Kapitulli 16: Grafika

Objektivat

- Te mesojme rreth metodave `paint()` dhe `repaint()`
- Perdorimi i metodes `drawString()` per te vizatuar tekste te ndryshme me ngjyra dhe lloje shkrimi.
- Vizatimi i vijave dhe formave
- Me teper rreth font-eve
- Vizatimi me grafiken 2D ne java.

Metodat `paint()` dhe `repaint()`

- Rerender
  - per te rishfaqur siperfaqen e afishimit
- Painting
  - System-triggered painting
  - Application-triggered painting
- `metoda paint()`
  - Shkruani metoden tuaj per te mbivendosur metoden default
  - Koka e metodes
    - `public void paint(Graphics g)`

- `Objekti Graphics`

  - I konfiguruar paraprakisht me vlerat e duhura per te vizatuar ne nje komponent
- `Metoda repaint()`
  - Perdoret kur dritarja duhet te update-ohet
  - Therret metoden `paint()`
  - Krijon nje objekt `Graphics`

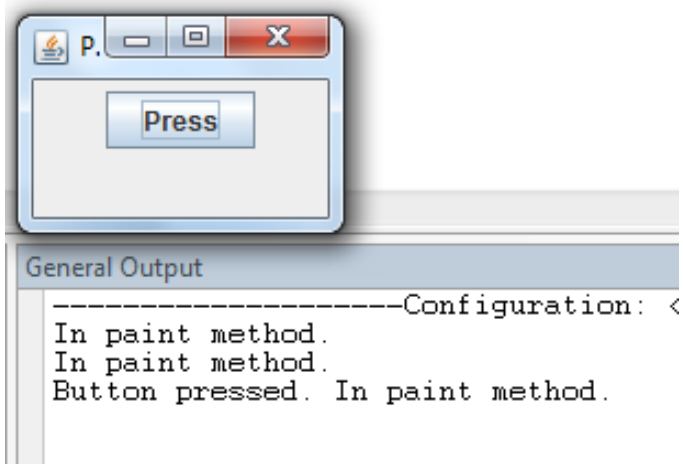
```
// Shembull
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class JDemoPaint extends JFrame implements
ActionListener
```

```
{
    JButton pressButton = new JButton("Press");
    public JDemoPaint()
    {
        setTitle("Paint Demo");
        setLayout(new FlowLayout());

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        add(pressButton);
        pressButton.addActionListener(this);
    }
    public void actionPerformed(ActionEvent e)
```

```
{
    System.out.print("Button pressed. ");
    repaint();
}
public void paint(Graphics g)
{
    super.paint(g);
    System.out.println("In paint method.");
}
public static void main(String[] args)
{
    JDemoPaint frame = new JDemoPaint();
```

```
        frame.setSize(150, 100);
        frame.setVisible(true);
    }
}
```



Perdorimi i metodes `setLocation()`

- Vendos nje komponent ne vendodhjen specifike brenda panelit te permbajtjes se nje `JFrame`
- Mund te ndryshoni pozicionimin e nje komponenti duke perdorur metoden `setLocation()`
  - `pressMe.setLocation(100, 50);`

