

Action Performed

Random Tip.



Compare and Contrast ActionPerformed and Init

	Init	ActionPerformed
How Often It Runs		
When it is run		
General Purpose		
Lines of Code Found there		
Lines of Code NOT found there		

Compare and Contrast ActionPerformed and Init

	Init	ActionPerformed
How Often It Runs	Once	The user decides.
When it is run		
General Purpose		
Lines of Code Found there		
Lines of Code NOT found there		

Compare and Contrast ActionPerformed and Init

	Init	ActionPerformed
How Often It Runs	Once	The user decides.
When it is run	When the applet begins	When the user clicks a button
General Purpose		
Lines of Code Found there		
Lines of Code NOT found there		

Compare and Contrast ActionPerformed and Init

	Init	ActionPerformed
How Often It Runs	Once	The user decides.
When it is run	When the applet begins	When the user clicks a button
General Purpose	To set up the screen	To respond to a user's request
Lines of Code Found there		
Lines of Code NOT found there		

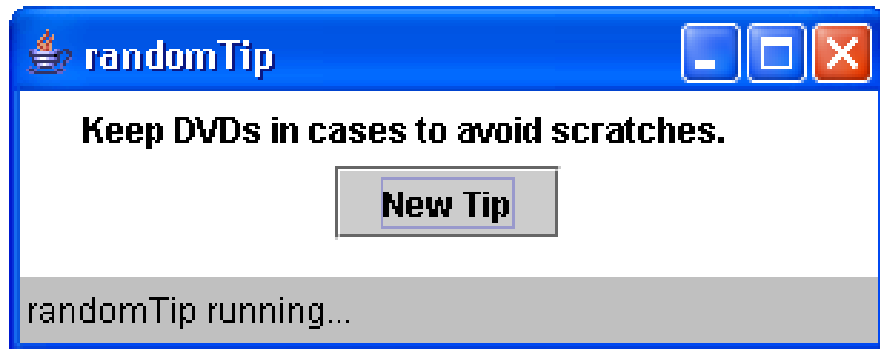
Compare and Contrast ActionPerformed and Init

	Init	ActionPerformed
How Often It Runs	Once	The user decides.
When it is run	When the applet begins	When the user clicks a button
General Purpose	To set up the screen	To respond to a user's request
Lines of Code Found there	Declaration Construction (new) Mutation (set) Add addActionListener setActionCommand	Mutation (set) Accessors (get) showStatus
Lines of Code NOT found there		

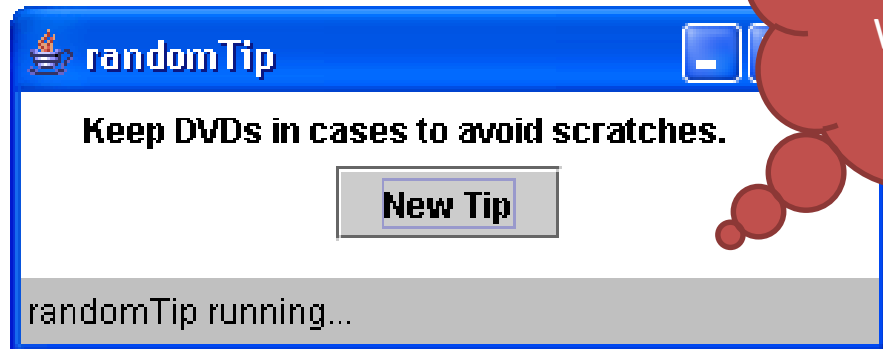
Compare and Contrast ActionPerformed and Init

	Init	ActionPerformed
How Often It Runs	Once	The user decides.
When it is run	When the applet begins	When the user clicks a button
General Purpose	To set up the screen	To respond to a user's request
Lines of Code Found there	Declaration Construction (new) Mutation (set) Add addActionListener setActionCommand	Mutation (set) Accessors (get) showStatus
Lines of Code NOT found there	Accessors (get) – you can, it is just pointless.	Construction (new) Add

This program gives a series of incredibly useful tips about DVD use.



