinate varname, number of sides);`

```
import java.awt.*; import java.applet.*;
public class shape extends Applet
{
    public void paint(Graphics g)
    {
        int x[] = {200, 300, 300, 400, 400};
        int y[] = {400, 200, 100, 200, 400};
        g.setColor(Color.red);
        g.drawPolygon(x, y, 5);
        g.fillPolygon(x, y, 5);
    }
}
```

// applet code = "shape" height = "500"
 width = "500">></applet>

X ——————
 —————— an



drawOval():

To display the circle or Ellipse shape

G.F:

graphics objectname.drawOval(left,top,width,height);
 graphics objectname.fillOval(left,top,width,height);

`import java.awt.*;`

`import java.applet.*;`

`public class shape extends Applet {`

`public void paint(Graphics g) {`

`g.setColor(Color.red);` output
`g.drawOval(250,100,50,300);`

`g.fillOval(250,100,50,300);`

`}`

`//Applet code = "shape" height = "500"`

`width = "500" </applet>`

draw arc:

To draw Arc in our applet screen

G.F.

graphics objectname.drawArc
`(left,top,width,height,startangle,sweepangle);`

graphics objectname.fillArc

`(left,top,width,height,startangle,sweepangle);`

public class Shape extends Applet

S

public void paint(Graphics g) {

g.setColor(Color.red);

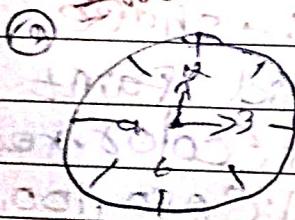
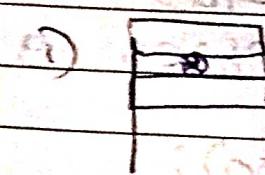
g.drawArc(200, 190, 100, 80, 90, 180);

g.fillArc(200, 190, 100, 80, 90, 180);

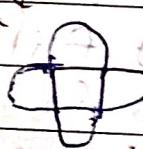
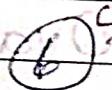
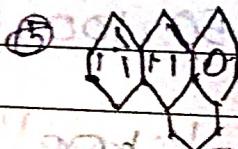
Applet code = "shape" height = "500"

width = "500" > </applet >

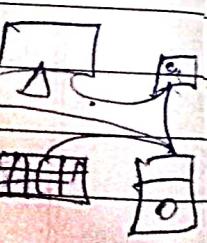
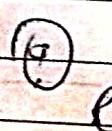
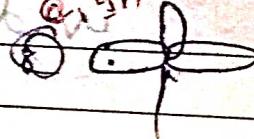
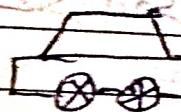
Q3 show: 619 works three choices



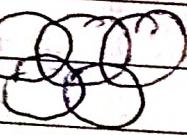
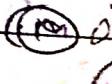
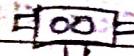
prob prob



7)



10)



Event Handling

import java.awt.event.*;

interfaces \Rightarrow it is a keyword

Event Listener is:

Action Listener

Adjustment Listener

Item Listener

key Listener

Mouse Listener

mouse Motion Listener

Text Listener

Window Listener

MouseListener:

MouseListener (Interface)

• MouseEvent (Class)

Method:

mouseEntered

mousePressed

mouseClicked

mouseReleased

mouseExited

public void mouseEntered (MouseEvent obj)

{}
y

inside of init method:

addMouseListener (this);

addMouseListener (this);

```
import java.awt.*;  
import java.applet.*;  
import java.awt.event.*;  
public class MyPanel extends Applet  
implements MouseListener
```

{

```
String str = " ";
```

```
public void init() {
```

```
addMouseListener(this);
```

}

```
public void mouseEntered(MouseEvent me) {
```

}

```
setBackground(Color.red);
```

```
str = "Mouse Entered - - -";
```

```
repaint();
```

}

```
public void mousePressed(MouseEvent me) {
```

}

```
setBackground(Color.green);
```

```
str = "Mouse Pressed - - -";
```

```
repaint();
```

}

```
public void mouseClicked(MouseEvent me) {
```

}

```
setBackground(Color.blue);
```

```
str = "Mouse Clicked - - -";
```

```
repaint();
```

}

```
public void mouseReleased(MouseEvent me) {
```

}

```
setBackground(Color.pink);
```

```
str = "Mouse Released - - -";
```

```
repaint();
```

}

```

public void mouseExited(MouseEvent me)
{
    setBackground(Color.YELLOW);
    str="Mouse Exited";
    repaint();
}
public void mousePaint(Graphics g)
{
    setForeground(Color.PINK);
    g.setFont(new Font("Vladimir Script",Font.BOLD,30));
    g.drawString(str,100,100);
}
// applet code="mbojml" height="500"
// width="500">><applet>

```

MouseMotionListener:

MouseMotionListener (interface)

Mouse Event (class)

method

mouseMoved

mouseDragged

get X()

get Y()

add MouseMotionListener(this);

MouseEvent objectname.get X();

MouseEvent objectname.get Y();

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class mm1Aym1 extends Applet implements MouseMotionListener {
    String str = " ";
    int x, y;

    public void init() {
        addMouseMotionListener(this);
    }

    public void mouseMoved(MouseEvent me) {
        setBackground("red");
        x = me.getX();
        y = me.getY();
        setBackground(Color.getRed());
        str = "Mouse moved. - " + x + "-" + y;
        repaint();
    }

    public void mouseDragged(MouseEvent me) {
        setBackground(Color.green);
        str = "Mouse Dragged " + x + "-" + y;
        repaint();
    }

    public void paint(Graphics g) {
        g.setForeground(Color.pink);
        g.setFont(new Font("Vladimir Script", Font.BOLD, 30));
        g.drawString(str, x, y);
    }
}

applet code="mm1Aym1" height="500".
width="500" border="1"
```

```
//  
     code="mmlpgml" height="500".  
    width="500"/> > /q m
```

keyListener:

keyListener (interface)
KEY Event (class)

Method:

keyPressed

keyReleased

keyTyped => getKeyChar()

To calling the getKeyChar()

KeyEvent objectname.getKeyChar();

show status)

inside of the init method:

requestFocus();
addKeyListener(this);