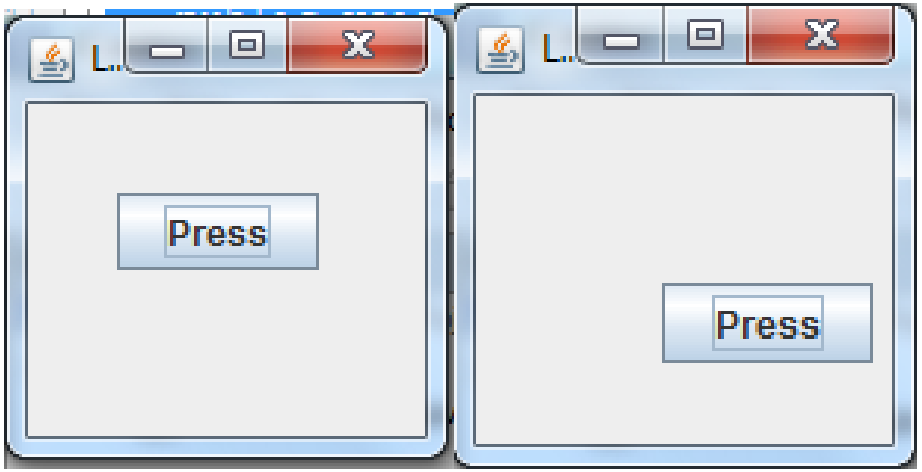
```
// Shembull
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
```

```
public class JDemoLocation extends JFrame
implements ActionListener
{
    JButton pressButton = new JButton("Press");
    int x = 0, y = 0;
    final int GAP = 30;
    public JDemoLocation()
    {
        setTitle("Location Demo");
        setLayout(new FlowLayout());

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
        add(pressButton);
        pressButton.addActionListener(this);
    }
    public void actionPerformed(ActionEvent e)
    {
        pressButton.setLocation(x, y);
        x += GAP;
        y += GAP;
    }
    public static void main(String[] args)
    {
        JDemoLocation frame = new JDemoLocation();
```

```
            frame.setSize(150, 150);
            frame.setVisible(true);
        }
}
```



Krijimi i objekteve `Graphics`

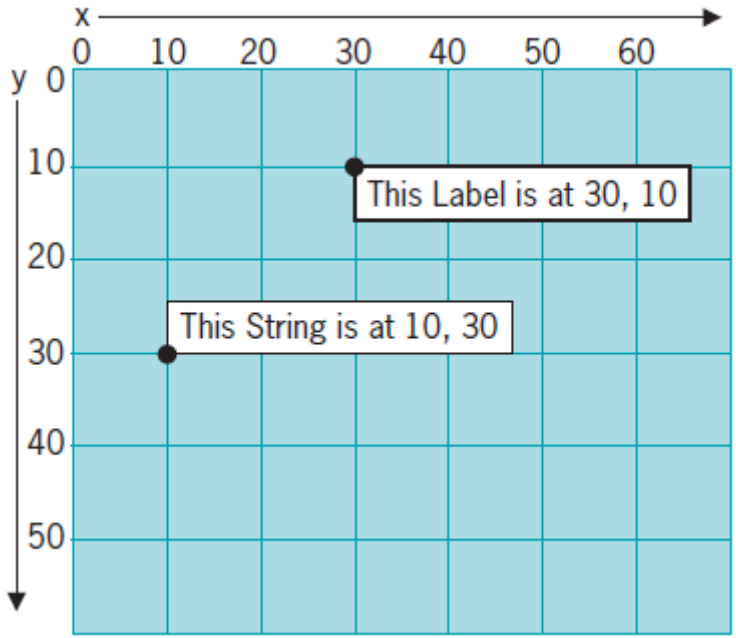
- Therrisni metoden `paint()`
  - Perdorni objektin `Graphics` te krijuar automatikisht
  - krijoni ndonje objekt `Graphics`
    - `Graphics draw = getGraphics();`

Perdorimi imetodes `drawString()`

- `Metoda drawString()`
  - Ju lejon te vizatoni nje `String` ne nje dritare `JFrame`.
  - Kerkon tre argumenta:
    - `String`

- **Koordianten x**
- **Koordianten y**

– **Eshte nje anetar i klases `Graphics`**



Perdorimi i metodave `setFont()` dhe `setColor()`

- **`Metoda setFont()`**
  - **Kerkon nje objekt `Font`**
- **Ju mund te krijoni nje objekt `Graphics` object per te perdorur nje font**
  - `somegraphicsobject.setFont(someFont);`

```
// Shembull
import javax.swing.*;
import java.awt.*;
public class JDemoFont extends JFrame
```

```
{
    Font bigFont = new Font("Serif", Font.ITALIC, 48);
    String hello = "Hello";
    public void paint(Graphics brush)
    {
        super.paint(brush);
        brush.setFont(bigFont);
        brush.drawString(hello, 10, 100);
    }
    public static void main(String[] args)
    {
        JDemoFont frame = new JDemoFont();
```

```
            frame.setSize(180, 150);
            frame.setVisible(true);
        }
}
```