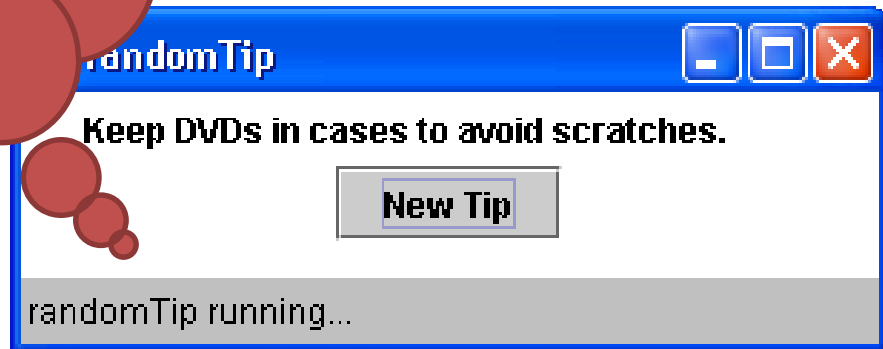
This program gives a series of incredibly useful tips about DVD use.



How many widgets are on the screen?
What type are they?

This program gives a series of incredibly useful tips about DVD use.

Hey, one of those widgets is changing... Which one?



Setting up the screen

In init method

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;  
import java.applet.Applet;
```

What are these?

```
public class randomTip extends Applet implements ActionListener  
{  
    JLabel tip;  
  
    public void init ()  
    {  
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");  
        JButton b1 = new JButton ("New Tip");  
        b1.addActionListener (this);  
        b1.setActionCommand ("tip me!");  
        add (tip);  
        add (b1);  
    }  
}
```



```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;  
import java.applet.Applet;
```

A. Libraries. Code
we are borrowing.

```
public class randomTip extends Applet implements ActionListener  
{  
    JLabel tip;  
  
    public void init ()  
    {  
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");  
        JButton b1 = new JButton ("New Tip");  
        b1.addActionListener (this);  
        b1.setActionCommand ("tip me!");  
        add (tip);  
        add (b1);  
    }  
}
```



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

What's this?

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

Code that makes
the applet surface.

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```



What's this?

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;  
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener  
{  
    JLabel tip;  
  
    public void init ()  
    {  
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");  
        JButton b1 = new JButton ("New Tip");  
        b1.addActionListener (this);  
        b1.setActionCommand ("tip me!");  
        add (tip);  
        add (b1);  
    }  
}
```



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

Code that links to the
"Actions" so our
buttons work

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener
{
```

```
    JLabel tip;
```

What's this?

```
    public void init ()
```

```
    {
```

```
         tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
```

```
        JButton b1 = new JButton ("New Tip");
```

```
        b1.addActionListener (this);
```

```
        b1.setActionCommand ("tip me!");
```

```
        add (tip);
```

```
        add (b1);
```

```
    }
```



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet
{
```

```
    JLabel tip;
```

```
    public void init ()
```

```
    {
```

```
         tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
```

```
        JButton b1 = new JButton ("New Tip");
```

```
        b1.addActionListener (this);
```

```
        b1.setActionCommand ("tip me!");
```

```
        add (tip);
```

```
        add (b1);
```

```
    }
```

A Global variable. It will be changing. It is declared here so that both init and AP can use it.



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener
{
```

```
    JLabel tip;
```

```
    public void init ()
```

```
    {
```

```
         tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
```

```
        JButton b1 = new JButton ("New Tip");
```

```
        b1.addActionListener (this);
```

```
        b1.setActionCommand ("tip me!");
```

```
        add (tip);
```

```
        add (b1);
```

```
    }
```

Why is there a box?



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends
{
```

```
    JLabel tip;
```

```
    public void init()
    {
```

```
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
```

It's highlighting the fact that the variable has already been declared above. There is no type (JLabel) here.