```
import java.awt.*;  
import java.awt.applet.*;  
import java.awt.event.*;  
public class Kel extends Applet implements  
keyListener
```

```
String str = "";  
int x = 30, y = 40;  
public void init()
```

{

Add key Listener(this);

requestFocus();

3

public void keyPressed(KeyEvent ke)

5

Show Status("key Down - - - - ");

3

public void keyReleased(KeyEvent ke)

5

Show Status("key Up - - - - ");

3

public void keyTyped(KeyEvent ke)

5

str += ke.getKeyChar();

repaint();

3

public void paint(Graphics g)

5

g.drawString(str, 20, 20);

3

<applet code="K11.java" height="500" width="500">

1. move 3 np. now set up
2. set up 3 np. now set up

3. set up 3 np. now set up

Controls:

- * Label
- * Button
- * Text box
- * Password box
- * Combo box
- * List box
- * Checkbox
- * Scroll bar
- * Option button
- * Layout
- * Frame
- * Menu Creation

Label:

Label \Rightarrow It is a class

Object create

Label objectname, -----,

Label creation (Growable)

Label objectname=new Label

"message"(or) varname,Label.e.Alignment

Alignment:

\Rightarrow LEFT

\Rightarrow RIGHT

\Rightarrow CENTER

To show the label in applet screen

add Clabel objectname);

Button

ActionListener

ActionEvent

action Performed

getActionCommand()

of : ActionEvent object name, getAction

method actionPerformed()

add ActionListener(this);

object create in constructor.

Button objectname,

Button creation :

Button objectname = new Button ("message");

to show the button control applet screen

add(buttonobjectname);

Action performed of Button

buttonobjectname.addActionListener(this);

Text box :

TextField

* It is a class

object creation:

TextField objectname; - - - . . .

control create:

objectname = new TextField("value", size);

to show applet screen:

add (TextField objectname);

→ Action Listener (interface) ↩

Action Event

actionPerformed()

getText()

setText()

getSelectedText()

GF:

TextField objectname.getText();

TextField objectname.getSelectedText();

TextField objectname.setText();

password box

object create:

TextField objectname; - - - . . .

setEchoChar

It is a property

GF:

TextField objectname.setEchoChar('symbol');

TextField objectname.addActionListener(this);

getEchoChar()

Text Area

It is a class

Object created: `WT = new TextArea()`

`TextArea objectname = ...`

OR `WT = new TextArea(Height, Width)`

`TextArea objectname = new TextArea(row value, column value);`

To show the Text area in Applet screen.

Or:

`add(TextArea objectname);`

11/7/23

```
control {
    import java.awt.*;
    import java.awt.event.*;
    public class controls extends Applet
        implements ActionListener
    {
```

Label l1,l2,l3;

Text Field t1,t2;

TextField TextArea ta;

Button b1,b2;

Public void init()

{

`l1 = new Label("Name");`

`l2 = new Label("Password");`

`l3 = new Label("Join");`

```

t1 = new JTextField("5");
t2 = new JTextField("5");
ta = newTextArea(5,15);
b1 = new Button("OK");
b2 = new Button("clear");
b1.addActionListener(this);
b2.addActionListener(this);
add(t1);
add(t2);
add(ta);
add(b1);
add(b2);
}
    
```

public void actionPerformed(ActionEvent ae)

```

String s1,s2;
s1=t1.getText();
//s1=Integer.parseInt(t1.getText()+" ");
s2=t2.getText();
if (ae.getSource()==b1)
    ta.setText(s1+"-"+s2);
else if (ae.getSource()==b2)
    {
        t1.setText("");
        t2.setText("");
        ta.setText("*");
    }
//applet code="content" height="500"
width="500">>/applet>
    
```

```

import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class addit extends Applet
    implements ActionListener
{
    Label la1, la2, la3;
    TextField t1, t2, t3;
    Button b1;
    public void init()
    {
        la1 = new Label("A value");
        la2 = new Label("B value");
        la3 = new Label("Addition");
        t1 = new TextField(15);
        t2 = new TextField(15);
        t3 = new TextField(16);
        b1 = new Button("add");
        b1.addActionListener(this);
        add(la1);
        add(t1);
        add(la2);
        add(t2);
        add(t3);
        add(b1);
    }
}

```

```

3
public void actionPerformed(ActionEvent me)
{
    int a, b;
    a = t1.getText();
    b = t2.getText();
    a = Integer.parseInt(t1.getText());
    b = Integer.parseInt(t2.getText());
    // t3.setText((a + b));
    c = a + b;
    t3.setText(c + " ");
}

```

```

import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class addit extends Applet
    implements ActionListener
    
```

Label l1, l2, l3;

TextField t1, t2, t3;

Button b1;

public void init()

l1 = new Label("A value");

l2 = new Label("B value");

l3 = new Label("Addition");

t1 = new TextField(15);

t2 = new TextField(15);

t3 = new TextField(16);

b1 = new Button("add");

b1.addActionListener(this);

add(l1);

add(t1);

add(l2);

add(t2);

add(l3);

add(t3);

add(b1);

public void actionPerformed(ActionEvent me)

{int a,b;}

a = t1.getText();

b = t2.getText();

a = Integer.parseInt(t1.getText());

b = Integer.parseInt(t2.getText());

//t3.setText((a+b));

c = a + b;

t3.setText(c + "");

Panel

object create:

Panel objectname;

panel creation:

BE: Panel objectname;

Panel objectname = new Panel();

To Show the Panel in applet screen:

• add(panelobjname);

To add the controls inside of panel:

panelobjname.add(controlobjname);

(Control obj name -> label, text, button)

import java.awt.*;

import java.applet.*;

import java.awt.event.*;

public class Panel extends Applet implements ActionListener

{

Label l1, l2, l3;

TextField t1, t2, t3;

Button b1, b2, b3, b4, b5, b6;

Panel p1, p2, p3, p4;

public void main init()

{

l1 = new Label("First value");

l2 = new Label("Second value");

l3 = new Label("Answer");

t1 = new TextField(15);

t2 = new TextField(15);

t3 = new TextField(15);

b1 = new Button("+");

b2 = new Button("-");

b3 = new Button("*");

b4 = new Button("/");

