import javax.microedition.lcdui.\*;  
import javax.microedition.midlet.MIDlet;  
public final class EmiMIDlet extends MIDlet implements CommandListener {

private static final int NUM\_SIZE = 20;

private final Command exitCmd = new Command(“Exit”, Command.EXIT, 2);

private final Command calcCmd = new Command(“Calc”, Command.SCREEN, 1);

private final TextField t1 = new TextField(“Principle Amount”, “”, NUM\_SIZE, TextField.DECIMAL);

private final TextField t2 = new TextField(“Rate of Interest(pcpa)”, “”, NUM\_SIZE, TextField.DECIMAL);

private final TextField t3 = new TextField(“Tenure(in months)”, “”, NUM\_SIZE, TextField.DECIMAL);

private final TextField tr = new TextField(“EMI”, “”, NUM\_SIZE, TextField.UNEDITABLE);

private final Alert alert = new Alert(“Error”, “”, null, AlertType.ERROR); // An alert to be reused for different errors.

private boolean isInitialized = false;

protected void startApp()  
{  
if (isInitialized)  
{  
return;  
}

Form f = new Form(“EMI Calculator”);  
f.append(t1);  
f.append(t2);  
f.append(t3);  
f.append(tr);  
f.addCommand(exitCmd);  
f.addCommand(calcCmd);  
f.setCommandListener(this);  
Display.getDisplay(this).setCurrent(f);  
alert.addCommand(new Command(“Back”, Command.SCREEN, 1));  
isInitialized = true;  
}

protected void destroyApp(boolean unconditional)  
{  
}

protected void pauseApp()  
{  
}

public double power(double a,double b)  
{  
double ans=1.0;  
int i;  
for(i=0;i<b;i++)  
ans=ans\*a;  
return ans;  
}

public void commandAction(Command c, Displayable d)  
{  
if (c == exitCmd)  
{  
destroyApp(false);  
notifyDestroyed();

return;  
}

double res = 0.0;  
double temp = 0.0;

try  
{  
double principle = getNumber(t1, “First”);  
double roi = getNumber(t2, “Second”);  
double tenure = getNumber(t3,”Third”);  
roi=roi/1200; //roi is taken annually, converting it to monthly  
temp=power((1+roi),tenure); //Math.power doesn’t work  
res=(principle\*roi\*temp)/(temp-1);  
}  
catch (NumberFormatException e)  
{  
return;  
}  
catch (ArithmeticException e)  
{  
alert.setString(“Divide by zero.”);  
Display.getDisplay(this).setCurrent(alert);  
return;  
}

String res\_str = Double.toString(res); //The resulted string may exceed the text max size.

if (res\_str.length() > tr.getMaxSize())  
{  
tr.setMaxSize(res\_str.length());  
}

tr.setString(res\_str);  
}

private double getNumber(TextField t, String type) throws NumberFormatException //Extracts the double number from text field.  
{  
String s = t.getString();

if (s.length() == 0)  
{  
alert.setString(“No ” + type + ” Argument”);  
Display.getDisplay(this).setCurrent(alert);  
throw new NumberFormatException();  
}

double n;

try  
{  
n = Double.parseDouble(s);  
}  
catch (NumberFormatException e)  
{  
alert.setString(type + ” argument is out of range.”);  
Display.getDisplay(this).setCurrent(alert);  
throw e;  
}

return n;  
}  
} // end of class ‘EmiMIDlet’ definition