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```

What's the name of
this widget?



```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;  
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener  
{  
    JLabel tip;  
  
    public void init ()  
    {  
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");  
        JButton b1 = new JButton ("New Tip");  
        b1.addActionListener (this);  
        b1.setActionCommand ("tip me!");  
        add (tip);  
        add (b1);  
    }  
}
```

Tip (It's global)



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```

What's the name of
this widget?

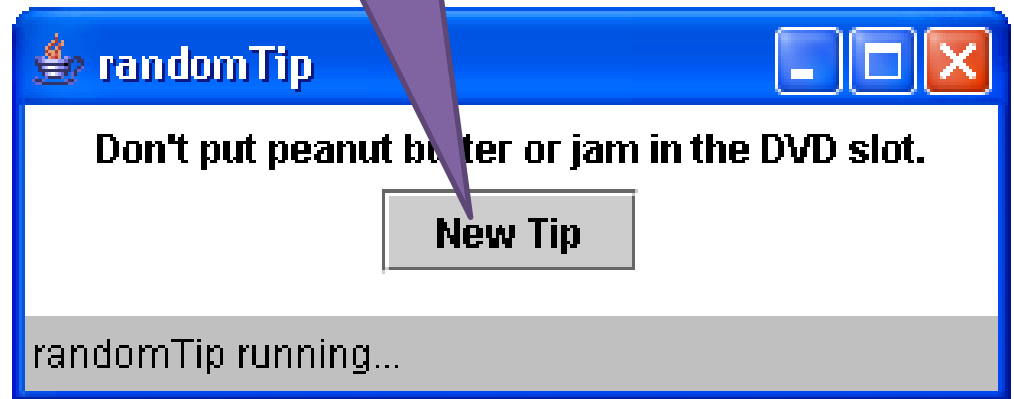


```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```

b1. It's not global.



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```

Which widget has an
Action Listener?



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;
```

```
public class randomTip extends Applet implements ActionListener
{
    JLabel tip;

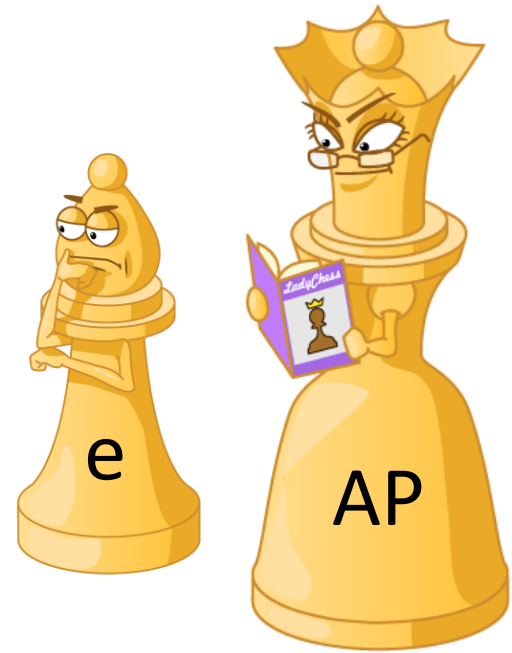
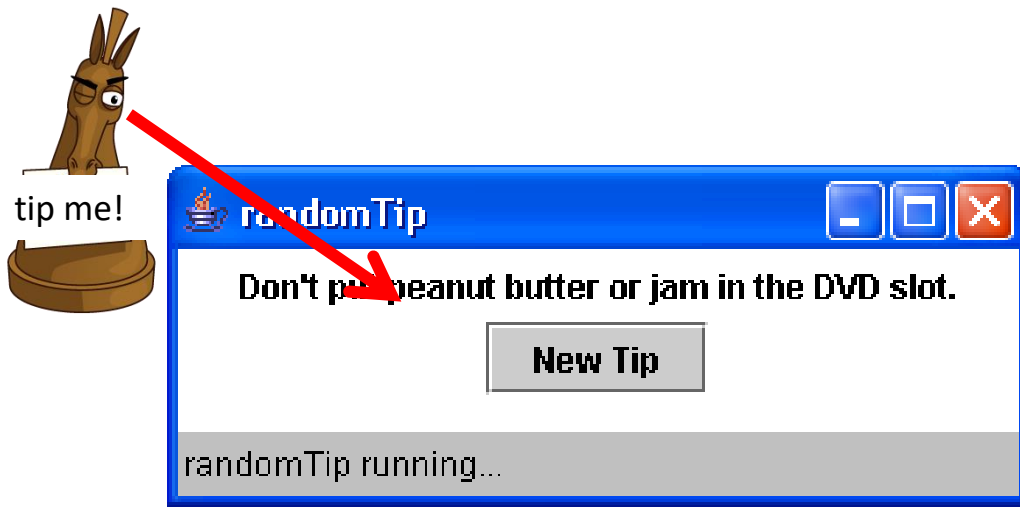
    public void init ()
    {
        tip = new JLabel ("Don't put peanut butter or jam in the DVD slot.");
        JButton b1 = new JButton ("New Tip");
        b1.addActionListener (this);
        b1.setActionCommand ("tip me!");
        add (tip);
        add (b1);
    }
}
```

