**Practical No. 30: Develop a program to draw following shapes, graphics and applets.**

1. **Cone**
2. **Cylinders**
3. **Cube**
4. **Square Inside a circle**
5. **Circle inside a square**
6. **Practical Significance:**

The graphics class of java provides methods for drawing many different types of shapes. Students will be able to use different methods of graphics programming to draw simple lines, figures of different shapes, images and text in different fonts and styles with different appearance of colors.

1. **Relevant Course Outcome:**

Develop a program using graphics and applet.

1. **Practical Outcome:**

Develop a program to draw following shapes, graphics and applets.

1. Cone
2. Cylinders
3. Cube
4. Square Inside a circle
5. Circle inside a square
6. **Minimum Theoretical Background:**

A java applet draws graphical image inside the space using the coordinate system. To draw different shapes, appropriate graphics method with proper syntax is required.

1. **Program Code:**

import java.awt.\*;

import java.applet.\*;

public class Diff\_Shapes extends Applet

{

public void paint(Graphics g)

{

/\*Cylinder\*/

g.drawString("(a).Cylinder",10,110);

g.drawOval(10,10,50,10);

g.drawOval(10,80,50,10);

g.drawLine(10,15,10,85);

g.drawLine(60,15,60,85);

/\*Cube\*/

g.drawString("(b).Cube",95,110);

g.drawRect(80,10,50,50);

g.drawRect(95,25,50,50);

g.drawLine(80,10,95,25);

g.drawLine(130,10,145,25);

g.drawLine(80,60,95,75);

g.drawLine(130,60,145,75);

/\*Squar Inside A Circle\*/

g.drawString("(c).Squar Inside A Circle",150,110);

g.drawOval(180,10,80,80);

g.drawRect(192,22,55,55);

/\*Circle Inside a Squar\*/

g.drawString("(d).Circle Inside a Squar",290,110);

g.drawRect(290,10,80,80);

g.drawOval(290,10,80,80);

}

}

/\*

<applet code="Diff\_Shapes" width=200 height=200>

</applet>

\*/

1. **Practical Related Questions:**
2. **Which of these methods is a part of Abstract Window Toolkit (AWT)?**
3. **display()**
4. **paint()**
5. **drawstring()**
6. **none of the above**

Ans. Paint

1. **Enlist the methods required to draw cone/cylinder.**

There are no predefined methods to draw a cylinder, cube and circle. Using existing methods of Graphics class like drawOval(), drawRect(), drawArc() etc., the task can be done.

1. **Explain the method with syntax to draw circle.**

To draw a circle keep the width and length the same. Used to draw an arc inside an imaginary rectangle whose upper left corner is at (x,y). The arc is drawn from the startAngle to startAngle + arcAngle and is measured in degrees. A startAngle of 0º points horizontally to the right (like the unit circle in math).

1. **Differentiate applets and applications.**

|  |  |  |
| --- | --- | --- |
| **Sr.**  **No.** | **Java Application** | **Java Applet** |
| 1 | Applications are just like a Java programs that can be execute independently without using the web browser. | Applets are small Java programs that are designed to be included with the HTML web document. They require a Java-enabled web browser for execution. |
| 2 | Application program requires a main function for its execution. | Applet does not require a main function for its execution. |
| 3 | Java application programs have the full access to the local file system and network. | Applets don’t have local disk and network access. |
| 4 | Applications can access all kinds of resources available on the system. | Applets can only access the browser specific services. They don’t have access to the local system. |
| 5 | Applications can executes the programs from the local system. | Applets cannot execute programs from the local machine. |
| 6 | An application program is needed to perform some task directly for the user. | An applet program is needed to perform small tasks or the part of it. |

1. **Exercise:**
2. **Write output of the program.**

/\*

<applet code = "SetBackgroundColorExample" width = 200 height = 200>

</applet>

\*/

import java.applet.Applet;

import java.awt.Color;

import java.awt.Graphics;

public class SetBackgroundColorExample extends Applet

{

public void paint(Graphics g)

{

/\*

\*Set background color of an applet using

\*void setBackground(Color c) method.

\*/

setBackground(Color.red);

}

}

1. **Develop a program to draw any 2 of the following shapes.**
2. **Cone**
3. **Cylinder**
4. **Cube**
5. **Cone**

import java.applet.\*;

import java.awt.\*;

public class Cone extends Applet

{

public void paint(Graphics g)

{

g.drawOval(80,280,320,100);

g.drawLine(240,50,82,320);

g.drawLine(240,50,398,320);

g.drawLine(240,330,398,330);

g.drawLine(240,50,240,330);

g.drawString("Radius",260,360);

g.drawString("Height",246,255);

g.drawString("Slant Height",340,210);

g.drawString("Cone",220,420);

}

}

/\*

<applet code="Cone.class" height=500 width=700>

</applet>

\*/

1. **Cylinder**

import java.applet.\*;

import java.awt.\*;

public class Cylinder extends Applet

{

public void paint(Graphics g)

{

g.drawString("Cylinder",80,50);

g.drawOval(50,60,100,50);

g.drawLine(50,80,50,200);

g.drawLine(150,80,150,200);

g.drawOval(50,180,100,50);

}

}

/\*

<applet code="Cylinder.class"

width=300 height=300>

</applet>

\*/

1. **Cube**

import java.applet.\*;

import java.awt.\*;

public class Cube extends Applet

{

public void paint(Graphics g)

{

g.drawString("Cube",95,110);

g.drawRect(80,10,50,50);

g.drawRect(95,25,50,50);

g.drawLine(80,10,95,25);

g.drawLine(130,10,145,25);

g.drawLine(80,60,95,75);

g.drawLine(130,60,145,75);

}

}

/\*

<applet code="Cube.class" width=300 height=300>

</applet>

\*/

1. **Develop a program to any one of the following:**
2. **Square Inside a circle**
3. **Circle inside a square**

import java.awt.\*;

import java.applet.\*;

public class Squar\_Circle extends Applet

{

public void paint(Graphics g)

{

/\*Squar Inside A Circle\*/

g.drawString("(a).Squar Inside A Circle",150,110);

g.drawOval(180,10,80,80);

g.drawRect(192,22,55,55);

/\*Circle Inside a Squar\*/

g.drawString("(b).Circle Inside a Squar",290,110);

g.drawRect(290,10,80,80);

g.drawOval(290,10,80,80);

}

}

/\*

<applet code="Squar\_Circle" width=200 height=200>

</applet>

\*/