Perdorimi i Ngjyrave

- `Metoda setColor()`
  - Percakton nje ngjyre `Graphics`
  - Perdorni nje nga 12 konstantet e klases `Color` si argumenta
    - `brush.setColor(Color.GREEN);`

Vizatimi i vijave dhe formave

- Java siguron disa metoda per vizatimin e nje shumellojshmeri vijash dhe formash

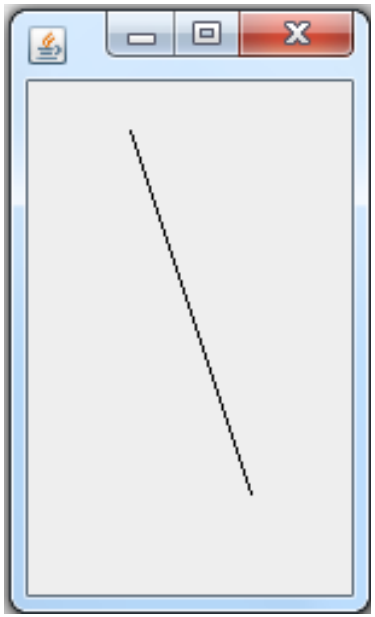
Vizatimi i vijave

- `Metoda drawLine()`
  - Vizaton nje vije te drejte midis dy pikave

- Merr 4 argumenta:
  - Koordinatat x dhe y te pikes se fillimit
  - Koordinatat x dhe y te pikes se mbarimit

```
//Shembull
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class JDemoLine extends JFrame
{
    public void paint(Graphics pen)
    {
```

```
        super.paint(pen);
        pen.drawLine(50, 50, 100, 200);
    }
    public static void main(String[] args)
    {
        JDemoLine frame = new JDemoLine();
        frame.setSize(150, 250);
        frame.setVisible(true);
    }
}
```



Vizatimi i Drejtkendeshit

- `Metoda drawRect()`
  - Vizaton vijen e jashtme te nje drejtkendeshi
- `Metoda fillRect()`
  - vizaton nje drejtkesdesh te mbushur
- Te dyja kerkojne 4 argumenta:
  - Koordinatat x dhe y te cepit lart majtas te drejtkendshit
  - gjeresine dhe gjatesine e drejtkendeshit

- `Metoda clearRect()`
  - Vizaton nje drejtkesndesh
  - Kerkon 4 argumenta:

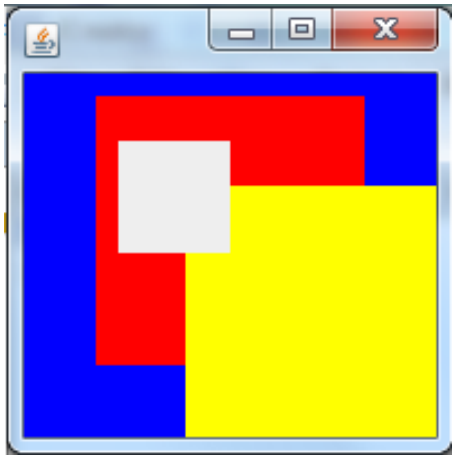
- Koordinatat x dhe y te cepit lart majtas te drejtkendshit
- gjeresine dhe gjatesine e drejtkendeshit
- shfaqet bosh ose “i fshire”

```
// Shembull

import javax.swing.*;
import java.awt.*;
import java.awt.Color;
public class JDemoRectangles extends JFrame
{
    Container con = getContentPane();
    public JDemoRectangles()
```

```
    {
        con.setBackground(Color.BLUE);
        con.setLayout(new FlowLayout());
    }    public void paint(Graphics gr)
{           super.paint(gr);            gr.setColor(Color.RED);
            gr.fillRect(40, 40, 120, 120);
            gr.setColor(Color.YELLOW);
            gr.fillRect(80, 80, 160, 160);
            gr.clearRect(50, 60, 50, 50);
      }
      public static void main(String[] args)
      {
          JDemoRectangles frame = new
```

```
    JDemoRectangles();
            frame.setSize(200, 200);
            frame.setVisible(true);
        }
    }
```



- `Metoda drawRoundRect()`
  - Krijon nje drejtkendesh me kende te rrumbullakosura
  - Kerkon 6 argumenta