The JButton. It's
Action Command is
"tip me!"



Using ActionPerformed

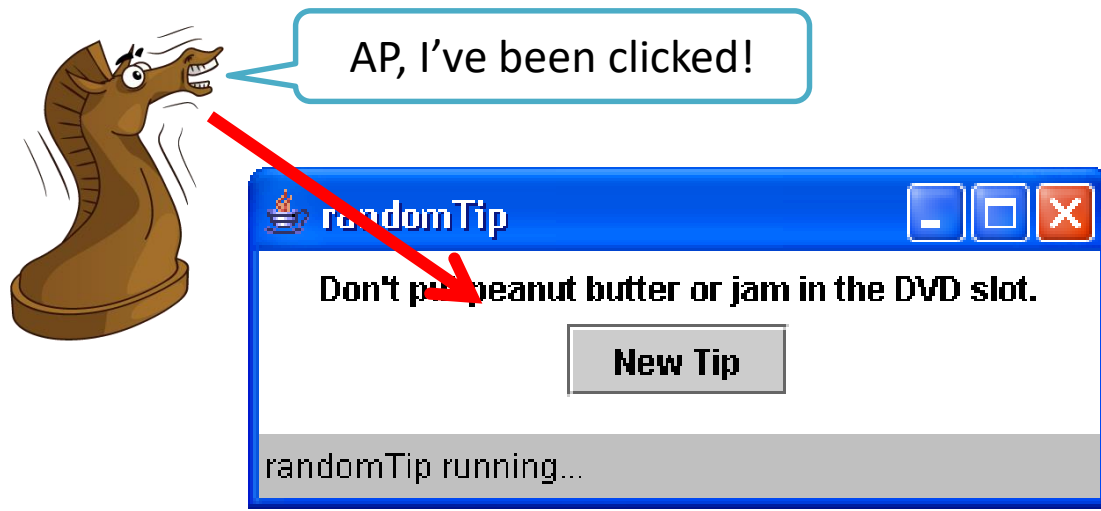
and 'e', the ActionEvent



```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



