



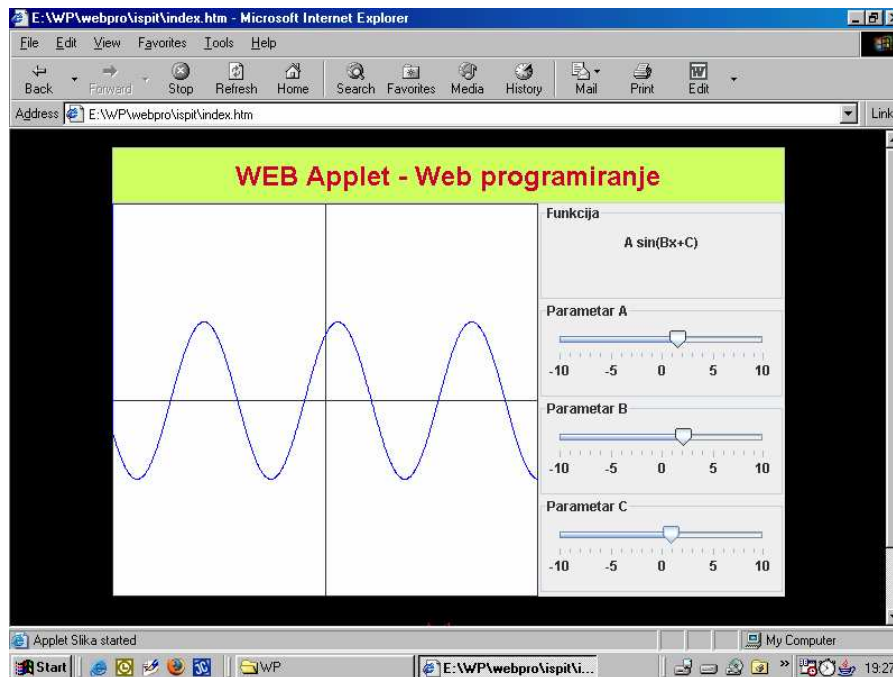
WEB Programiranje

[A p p l e t]

Davor Menon
davor.menon@gmail.com

20. veljače 2006.