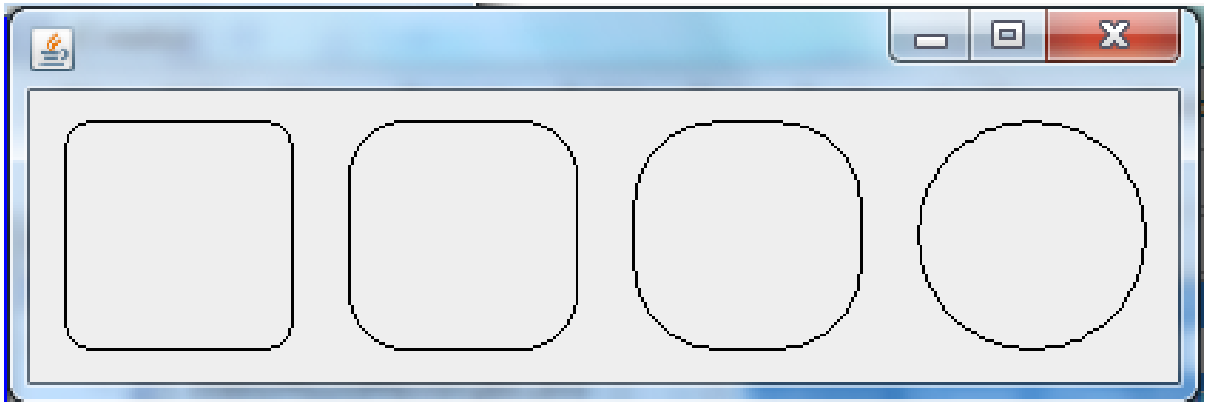
```
// Shembull

import javax.swing.*;
```

```
import java.awt.*;
public class JDemoRoundRectangles extends JFrame
{
    public void paint(Graphics gr)
    {
        super.paint(gr);
        int x = 20;
        int y = 40;
        final int WIDTH = 80, HEIGHT = 80;
        final int HORIZONTAL_GAP = 100;
        for(int arcSize = x; arcSize <= HEIGHT; arcSize +=
20)
```

```
            {
                gr.drawRoundRect(x, y, WIDTH, HEIGHT,
arcSize, arcSize);
                x += HORIZONTAL_GAP;
            }
        }
        public static void main(String[] args)
        {
            JDemoRoundRectangles frame = new
JDemoRoundRectangles();
            frame.setSize(420, 140);
            frame.setVisible(true);
```

```
    }
}
```



Krijimi i drejtkendeshave te hijezuar

- `Metoda draw3DRect()`
  - Nje variant i metodes `drawRect()`
  - Vizaton nje drejtkendesh qe shfaqet “i hijezuar” ne dy skajet.

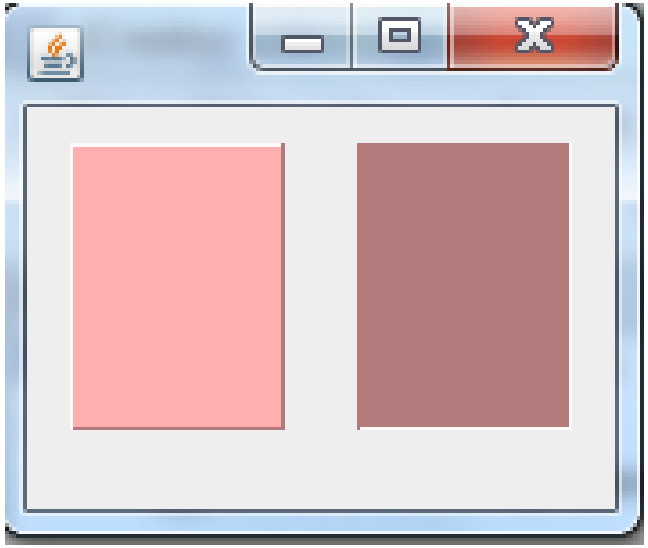
- Mban nje argument te tipit boolean:
  - `true` nese drejtkendeshi eshte me i erret djathtas dhe poshte
  - `false` nese drejtkendeshi eshte me i erret majtas dhe lart.