```
public void actionPerformed(ActionEvent e)
{
    if (e.getActionCommand().equals("tip me!"))
    {
        int n = (int) ... + 1);
        if (n == 1)
            tip.setText("Please use cases to avoid scratches.");
        else if (n == ...)
            tip.setText("Don't use DVDs as frisbees.");
        else
            tip.setText("Don't eat the DVDs. They aren't tasty.");
    }
}
```



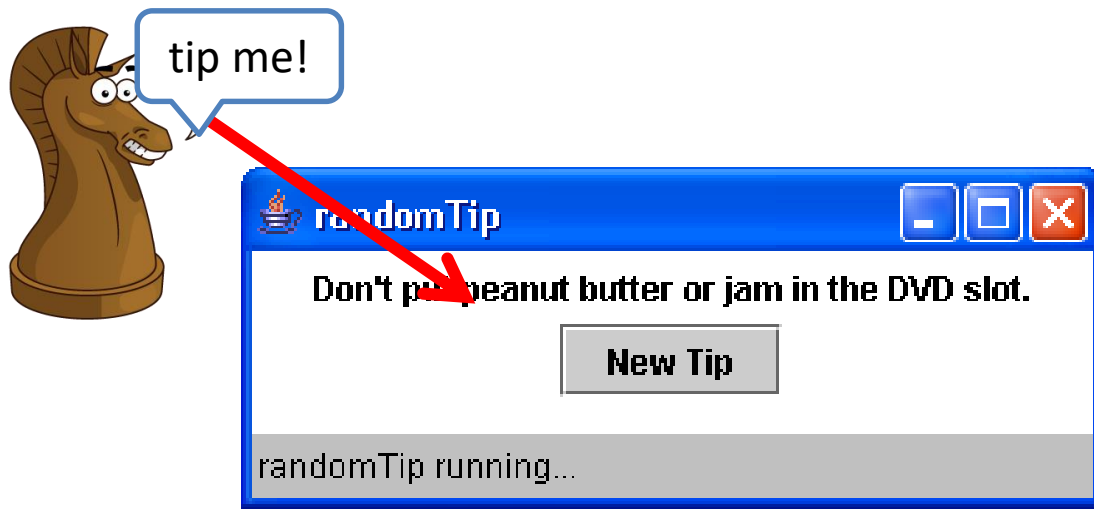
```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



What's the AC?



```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



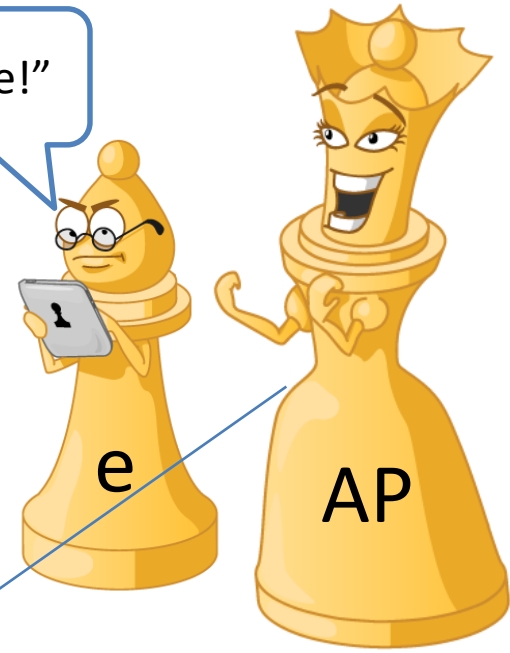
Got it. "tip me!"



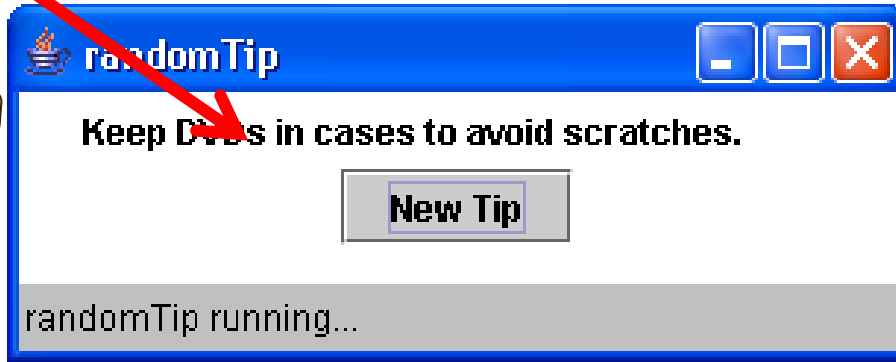
```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



"tip me!"