```

b9 = new JButton("%");
b6 = new JButton("clear");
P1 = new JPanel();
P2 = new JPanel();
P3 = new JPanel();
P4 = new JPanel();
P1.add(c1); P2.add(c2); P4.add(c3);
P3.add(c1); P2.add(c4); P3.add(c5); P4.add(c6);
P1.add(c1); P1.add(c2); P2.add(c2); P3.add(c3);
P3.add(c4); P3.add(c5); P4.add(c3); P4.add(c6);
add(P1); add(P2); add(P3); add(P4);
b1. addActionListener(this);
b2. addActionListener(this);
b3. addActionListener(this);
b4. addActionListener(this);
b5. addActionListener(this);
b6. addActionListener(this);
    
```

g

```

public void actionPerformed(ActionEvent ae)
{
```

```

    int a,b,cans;
    a = Integer.parseInt(t1.getText());
    b = Integer.parseInt(t2.getText());
    if (ae.getSource() == b1)
    {
```

```
        ans = a+b;
```

```
t3.setText(ans + " ");
```

g

```

else if (ae.getSource() == b2)
{ ans = a-b; t3.setText(ans + " "); }
else if (ae.getSource() == b3)
{ ans = a*b; t3.setText(ans + " "); }
```

```

case if (ae.getSource() == b4)
{
    ans = a/b; t3.setText("ans = " + ans);
}
else if (ae.getSource() == b5)
{
    ans = a % b; t3.setText("ans = " + ans);
}
else if (ae.getSource() == b6)
{
    ans = a * b;
}

t1.setText("a = " + a);
t2.setText("b = " + b);
t3.setText("ans = " + ans);

applet.setLayout(new GridLayout(3, 1));
applet.add(t1);
applet.add(t2);
applet.add(t3);

Panel panel = new Panel();
panel.setLayout(new GridLayout(2, 1));
panel.add(t1);
panel.add(t2);
applet.add(panel);
applet.add(t3);
applet.add(b1);
applet.add(b2);
applet.add(b3);
applet.add(b4);
applet.add(b5);
applet.add(b6);

panel.add(b1);
panel.add(b2);
panel.add(b3);
panel.add(b4);
panel.add(b5);
panel.add(b6);

b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
b5.addActionListener(this);
b6.addActionListener(this);
}

```

first value

second value

Ques 4) Find the answer of $\frac{1}{\sqrt{2}} \sin(\theta + 30^\circ)$

$$\begin{aligned}
 &\text{Given: } \sin(\theta + 30^\circ) \\
 &\text{Using formula: } \sin(A+B) = \sin A \cos B + \cos A \sin B \\
 &\text{So, } \sin(\theta + 30^\circ) = \sin \theta \cos 30^\circ + \cos \theta \sin 30^\circ
 \end{aligned}$$

$$\begin{aligned}
 &\sin 30^\circ = \frac{1}{2}, \cos 30^\circ = \frac{\sqrt{3}}{2} \\
 &\text{So, } \sin(\theta + 30^\circ) = \left(\frac{1}{2}\right) \sin \theta + \left(\frac{\sqrt{3}}{2}\right) \cos \theta
 \end{aligned}$$

$$\begin{aligned}
 &\text{Now, } \frac{1}{\sqrt{2}} \sin(\theta + 30^\circ) \\
 &= \frac{1}{\sqrt{2}} \left[\left(\frac{1}{2}\right) \sin \theta + \left(\frac{\sqrt{3}}{2}\right) \cos \theta \right]
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{2\sqrt{2}} \sin \theta + \frac{\sqrt{3}}{2\sqrt{2}} \cos \theta \\
 &= \frac{1}{2\sqrt{2}} \sin \theta + \frac{\sqrt{3}}{2\sqrt{2}} \cos \theta
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{2\sqrt{2}} \sin \theta + \frac{\sqrt{3}}{2\sqrt{2}} \cos \theta
 \end{aligned}$$

check box:

class → check box

object create:

checkbox objectname, ...;

checkbox creation GF:

```
checkbox objectname=new checkbox("messno",  
true/false);
```

Show applet screen:

```
add (checkbox objectname);
```

ItemListener (inteface)

ItemEvent (class)

itemStateChanged:

getState()

```
additemlistener(this);
```

getState()

GF:

```
checkbox objectname.getState();
```

```
checkbox objectname.addItemListener(this);
```

13/7/93 import java.awt.*;

```
import java.applet.*;
```

```
import java.awt.event.*;
```

public class check extends Applet

implements ItemListener

checkbox c1,c2;

TextField t1,

public void init()

c1=new checkbox("yellow");

c2=new checkbox("Red");

t1=new TextField(15);

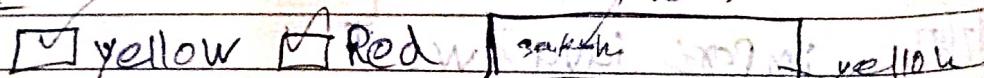
```
t1 = new TextField(15);  
c1.addItemListener(this);  
c2.addItemListener(this);  
add(c1);  
add(c2);  
add(c3);
```

public void itemStateChanged(ItemEvent e)