- `Metoda fill3DRect()`
  - Krijon drejtkendesh te mbushur tredimensionale.

```
// Shembull
import javax.swing.*;
import java.awt.*;
import java.awt.Color;
public class JDemo3DRectangles extends JFrame
```

```
{
    public void paint(Graphics gr)
    {
        super.paint(gr);
        final int WIDTH = 60, HEIGHT = 80;
        gr.setColor(Color.PINK);
        gr.fill3DRect(20, 40, WIDTH, HEIGHT, true);
        gr.fill3DRect(100, 40, WIDTH, HEIGHT, false);
    }
    public static void main(String[] args)
    {
```

```
        JDemo3DRectangles frame = new
JDemo3DRectangles();
        frame.setSize(180, 150);
        frame.setVisible(true);
    }
}
\
```

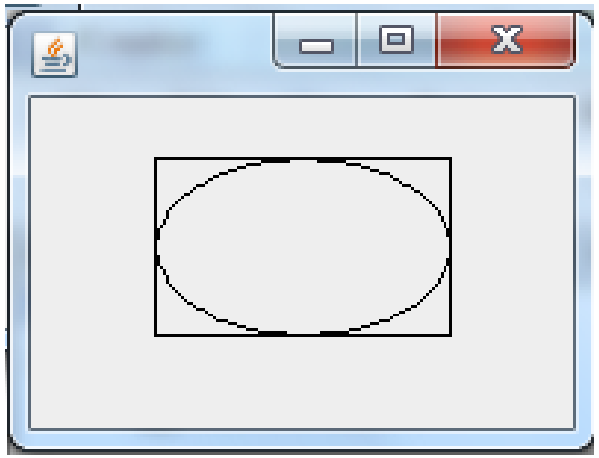


Vizatimi i rratheve

- **`Metodat drawOval()` dhe `fillOval()`**
  - **vizatojne rrathe duke marre te njejtet argumenta si drejtkendeshi.**

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class JDemoOval extends JFrame
{
    public void paint(Graphics tool)
    {
        super.paint(tool);
```

```
            tool.drawRect(50, 50, 100, 60);
            tool.drawOval(50, 50, 100, 60);
        }
        public static void main(String[] args)
        {
            JDemoOval frame = new JDemoOval();
            frame.setSize(200, 150);
            frame.setVisible(true);
        }
}
```



Vizatimi i harqeve

- `drawArc()` method arguments:
  - x- and y-coordinates of the upper-left corner of an imaginary rectangle that represents the bounds of the imaginary circle that contains the arc
  - The width and height of the imaginary rectangle that represents the bounds of the imaginary circle

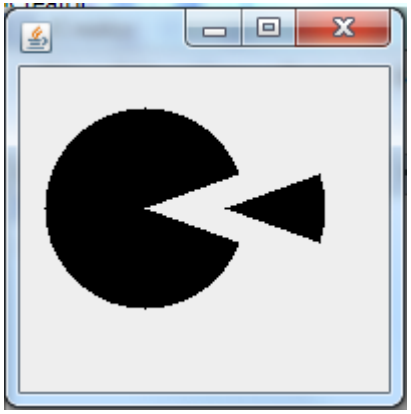
that contains the arc
  - The beginning arc position
  - The arc angle
- `fillArc()` method
  - Creates a solid arc
    - Two straight lines are drawn from the arc endpoints to the center of the imaginary circle whose perimeter the arc occupies

```
import javax.swing.*;
import java.awt.*;
public class JDemoFillArc extends JFrame
{
    public void paint(Graphics g)
```

```
{
        super.paint(g);
        g.fillArc(20, 50, 100, 100, 20, 320);
        g.fillArc(60, 50, 100, 100, 340, 40);

}
public static void main(String[] args)
{
        JDemoFillArc frame = new JDemoFillArc();
        frame.setSize(200, 200);
        frame.setVisible(true);
}
```