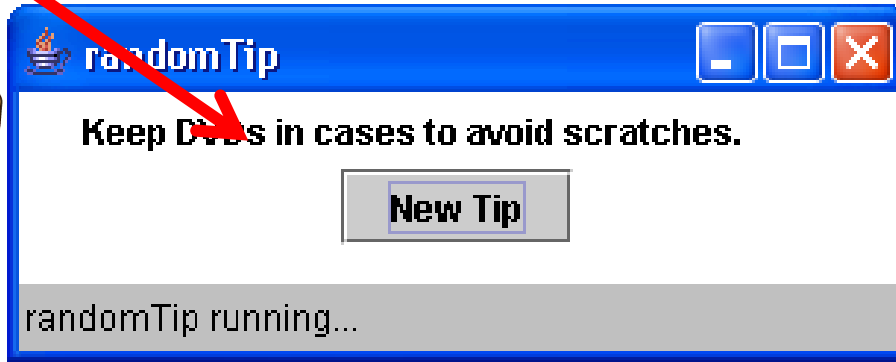
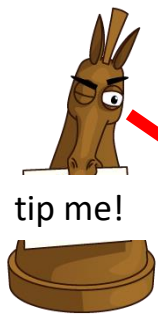
```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



"tip me!"



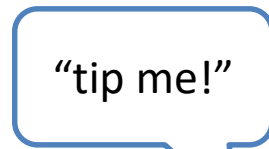
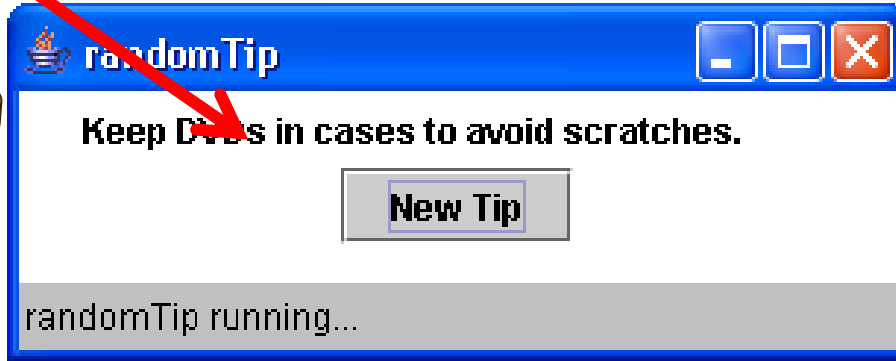
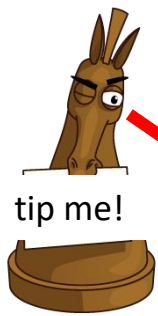
```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



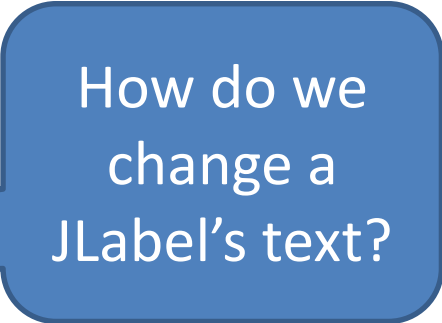
"tip me!"