```
if (c1.getState())  
t1.setBackground(Color.YELLOW);  
else  
t1.setBackground(Color.PINK);  
if (c2.getState())  
t1.setForeground(Color.RED);  
else  
t1.setForeground(Color.GREEN);
```

Applet code :- check height = "50" width = "50"

Output



Radio button (or) option button

class:

checkbox

checkboxGroup

Object Create:-

checkbox objname, - - - - -

checkboxGroup objname, - - - - -

CheckboxGroup Creation : GF

checkboxGroup object name - new CheckboxGroup();

RadioButton creation : GF

checkbox object name = new Checkbox("message",
checkboxGroupObjectname, true);
GF:

add (checkbox object name);

GF:

checkbox object name.addItemListener(this);

ItemListener (Interface)

ItemEvent (class)

itemStateChanged (C)

GF:

checkboxGroupObjectname.getSelectedCheckbox().
getLabel();

getSelectedcheckbox();

getLabel();

getState();

Program:

import java.awt.*;

import java.applet.*;

import java.awt.event.*;

public class Radio extends Applet

implements ItemListener

{

Label l1; TextField t1;

checkboxGroup rg; checkbox c1, c2;

```
public void init()
```

```
{
```

```
l1 = new Label("Gender");  
t1 = new TextField(15);  
cg = new CheckboxGroup();  
c1 = new Checkbox("male", cg, false);  
c2 = new Checkbox("female", cg, false);  
c1.addItemListener(this);  
c2.addItemListener(this);  
add(c1);  
add(c2);  
add(t1);  
add(l1);  
add(cg);
```

```
}
```

Gender

```
public void itemStateChanged(ItemEvent ie)
```

```
{
```

```
if (c1.getState())
```

```
t1.setText(cg.getSelectedCheckbox().getLabel());
```

```
else
```

```
t1.setText("Female");
```

```
Swing.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

Combo box:

choice → it is a class

Object name:

Choice objectname

Combobox Creation

Choice objectname = new Choice();

To add the items of Combobox

GE:

choicebooks

```
choice objectname.add("Message");
```

GF:show:

add Echo~~choice~~objectname); WDO - 5

(A381) - 11.03.2019 (A381) - 11.03.2019

103275-1993: Name: Akbar

Dem ist einer (1993) 2010-0

Tremkistener

Item Event: (3) b(b); (11) b(b)

item State Changed (17) b (6)

→ Zn^{2+} (Zink) reagiert mit Ionen des Säureanhydrids

add ItemListener(this)

988 Selected Items (1) 341 610V 310UG

get Selected Index()

5. $\frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$ (The area of the shaded region is $\frac{1}{4}$ of the total area of the square.)

To calling the property `get`:

choice objectname.getSelectedTeam();

choice objectname.getSelectedIndex();

To get the inside of button valves

REF: [https://www.english-test.net/test/speaking/](#) 16 pages 61

ActionEvent object name . getSource () ,

```

import java.awt.*;
import java.awt.event.*;
import java.applet.*;

public class combo extends Applet implements
    ItemListener

```

S

Label l1, l2;

TextField t1;

Choice c;

public void init()

S

l1 = new Label("Available Course");

l2 = new Label("Selecting course");

t1 = new TextField(20);

c = new Choice();

c.add("HDCA"); c.add("DCA");

c.add("DMO"); c.add("NET");

c.add("JAVA"); c.add("PYTHON");

add(l1); add(c); add(l2);

add(t1);

c.addItemListener(this);

y

public void itemStateChanged(ItemEvent ie)

S

t1.setText(c.getSelectedItem() + "-" +

e.getSelectedIndex());

B) ~~selectedItem = c.getSelectedItem();~~B) ~~c.getSelectedItem() = e.getSelectedIndex();~~

// APPLET code = "combo" height = "500"

width = "500" > </applet >

out-

Available course

<input checked="" type="checkbox"/>	selected course	DMO
<input type="checkbox"/>	HDCA	
<input type="checkbox"/>	DCA	
<input type="checkbox"/>	DMO	
<input type="checkbox"/>	NET	
<input type="checkbox"/>	JAVA	
<input type="checkbox"/>	python	

List Box:class:TextAreaListobject create:

TextArea objectname, - - -

List Objectname, - - -

TextArea creation \rightarrow TextArea objectname = new TextArea("your text")List box creation - if:

List object name = new List()

To add the Listbox items - if:

List objectname.add("message");

To show the Listbox and TextArea - if:

add(TextArea objectname);

add(List objectname);

GF:

List objectname.addItemListener(this);

ItemListener

ItemEvent

itemStateChanged

getSelectedItem()

getSelectedable()

append()

getActionCommand()

(if - else)

if (listbox.getSelectedItem() == first) {

import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public void class extends APPLET
implements ItemListener

{

Label l1, l2;

TextArea ta1;

LIST ls;

public void init();

{

l1=new Label("Language");

l2=new Label("known Language");

ta1=new TextArea(3,13);

ls=new LIST();

ls.add("Tamil");

ls.add("English");

ls.add("Malayalam");

ls.add("Telugu");

ls.add("Hindi");

ls.add("French");

ls.addItem("Java");

add(l1);

add(ls);

add(l2);

add(ta1);

ls.addItemListener(this);

}

public void itemStateChanged(ItemEvent ie)

{

//ta1.setText(ls.getSelectedItem());

String str[5] = ls.getSelectedItem();
String s = " ";
int i;
for (i = 0; i < str.length; i++)
{
 s = s + str[i] + "\n";
 ta.setText(s + "");
}
1
2
3
4

11 <applet code="list" height="500"
width="500" > </applet>

Scrollbar:

17/07/23

class : Scrollbar

object create:-

Scrollbar object name;

Scrollbar creation GF:-

Scrollbar objectname = new Scrollbar();

or

add new(Scrollbar(Scrollbar.scrollBarTypes,
x,y,minimun,maximum));