```
}
```



Creating Polygons

- `drawPolygon()` method
  - Draws complex shapes
  - Requires three arguments:
    - The integer array, which holds a series of x-coordinate positions

    - The second array, which holds a series of corresponding y-coordinate positions
    - The number of pairs of points to connect
- `fillPolygon()` method
  - Draws a solid shape
  - If the beginning and ending points are not identical, two endpoints are connected by a straight line before the polygon is filled with color
- `addPoint()` method
  - Adds points to a polygon indefinitely

Kopjimi i nje zone

- `copyArea()` method

- Requires six parameters:
  - The x- and y-coordinates of the upper-left corner of the area to be copied
  - The width and height of the area to be copied
  - The horizontal and vertical displacement of the destination of the copy

Perdorimi i metodes `paintComponent()` me `JPanels`

- Perdorni metoden `paintComponent()` kur krijoni vizatime ne nje `JPanel`
- `JFrame` nuk eshte " child " `JComponent`

- Per kete arsye nuk ka metode te veten `paintComponent()` method

Me teper rreth klases Font

- `Metoda getAvailableFontFamilyNames()`
  - Eshte pjese e klases `GraphicsEnvironment` qe ben pjese ne paketen `java.awt`
  - Kthen nje vektor me objekte `String` qe jane emrat e fonteve te vlefshme.
  - Ju nuk mund te krijoni nje objekt ne menyre direkte te `GraphicsEnvironment`
    - Therrisni metoden statike `getLocalGraphicsEnvironment()`

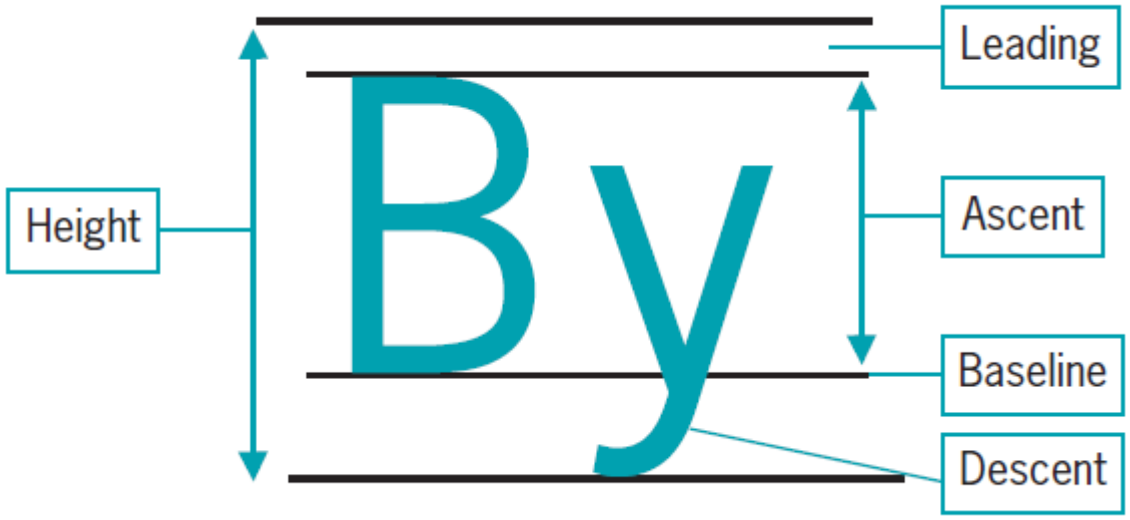
Zbulimi i statistikave te ekranit duke perdorur klasen `Toolkit`

- **`Metoda getDefaultToolkit()`**
  - **Siguron informacion rreth sistemit ne perdorim**
- **`Metoda getScreenResolution()`**
  - **kthen numrin e pixels si nje integer**
- **Ju mund te krijoni nje objekt `Toolkit` dhe te merrni rezolucionin e ekranit duke perdorur kodin:**