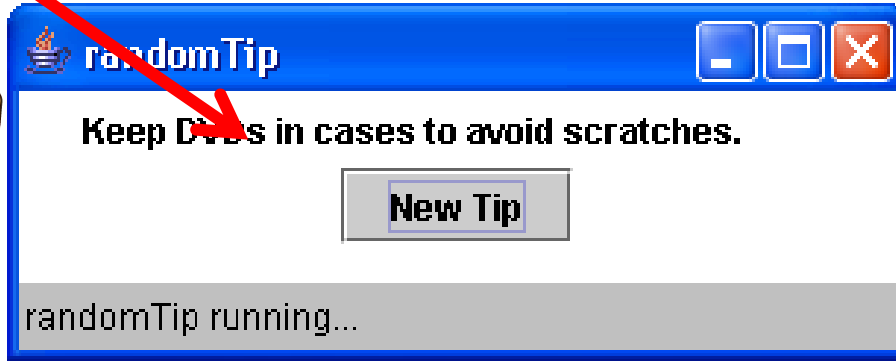
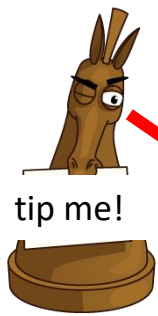


```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```



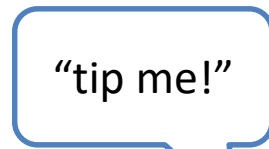
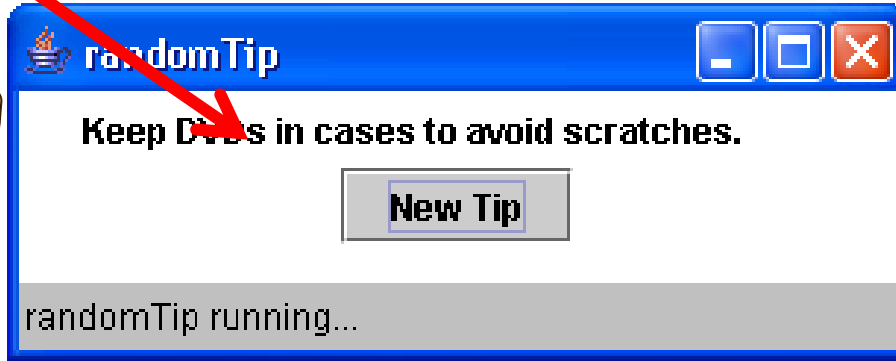
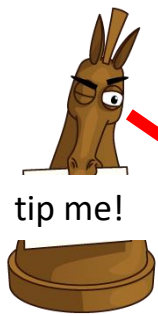


"tip me!"

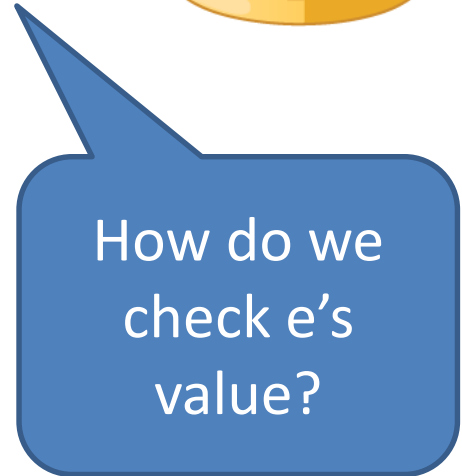


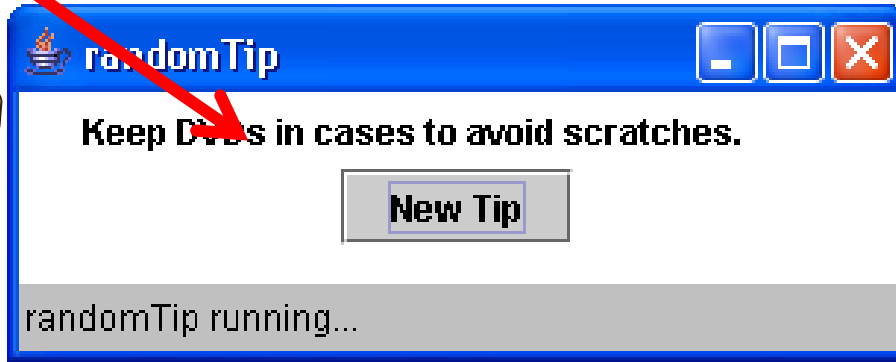
```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```

setText.



```
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("tip me!"))
    {
        int n = (int) (Math.random () * 3 + 1);
        if (n == 1)
            tip.setText ("Keep DVDs in cases to avoid scratches.");
        else if (n == 2)
            tip.setText ("Don't use DVDs as frisbees.");
        else
            tip.setText ("Don't eat the DVDs. They aren't tasty.");
    }
}
```





"tip me!"



```
public void actionPerformed (ActionEvent e)
```

```
{  
  if (e.getActionCommand ().equals ("tip me!"))  
  {  
    int n = (int) (Math.random () * 3 + 1);  
    if (n == 1)  
      tip.setText ("Keep DVDs in cases to avoid scratches.");  
    else if (n == 2)  
      tip.setText ("Don't use DVDs as frisbees.");  
    else  
      tip.setText ("Don't eat the DVDs. They aren't tasty.");  
  }  
}
```

Like this.