Scrollbar types:-

HORIZONTAL

VERTICAL

AdjustmentListener(Interface)

Adjustment Event

adjustmentValueChanged(adjustmentValue);
addadjustmentListener(this);

getValue()

Set Maximum()

set Minimum()

Set Orientation (Scrollbar.ZYPOS);

Labwork

Lab Name

Length

Left

Right

Mid

DOB

currentdate

Age

Length

Left

Right

Mid

Current

Date

```

import java.awt.*;
import java.applet.*;
import java.awt.event.*;
class Sbox extends Applet implements
    AdjustmentListener
{
    scrollbar s1,s2,s3;
    TextField t1,t2,t3;
    Label l1,l2,l3,l4,l5,l6,l7;
    public void init()
    {
        l1=new Label("Red");
        l2=new Label("green");
        l3=new Label("blue");
        l4=new Label("Previous");
        l5=new Label("Red value");
        l6=new Label("green value");
        l7=new Label("blue value");
        t1=new TextField(15);
        t2=new TextField(15);
        t3=new TextField(15);
        s1=new Scrollbar();
        s1.setMaximum(255);
        s1.setOrientation(Scrollbar.HORIZONTAL);
        s2=new Scrollbar();
        s2.setMaximum(255);
        s2.setOrientation(Scrollbar.HORIZONTAL);
        s3=new Scrollbar();
        s3.setMaximum(255);
        s3.setOrientation(Scrollbar.HORIZONTAL);
        s1.addAdjustmentListener(this);
        s2.addAdjustmentListener(this);
        s3.addAdjustmentListener(this);
    }
}

```

```

    add(11); add(s1);
    add(12); add(s2);
    add(13); add(s3);
    add(14);
    add(15); add(z1);
    add(16); add(z2);
    add(17); add(z3);
    my

```

public void adjustmentValueChanged
(AdjustmentEvent ae)

{

```

    t1.setText(s1.getValue() + "1");
    t2.setText(s2.getValue() + "2");
    t3.setText(s3.getValue() + "3");

```

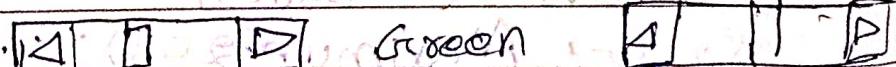
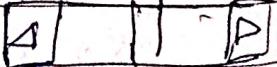
Color c = new Color(s1.getValue(), s2.getValue()
+ (s1.getValue() * 100) + s3.getValue());

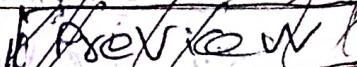
lu.setBackground(c);

my
my

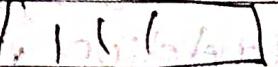
/applet code = "sbar", height = "1500"
width = "500" > </applet>

~~output~~ = "sbar" width = "500" height = "1500"

Red :  Green : 

Blue :  Effect : 

Red value 1, 251 : green value 1402

blue value 1111 : 



try catch, catch . . . finally?

```
import java.io.*;
```

```
class exa
```

```
{
```

```
public static void main(String args) throws IOException {
```

```
    DataInputStream ds = new DataInputStream(System.in);
```

```
    String na, pl; int ag;
```

```
    try {
```

```
        System.out.println("Enter ur name");
```

```
        na = ds.readLine();
```

```
        System.out.println("Enter ur Age");
```

```
        ag = Integer.parseInt(ds.readLine());
```

```
        System.out.println("Enter ur Place"); pl = ds.readLine();
```

```
        System.out.println("Given Information == == == == ");
```

```
        System.out.println("Name=" + na + "Age=" + ag + "Place=" + pl);
```

```
    } catch (NumberFormatException e) {
```

```
        System.out.println("Please enter correct Datatype Values");
```

```
        e.printStackTrace();
```

```
    }
```

```
    finally {
```

```
        System.out.println("Thank you");
```

```
}
```

Q) To set the two values and print +,-,/,
P.

WAP Q) To get wanted values and print That is

Q) To get no of students & subjects, marks &
find total, aver (array)

Thread: Even to light weight process

Thread Means Process

Multithread means Multi Process

Thread → It is a class

* series of execued statements

* nested sequence of method calls

* thread as light weight process

Thread Methods

start()

stop()

run()

runnable()

sleep()

getName()

getPriority()

try {
 import java.io.*;
} catch (...) class wanted

8

```
public class static void main (String args[])
    throws IOException
```

S

```
    DataInputStream x = new DataInputStream (System.in);
```

```
    int a, i, b;
```

```
    try :
```

S

```
        Vector v = new Vector (2);
```

S.O.P ("Enter how many values : ");

```
a = Integer.parseInt (x.readLine());
```

```
for (i = 1; i <= a; i++)
```

S

```
    S.O.P ("Enter " + i + " value");
```

input stream 10 20 30

output stream 10 20 30

input stream 10 20 30

output stream 10 20 30

input stream 10 20 30

output stream 10 20 30

input stream 10 20 30

output stream 10 20 30

input stream 10 20 30

output stream 10 20 30

input stream 10 20 30

output stream 10 20 30

To calling the thread class:

Thread objectName = new Thread();

Thread objectName.start();

Thread objectName.stop();

GF:

public void run()

S

statements;

B: switch visit and visit();

GF: final int STATEMENT; STATEMENT

Thread.sleep(value);

milliseconds.