```
Toolkit tk =
  Toolkit.getDefaultToolkit();
int resolution =
  tk.getScreenResolution();
```

- **`Klasa Dimension`**
  - **Perdoret per te perfaqesuar gjeresine dhe gjatesine e nje komponenti ne nderfaqen e perdoruesit**
  - **Konstruktoret:**
    - **`Dimension()` Krijon nje objekt te klases `Dimension` me gjeresi dhe gjatesi 0**
    - **`Dimension(Dimension d)` krijon nje objekt me te njejtat dimensione si dimensioni d qe merr si argument.**
    - **`Dimension(int width, int height)` krijon nje Dimension me gjeresi dhe gjatesi te percaktuar.**

- **`Metoda getScreenSize()`**
  - **eshte anetar i objektit `Toolkit`**
  - **Kthen nje objekt te tipit `Dimension`, e cila specifikon gjeresine dhe gjatesine e ekranit ne pixels.**



- **Leading**
  - **hapesira midis baselines**
- **Ascent**
- **Gjatesia e nje germe kapitalenga baseline**
- **Descent**
  - **pjesa e karaktereve qe zbret poshte baseline**
- **Gjatesia e nje font-i**
  - **shuma e leading, ascent, dhe descent**
- **`Metoda getFontMetrics()`**
  - **jep gjatesine e nje font-i**
  - **Kthen objektin `FontMetrics`**
- **Perdorni nje nga metodat e klases `FontMetrics`**

me objekt, per te kthyer nje nga statistikat `Font`-it

- `public int getLeading()`
- `public int getAscent()`
- `public int getDescent()`
- `public int getHeight()`

- `Metoda stringWidth()`
  - Kthen gjeresine e nje `String`-u
  - Kerkon emrin e `String`-ut
  - Eshte anetar i klases `FontMetrics`

Vizatimi me grafiket Java 2D

- Java 2D
  - Kualitet me i larte, grafike dydiemsionale (2D), imazhe dhe tekst

- **`Klasa Graphics2D`**
  - **Vecorite perfshijne:**
    - **Fill patterns**
    - **Strokes**
    - **Anti-aliasing**
  - **ndodhet ne paketen `java.awt`**
- **Procesi i vizatimit me objektet Java 2D:**
  - **Specifikimi i atributave (rendering)**
  - **Vendosja e konturit te vizatimit (drawing stroke)**
  - **krijon objektet qe do vizatohen**

Specifikimi i “Rendering Attributes”

- Perdorni metoden `setColor()`
  - Specifikoni ngjyrat 2D
  - Perdorimi i nje objekti `Graphics2D` dhe vendosja e ngjyres se zeze
    - `gr2D.setColor(Color.BLACK);`
- Fill patterns
  - Kontrollon se si do te mbushet nje objekt vizatimi
  - Mund te jete solid, gradient, texture, ose pattern
  - Krijohet duke perdorur metoden `setPaint()` te `Graphics2D`.
- Gradient fill
  - Kalimi gradual nga nje ngjyre ne nje ngjyre tjeter

- Acyclic gradient
- Cyclic gradient

Vendosja e nje konturi

- Stroke
  - Perfaqeson nje levizje te vetme
  - `Metoda setStroke()`
- `Nderfaqja Stroke`
- `Klasa BasicStroke`
- Stilet Endcap
  - Aplikohet ne fund te vijave qe nuk bazhkohen me vija te tjera.

- **Stilet Juncture**
  - **Per vijat qe bashkohen**

    Krijimi i objekteve qe do vizatohen
  - **Perdorni paketen `java.awt.geom`**

Vijat

- **`Line2D.Float`**
- **`Line2D.Double`**
- **`Point2D.Float`**
- **`Point2D.Double`**

Drejtkendeshat

- **`Rectangle2D.Float`**
- **`Rectangle2D.Double`**
- **`Rectangle2D.Float rect = new`**

```
  Rectangle2D.Float(10F, 10F, 50F, 40F);
```

Rrathet

- **Ellipse2D.Float**
- **Ellipse2D.Double**
- **Ellipse2D.Float ell = new Ellipse2D.Float(10F, 73F, 40F, 20F);**

Harqet

- **Arc2D.Float**
- **Arc2D.Double**
- **Arc2D.PIE**
- **Arc2D.CHORD**
- **Arc2D.OPEN**
- **Arc2D.Float ac = new Arc2D.Float**

```
(10,133,30,33,30,120,Arc2D.PIE);
```

Poligonet

- `GeneralPath pol = new GeneralPath();`
- `moveTo()`
- `lineTo()`
- `closePath()`