1 Slika appleta



Slika 1: Moj Applet

2 Kod Appleta

Applet se može naći na mojim stranicama www.mathos.hr/~dmenon

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

```
public class Slika extends JApplet implements ChangeListener {
```

```
    Ploca papir;
```

```
    public void init() // Ono sto browser izvodi prije pokretanja Apleta
```

```
{
    JTextField naslov = new JTextField();

    naslov.setBackground(new Color(204, 255, 102));
    naslov.setEditable(false);
    naslov.setFont(new Font("Dialog", 1, 24));
    naslov.setForeground(new Color(204, 0, 51));
    naslov.setHorizontalAlignment(JTextField.CENTER);
    naslov.setText("WEB Applet - Web programiranje");
    naslov.setPreferredSize(new Dimension(450, 50));

    JSlider ParametarA;
    JSlider ParametarB;
    JSlider ParametarC;
    ParametarA = new JSlider(-10,10,1);
    ParametarB = new JSlider(-10,10,1);
    ParametarC = new JSlider(-10,10,0);

    ParametarA.setMinorTickSpacing(1);
    ParametarA.setMajorTickSpacing(5);
    ParametarA.setLabelTable(ParametarA.createStandardLabels(5));
    ParametarA.setPaintTicks(true);
    ParametarA.setPaintLabels(true);

    ParametarB.setMinorTickSpacing(1);
    ParametarB.setMajorTickSpacing(5);
    ParametarB.setLabelTable(ParametarB.createStandardLabels(5));
    ParametarB.setPaintTicks(true);
    ParametarB.setPaintLabels(true);

    ParametarC.setMinorTickSpacing(1);
    ParametarC.setMajorTickSpacing(5);
    ParametarC.setLabelTable(ParametarA.createStandardLabels(5));
    ParametarC.setPaintTicks(true);
    ParametarC.setPaintLabels(true);

    ParametarA.addChangeListener(this);
    ParametarB.addChangeListener(this);
    ParametarC.addChangeListener(this);

    JPanel kucicaA, kucicaB, kucicaC;
```

```
kucicaA = new JPanel();
kucicaA.setBorder(new TitledBorder("Parametar A"));
kucicaA.add(ParametarA);

kucicaB = new JPanel();
kucicaB.setBorder(new TitledBorder("Parametar B"));
kucicaB.add(ParametarB);

kucicaC = new JPanel();
kucicaC.setBorder(new TitledBorder("Parametar C"));
kucicaC.add(ParametarC);

JLabel oznaka = new JLabel("A sin(Bx+C)");
JPanel kucicaFje = new JPanel();
kucicaFje.setBorder(new TitledBorder("Funkcija"));
kucicaFje.add(oznaka);

JPanel gumbici = new JPanel();
gumbici.setLayout(new GridLayout(4,1));
gumbici.add(kucicaFje);
gumbici.add(kucicaA);
gumbici.add(kucicaB);
gumbici.add(kucicaC);

Sinus funkcija;
funkcija = new Sinus(ParametarA, ParametarB, ParametarC);

papir = new Ploca(funkcija,-5,5,-5,5);

getContentPane().setLayout(new BorderLayout());
getContentPane().add(naslov, BorderLayout.NORTH);
getContentPane().add(papir, BorderLayout.CENTER);
getContentPane().add(gumbici, BorderLayout.EAST);
}

private class Ploca extends JPanel
{
    double xm;
    double xM;
    double ym;
```

```
double yM;
Sinus f;

int n = 1000;

Ploca()
{
    xm = -5;
    xM = 5;
    ym = -5;
    yM = 5;
    f = new Sinus(1,1,0);
}

Ploca(Sinus fja, double a, double b, double c, double d)
{
    xm = a;
    xM = b;
    ym = c;
    yM = d;
    f = fja;
}

public void paintComponent(Graphics g)
{
    setBackground(Color.white);
    super.paintComponent(g);

    UTocka M = new UTocka(getWidth(),getHeight());

    g.drawRect(0,0,getWidth()-1,getHeight()-1);

    UTocka P = uvoz(new VTocka(0,0),M);

    g.drawLine(P.x,0,P.x,M.y-1);
    g.drawLine(0,P.y,M.x-1,P.y);

    crtaj(f, Color.blue, g, M);
}

public void crtaj(Sinus fja, Color boja, Graphics obris, UTocka M)
```

```
{

    obris.setColor(boja);

    UTocka u = new UTocka();
    VTocka v = new VTocka();
    UTocka su = u; // staro u

    double h = ( xM - xm ) / n;

    for (int i=0; i<n; i=i+1)
    {
        v.x = xm + i * h;
        v.y = fja.vrijednost(v.x);
        u = uvoz(v, M);
        obris.drawLine(su.x,su.y,u.x,u.y);
        su = u;
    }

    obris.dispose();
}

public UTocka uvoz(VTocka T, UTocka U)
{
    int sirina = U.x;
    int dubina = U.y;

    UTocka I = new UTocka();

    I.x = (int) Math.round( ( T.x - xm )/( ( xM - xm ) / ( sirina ) ) );
    I.y = (int) Math.round( ( yM - T.y )/( ( yM - ym ) / ( dubina ) ) );

    return I;
}

public VTocka izvoz(UTocka T, UTocka U)
{
    int sirina = U.x;
    int dubina = U.y;

    VTocka I = new VTocka();
```

```
        I.x = xm + ( T.x * ( ( xM - xm )/sirina ) );
        I.y = xM - ( T.y * ( ( yM - ym )/dubina ) );

        return I;
    }
}

private class UTocka
{
    int x;
    int y;

    UTocka()
    {
    }

    UTocka(int a, int b)
    {
        x = a;
        y = b;
    }
}

private class VTocka
{
    double x;
    double y;

    VTocka()
    {
    }

    VTocka(double a, double b)
    {
        x = a;
        y = b;
    }
}

private class Sinus
```

```
{
    JSlider A, B, C;

    Sinus(int a, int b, int c)
    {
        A = new JSlider(-10,10,a);
        B = new JSlider(-10,10,b);
        C = new JSlider(-10,10,c);
    }

    Sinus(JSlider a, JSlider b, JSlider c)
    {
        A = a;
        B = b;
        C = c;
    }

    public double vrijednost(double x)
    {
        return A.getValue() * Math.sin(B.getValue()*x+C.getValue());
    }

    public String izraz()
    {
        return A.getValue()+" sin ( "+B.getValue()+" x "+C.getValue()+"
    }

}

public void stateChanged(ChangeEvent evt)
{
    papir.repaint();
}
}
```