Programme:

class th1 extends Thread

S

Thread t;

public th1()

S

t = new Thread(this);

t.start();

S

public void run()

S

try

{

int i;

for(i=0; i<10; i++)

S

System.out.println("child" + (i+1));

Thread.sleep(250);

S

S

catch (InterruptedException ex)

{

S.O.P("Thread error :- "));

}

catch (Exception ex)

{

System.out.println(ex);

}

}

class main

{

public static void main (String args)

{

thr t = new thr();

int i;

try

{

for (i=0; i<10; i++)

{

S.O.P("main "+(i+1));

Thread.sleep(500);

}

Catch (Exception ex)

{

S.O.P(ex);

}

}

By

W3 Objectobject \Rightarrow child \Rightarrow 20object \Rightarrow Hello \Rightarrow 1030m \Rightarrow Adi \Rightarrow 30main() \Rightarrow main Thread \Rightarrow 730main() \Rightarrow main Thread \Rightarrow 500

final use

SQL

Structure Query Language

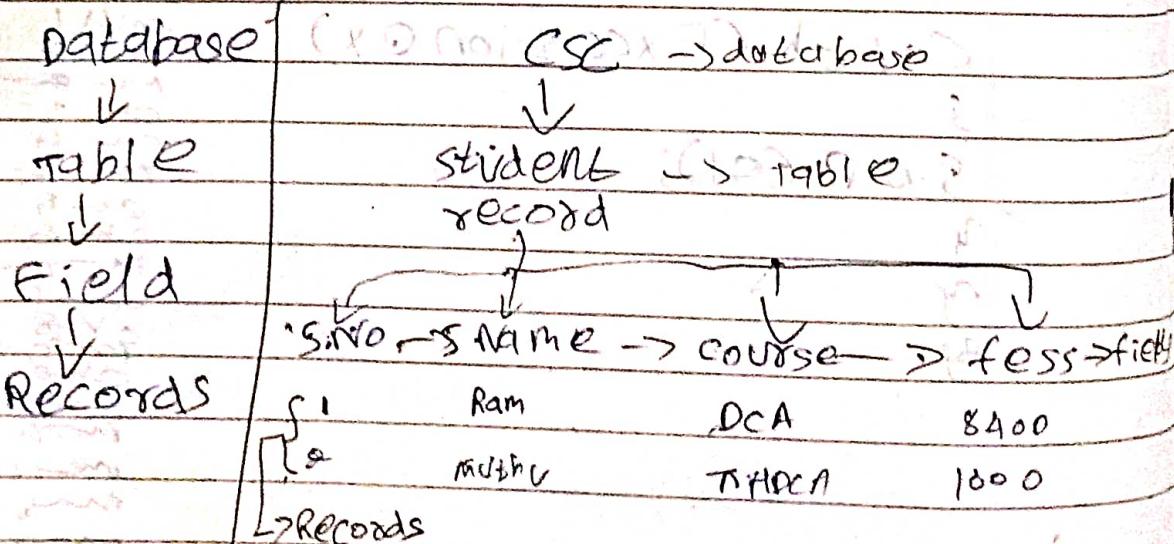
use :-

To maintain database

Types

DBMS - database management system

RDBMS - Relational database management sys

database \rightarrow collection of tabletable \rightarrow collection of fieldfield \rightarrow collection of records

servertype : Database Engine
servername : CSC-PC\SQLExpress
Authentication : SQL Server Authentication
Login : sa
password : ESS
[CONNECT]

click on connect button

how can open SQL software

way 1:

start menu → All Programs → Microsoft SQL Server 2005 Folder → Microsoft SQL Server Management Studio Express icon

New Query

Scrl → control+N

database creation:

create database database name

ex: create database demo

Scrl → F5 → enter query

go use database demo

gf: 971 → use database name

table creation: 392. schema

create table tablename (field name data type
field name data type, field name data type, ...)

How can view our table?

select * from tablename

* → All in table

How can view particular field in our table?

select fieldname from tablename

How can insert record?

In query

insert into tablename values (value,value,value,value,...)

How can insert particular records in foreign Field?

insert into tablename.(fieldname,fieldname,...)
values (value,value,...)

Update query:

update table

to set all records; \rightarrow update tablename setfieldname=values

particular records;

update tablename setfieldname=values where condition

Program:

create database condB

use condB

create table empde(emplno int, empname char(20))

(empB. stdB. stdsal int)

insert into empde values(1,'Ram',70000)

insert into empde (emplno, empname) values

(2,'Renu')

update empde . set sal=10000

update empde set sal=70000 where emplno=1

select * from empde

select empname, sal from empde.

OR

emplno	empname	sal
--------	---------	-----

1	Ram	70000
---	-----	-------

2	Renu	10000
---	------	-------

Connection

Date 19/3/2021
Page No.

Java DataBase Connection

~~sun.jdbc.odbc.JdbcOdbcDriver~~
~~sun.jdbc.odbc.JdbcOdbcDriver~~

connection = DriverManager.getConnection()

how can connect Java to SQL

→ Click Start menu

→ type control panel

→ click Administrative Tools

→ click data source (ODBC)

→ click a box will be appear

→ click Add button

→ another box will be appear

→ Select SQL Server

→ Click FINISH

→ Type database name

→ Click Next

→ a box appear

→ Click . (With SQL Server authentication
using a login id and password
enter by user)

→ Type login id and password

→ Click Next

→ Select default database name

→ Click FINISH

→ Click TEST data Source

a box appears A new shot

→ TEST COMPLETE SUCCESSFULLY

Package :

```
import java.util.*;  
import java.sql.*;
```

Class Address :

{

how can insert record in Java program.

