

Uitwerking derde deeltentamen Imperatief Programmeren

donderdag 8 april 2004

Opgave 1

- Een abstracte klasse dient voor het vastleggen van een gedeeltelijke implementatie van een interface.
- Een `KeyListener` reageert op het gebruik van het toetsenbord. Een klasse implementeert deze interface door een methode `keyTyped(KeyEvent)` te definiëren. Een object van zo'n klasse wordt aan een component toegevoegd met behulp van de methode `addKeyListener(KeyListener)` van `Component`.
- Een `FileInputStream` interpreteert de inhoud van een bestand als bytes, een `FileReader` als characters.
- De klasse `BasicStroke` dient voor het vastleggen van eigenschappen van krommen (zoals patroon en pendikte) in `Graphics2D`.
- Een `Map` representeert een geïndexeerde verzameling. Als indices kunnen strings of andere objecten worden gebruikt. `Map` is een interface; deze wordt gebruikt door objecten aan te maken van een implementerende klasse zoals `HashMap`.
- Het object `System.in` representeert de stroom bytes die van het toetsenbord afkomstig is. Het wordt gewoonlijk gebruikt als argument in de constructor van een `InputStreamReader`.

Opgave 2

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

class WindowSluiter extends WindowAdapter
{
 public void windowClosing(WindowEvent e)
 {
 System.exit(0);
 }
}
```
- ```
in = new FileInputStream(naam);
tekst.setText("");
teken = in.read();
teller = 0;
while (teken != -1)
{
    uit = hex(teken);
    tekst.append(uit + " ");
    teken = in.read();
    teller++;
}
```

```

        if (teller >= 50)
        {
            tekst.append("\n");
            teller = 0;
        }
    }
    in.close();
c. public static String hex(int n) // pre 0 <= n <= 255
    {
        return hexdig(n / 16) + hexdig(n % 16);
    }
    public static String hexdig(int n) // pre 0 <= n <= 15
    {
        if (n <= 9) return "" + n;
        return "" + (char)(n - 10 + 'A');
    }
d. Als de laatste twee opdrachten in de constructor waren vervangen door
    this.add(BorderLayout.SOUTH, openButton); , was de 'Open'-knop zo breed
    als het hele frame geweest.
e. private void lees(String naam)
    {
        int r, g, b, hor, vert;
        InputStream in;
        Color uit;
        try
        {
            in = new FileInputStream(naam);
            hor = 0; vert = 0;
            while (true)
            {
                r = in.read();
                if (r == -1) break;
                g = in.read();
                if (g == -1) break;
                b = in.read();
                if (b == -1) break;
                uit = new Color(r, g, b);
                canvas.teken(uit, hor, vert);
                hor++;
                if (hor >= 40)
                {
                    hor = 0;
                    vert++;
                }
            }
            canvas.repaint();
            in.close();
        }
        catch (Exception e)
        {
            System.out.println("Lezen mislukt, " + e.getMessage());
        }
    }
}

```

Opgave 3

```
private void lees(String naam)
{
    FileReader fr;
    BufferedReader br;
    String regel;
    StringTokenizer st;
    Set woorden;
    int teller;
    woorden = new HashSet();
    teller = 0;
    tekst.setText("File: " + naam + "\n");
    try
    {
        fr = new FileReader(naam);
        br = new BufferedReader(fr);
        while ((regel = br.readLine()) != null)
        {
            st = new StringTokenizer(regel, " ");
            while (st.hasMoreTokens())
            {
                woorden.add(st.nextToken());
                teller++;
            }
        }
        tekst.append("Aantal woorden: " + teller + "\n");
        tekst.append("Aantal verschillende: " + woorden.size());
    }
    catch (Exception e)
    {
        tekst.setText("Lezen mislukt, " + e.getMessage());
    }
}
```