// Prepared Statement ps=con.PrepareStatement()

Prepared Statement ps=con.PrepareStatement

("insert into Employee Values(?, ?, ?))")

ps = preparedStatement object name

con = Connection object name

Employee = SQL table name

? ? ? = how many SQL fields you give

now can Update In SQL

datatype variable = PreparedStatement object

ps.executeUpdate();

return true;

else return false;

Program :-

```

import java.io.*;
import java.util.*;
import java.sql.*;

class addrec
{
    public static void main(String args[])
        throws IOException
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection con=DriverManager.getConnection
            ("jdbc:odbc:ess","sa","sa@005");
        int id,sal;
        String na;
        DataInputStream ds=new DataInputStream
            (System.in);
        System.out.println("Enter ur Eid:");
        id=Integer.parseInt(ds.readLine());
        System.out.println("Enter ur Name:");
        na=ds.readLine();
        System.out.println("Enter ur Salary:");
        sal=Integer.parseInt(ds.readLine());
        PreparedStatement ps=con.prepareStatement
            ("insert into empde values(?, ?, ?)");
        ps.setInt(1,id);
        ps.setString(2,na);
        ps.setInt(3,sal);
        int n=ps.executeUpdate();
        System.out.println("one record added.....");
        con.close();
    }
}

```

Date :

Page:

y

catch (Exception ex)

E

S.O.P (ex);

y

new Exception("How many children?")

y

new Exception("How many children?")

new Exception("How many children?")

new Exception("How many children?")

(Exception("How many children?"))

new Exception("How many children?")

LayOut:

Types:

- 1) Flow Layout
- 2) Grid Layout
- 3) BorderLayout
- 4) Card Layout

D) Flow Layout Creation:

API: → Setter

```
setLayout(new FlowLayout(FlowLayout.Alignment,  
                           rowValue, columnValue));  
add(new ControlName("message"));
```

Alignment:

LEFT

RIGHT

CENTER

Note: Change the above alignment when
resize the applet window.

Program

```
import java.awt.*;  
import java.applet.*;  
import java.awt.event.*;  
public class flow extends Applet implements  
ActionListener
```

{

Button b1, b2, b3;

Public void init()

{

b1=new Button("LEFT")

b2=new Button("Right")

b3=new Button("CENTER") "center");

b1.addActionListener(this)

b2.a

b3.

Cm
C

add(b1); add(b2); add(b3);

```

public void actionPerformed(ActionEvent e)
{
    if(e.getSource() == b1)
        setLayout(new FlowLayout(FlowLayout.LEFT));
    else if(e.getSource() == b2)
        setLayout(new FlowLayout(FlowLayout.RIGHT));
    else if(e.getSource() == b3)
        setLayout(new FlowLayout(FlowLayout.CENTER));
}

```

//applet code = "flow" height="500" width="500" <applet>

g) grid layout:

grid layout -> 2 rows, 2 columns

Programmatic representation :-

setLayout(new GridLayout(rowValue, columnValue));

• add(new controlName("message1"));

import java.awt.*; // for window

import java.applet.*; // for window

import java.awt.event.*; // for window

```

public class grid extends Applet
{

```

Button b1[] = new Button[5];

public void init()

{

setLayout(new GridLayout(5, 2));

int i;

for(i=0; i<b1.length; i++)

b1[i] = new Button("CS"+(i+1));

add(b1[i]); add(b1[i]);

} // end of loop

//applet code = "grid" .height="500" .width="500" <applet>

Border Layout

C.R.

```
setLayout(new BorderLayout());
add(new controlname ("message"),
    BorderLayout.Alignment);
```

Alignment:

NORTH, SOUTH

EAST, WEST

CENTER

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import java.util.*;
public class border extends Applet
    implements ActionListener
```

Panel p1;

Table t1, t2, t3; // three tables with 660

TextArea ta;

Button b1[] = new Button[15];

Random rnd;

public void init() {

// creating a frame here

int i;

p1 = new Panel();

p1.setLayout(new GridLayout(15, 12));

spcl = new Random();

for (i = 0; i < b1.length; i++)

{

b1[i] = new Button("H1");

b1[i].setBackground(new Color(rnd.nextInt(255) + 122, 255), rnd.nextInt(255), rnd.nextInt(255));

```
P1.add(b1);
b1.addActionListener(this);
}
l1=new Table("North");
l2=new Table("South");("East");
l3=new Table("South");
ta=new TextArea();
setLayout(new BorderLayout());
add(P1,BorderLayout.WEST);
add(l1,BorderLayout.NORTH);
add(l2,BorderLayout.EAST);
add(l3,BorderLayout.SOUTH);
add(ta,BorderLayout.CENTER);
}
public void actionPerformed(ActionEvent ae)
{
    Button b1=(Button)ae.getSource();
    ta.setBackground(b1.getBackground());
}
//applet code="border1",height="500"
width="500"></applet>
```

CardLayout

GFR

```
setLayout(new CardLayout());  
Panel objectname = new Panel();  
add ("name", objectname);  
..  
show(this, "Paneldobiname");
```

Planned: 10/11/03, done: 10/11/03, 19%
Actual: 10/11/03, done: 10/11/03, 100%

```
import java.awt.*;  
import java.applet.*;  
import java.awt.event.*;
```

```
public class card extends Applet
```

implements ActionListener.

8

```
: Panel p1, p2, p3; JButton b1, b2;
```

```
: Checkbox bsi, bse, BSI, qse;
```

```
: CardLayout c;
```

```
: public void init()
```

```
: JPanel p1, p2, p3;
```

```
: c=new CardLayout();
```

```
: p1=new Panel();
```

```
: p1.setLayout(c);
```

```
: b1=new JButton("Basic");
```

```
: b2=new JButton("Advanced");
```

```
: b1.addActionListener(this);
```

```
: b2.addActionListener(this);
```

```
: p2=new Panel();
```

```
: p3=new Panel();
```

```
: bsi=new Checkbox("DMO");
```

```
: bse=new Checkbox("DTP");
```

```
Q1 = new checkbox("Ques1");
Q2 = new checkbox("Ques2");
R1.add(Q1);
R2.add(Q2);
P1.add(Q1);
P2.add(Q2);
P3.add(Q1);
P4.add(Q2);
P1.add(P2,"bas");
P1.add(P3,"adv");
add(P1);
add(P2);
add(P3);
add(P4);
```

public void actionPerformed(ActionEvent ae)

```
{ if(ae.getSource() == b1)
    C.show(P1,"bas");
else if(ae.getSource() == b2)
    C.show(P1,"adv");}
```

Kapplet code = "card".height="500" width="500">></applet>

Output: click basic button

Basic Advanced DMO DTP

click Advanced button

Basic Advanced JAVA .NET

Frame

it is a class

WindowListener(WindowAdapter)
WindowEvent
windowClosing
dispose()

Frame creation ex

Frame objectname = new Frame("message");

show();

frameobject.show();

setSize()

Frame objectname . setSize(width, height);

Frame objectname . setSize(400, 300);

Program

```
import java.awt.*;
```

```
import java.applet.*;
```

```
import java.awt.event.*;
```

```
class myframe extends Frame
```

```
{
```

```
myframe (String title)
```

```
{
```

```
super(title);
```

```
addWindowListener(new WindowAdapter())
```

```
{
```

```
public void windowClosing(WindowEvent e)
```

```
E
dispose();
}
}
public class frame extends Applet
implements ActionListener
{
public void init()
{
Button b1=new Button("My Create
Frame");
b1.addActionListener(this);
f.add(b1);
}
public void actionPerformed(ActionEvent
ae)
{
myFrame=new myFrame("My new window");
f.show();
f.setSize(300,500);
}
}
//applet code="frame" height="500"
width="500"></applet>
```

Menu Creation

MenuBar:

It is a class.

MenuBar object creation

MenuBar creation objectname; . .;

MenuBar creation GF:

MenuBarObjectname = newMenuBar();

To show the Menubar in Applet screen
using window method

window.setMenuBar(MenuBarObjectname);

Menu → It is a class

object creation - - -

Menu object, - - -;

Menu creation GF -

Menu Objectname = new Menu("Message");
Ex: File, Edit, - - -

GF:

MenuBarObjectname.add(MenuObjectname);

MenuItem :

To add the list of items of MENU

Ex. NEW, SAVE, OPEN, --, etc.

menuItem → It is a class

object create

MenuItem objectname = new MenuItem("message");

GF:

MenuItem objectname = new MenuItem("message");

GF:

menuobjectname.add(MenuItem objectname);

checkbox MenuItem GF

checkbox MenuItem objectname = new CheckBoxMenuItem("Message");

Ex: NEW, SAVE, --

GF:

menuobjectname.add(CheckboxMenuItem objectname);

GF:

MenuItem objectname = new MenuItem("-");

MyMenuHandler GF

MyMenuHandler objectname = new MyMenuHandler(this);

Menu Creations

Header file:

```
import javax.swing.*;
```

```
Container objectname = getContentPane();
```

```
containerObjectName.add(controlObjectname, layout);
```

FrameName \Rightarrow JFrame

Classes:

```
JFileChooser objectname = new JFileChooser();
JFileChooser objectname.showOpenDialog(this);
JFileChooser objectname.showSaveDialog(this);
```

```
FileReader objectname = new FileReader
(FileChooser objectname.getSelectedFile());
```

```
for (int i = 0; i < objectname.getNameCount(); i++)
    read(FileReader objectname, this);
```

```
FileWriter objectname = new FileWriter
```

```
(FileChooser objectname.getWriter());
```

```
textAreaName.write(FileWriter objectname);
```

Function

```
System.exit(0);
```

Cut()

COPY()

PASTE()

Select All()

Notes:

class objectname.setSize(height, width);
 class objectname.setVisible(true);

Program:

```

import java.io.*;
import java.awt.*;
import java.event.*;
import javax.swing.*;
public class notepad extends JFrame
  implements ActionListener
{
  JMenuBar mb;
  JMenu file, edit;
  JMenuItem ne, opisaj, pku, pa, sa, ex;
  JTextArea ta;
  public notepad()
  {
    mb = new JMenuBar();
    file = new JMenu("File");
    edit = new JMenu("Edit");
    ne = new JMenuItem("New", 'n');
    opisaj = new JMenuItem("Open", 'o');
    sa = new JMenuItem("Save", 's');
    cp = new JMenuItem("Copy", 'c');
    cu = new JMenuItem("Cut", 'x');
    pa = new JMenuItem("Paste", 'p');
    sa1 = new JMenuItem("SelectAll", 'a');
    ex = new JMenuItem("Exit", 'e');
    ta = new JTextArea();
  }
}
  
```

```

mb.add(File);
mb.add(Edit);
File.add(ne);
File.add(OP);
File.add(SA);
File.add(EX);
Edit.add(CP);
Edit.add(CU);
Edit.add(PA);
Edit.add(SAL);
ne.addActionListener(this);
OP.addActionListener(this);
SA.addActionListener(this);
CP.addActionListener(this);
CU.addActionListener(this);
PA.addActionListener(this);
SAL.addActionListener(this);
EX.addActionListener(this);
setJMenuBar(mb);
Container c = getContentPane();
c.setLayout(BorderLayout.CENTER);
public void actionPerformed(ActionEvent ae)
{
    if(ae.getSource() == ne)
        ta.setText("New");
    if(ae.getSource() == CP)
        ta.setText("Copy");
}

```

```
else if (ae.getSource() == OP)
{
    JFileChooser f = new JFileChooser();
    f.showOpenDialog(this);
    FileReader fl = new FileReader(
        f.getSelectedFile());
    ta.read(fl, this);
}

else if (ae.getSource() == SP)
{
    JFileChooser f = new JFileChooser();
    f.showSaveDialog(this);
    FileWriter fl = new FileWriter(
        f.getSelectedFile());
    ta.write(fl);
}

else if (ae.getSource() == CU)
{
    ta.cut();
}

else if (ae.getSource() == CP)
{
    ta.copy();
}

else if (ae.getSource() == P)
{
    ta.paste();
}

else if (ae.getSource() == SA)
{
    ta.selectAll();
}
```

