

Version 10.0 – 11/15/2007

Purpose	Command	Example
Section I: Numeric Section		
Generate pseudo random numbers	java.lang.Random	<pre> import java.util.Random; int x; //declares a global integer Random r = new Random(); //one time outside of all methods public void actionPerformed(ActionEvent e) { x = r.nextInt(5); //Gives numbers from 0-4 System.out.println("x = " + x); } </pre>
Converting a string into a floating point value (double)	Double.parseDouble Integer.parseInt	<pre> String str1 = "1337"; //set up a string that has a number in it double x; //From String str1 To double x //Note: If the value of str1 is null (if there is no string), trim() will throw a //NullPointerException. If you don't use trim(), make sure //there's no trailing white space. For JDK 1.2.x or better: try { x = Double.parseDouble(str1.trim()); } </pre>

		<pre> catch (NumberFormatException e) { System.out.println("There was a problem formatting the string, bye"); } //When this command is done, x has the value 1337, if str1 contained //other types of characters like \$>#@jkaBN, then the parseDouble command //would have thrown an exception int y = Integer.parseInt(temp); //also works for ints </pre>
Formatting numbers with specific requirements	DecimalFormat	<pre> Double dd=new Double(1400.54); //some number you have found String gg; //This string will hold the result of our formatting java.text.DecimalFormat df2 = new java.text.DecimalFormat("###,##0.00"); gg = df2.format(dd.doubleValue()); //Scientific notation (JDK 1.2.x on up): java.text.DecimalFormat de = new java.text.DecimalFormat("0.0000000000E00"); gg = de.format(dd.doubleValue()); </pre>
Using math constants	Math.	<pre> //no imports necessary when using Math double radius = 2.5; double circum = Math.PI*2*radius; </pre>
Using math methods	Math.	<pre> //no imports necessary when using Math System.out.println(Math.pow(2,5)); //prints 2 to the 5th power //Figure out the square root of a number double in; in = 5.6; </pre>

		double squareRoot = Math.sqrt(in);
Rounding off a float or a double	An old trick taught to me by Mr. Brodt from Haverford High School	//Let's truncate the number after moving it over a certain # of spaces //let's say the variable to round off is called cost double cost = <some calculation>; cost = cost *100; //move the decimal over 2 places cost = (int) (cost + .5); //add .5 and then chop off decimal cost = cost /100; //move the decimal place back //This trick also works for rounding off to thousandths, etc. //just change the 100 to a 1000 in both places
Get the current date and time and format for printing	Date and DateFormat	import java.util.Date; import java.util.DateFormat; public static void menu() { Date now = new Date(); String display = DateFormat.getDateTimeInstance(DateFormat.FULL, DateFormat.FULL).format(now);
Section II: Console Section		

Reading in data from the keyboard in a console application	java.util.Scanner	NOTE: This requires JDK 1.5 import java.util.Scanner; public class ConsoleDemo1 { public static void main(String[] args) { Scanner input = new Scanner(System.in); //do this once to be able to read in data int x; double y; String name; System.out.println("Please enter your name:"); name = input.nextWord(); //Reads in a string from the keyboard with no spaces System.out.println("Please enter a whole number"); x = input.nextInt(); System.out.println("Please enter a decimal number:"); y = input.nextDouble(); } } import java.util.Scanner; public class ConsoleDemo2 { //To read in a value without spaces followed by a value with spaces public static void main(String[] args) { Scanner input = new Scanner(System.in); //do this once to be able to read in data int x; x = input.nextInt(); //read an int //Must skip the newline in order to read data with spaces next input.skip("\n"); String phrase; phrase = input.nextLine(); }
--	-------------------	--

		}
Printing out variables and phrases to the console	System.out.print and System.out.println	System.out.print("this phrase will be on one line"); System.out.print("***** these stars will on the same line as above"); int myVar = 6; System.out.println("The variable is " + myVar);
Printing out control characters to the console	System.out.print and System.out.println	System.out.print("This will be on one line\nthis on another"); System.out.println("\t\t\tThis will be after 3 tabs"); System.out.println("Here's how to print out a double quote\" OK?"); System.out.println("Here's how to print a back slash \\ OK?");

Section III: Swing Section		
Determining which button was clicked(or menu item, combo box)	<code>e.getSource()</code>	<pre> public void actionPerformed(ActionEvent e) { //was this the radius button if(e.getSource() == radiusBUT) { //logic here for radius button } if(e.getSource() == sodaBUT) { //logic here for soda button } if(e.getSource() == fileExitMI) { System.exit(0); //exit the program } } </pre>
Setting the background of your frame as an image (Props to Ryan Knapp 09)	<code>getLayeredPane().add</code> among other commands	<pre> //Add this to the constructor of your Frame public JavaOutletFrame() { //Load the background Image ImageIcon backIm = new ImageIcon(ClassLoader.getResource("images/CoffeeDrinker.jpg")); JLabel backLBL = new JLabel(backIm); //make a JLabel from background image //Set the bounds of the label to be the whole window </pre>

		<pre> backLBL.setBounds(0, 0, backIm.getIconWidth(), backIm.getIconHeight()); getLayeredPane().add(backLBL, new Integer(Integer.MIN_VALUE)); JPanel myPanel = new JPanel(); myPanel.setOpaque(false); setContentPane(myPanel); try { jbInit(); } catch (Exception ex) { ex.printStackTrace(); } } </pre>
<p>Changing the icon of your swing app(Props to Chris Bouchard 06)</p>	<p>setIconImage and getDefaultToolkit</p>	<pre> //YOU NEED TO FIND ICON IMAGE FILES OR MAKE YOUR OWN. //HERE IS A SITE WHERE I FOUND SOME FREE DOWNLOADABLE ICONS http://www.icongalore.com/sales/purchase-and-download.php public class yourFrame extends JFrame implements.... //Declare this as a global variable Image image1; //inside the jbInit() method somewhere, add this private void jbInit() { //other stuff //load the image image1 = Toolkit.getDefaultToolkit().getImage(java.net.URLClassLoader .getSystemResource("images/world.gif")); //make the image the icon image for this project setIconImage(image1); } </pre>

<p>Making sure that only certain characters are entered into a textfield(props to Jon Diaz)</p>	<p>Creating a class that extends KeyAdapter</p>	<pre>public class yourFrame extends JFrame implements... //assuming you have a textfields called monTF, etc TextField monTF = new JTextField(); //Gen by JBuilder TextField tuesTF = new JTextField(); TextField wedTF = new JTextField(); TextField thurTF = new JTextField(); TextField friTF = new JTextField(); //YOU ADD A GLOBAL DIGIT LISTENER SOMEWHERE //OUTSIDE OF ANY METHOD //Create a digitKeyListener DigitKeyListener dg = new DigitKeyListener(); //inside the jbInit() method somewhere private void jbInit() { //other stuff monTF.addKeyListener(dg); //add digit key listeners to each tuesTF.addKeyListener(dg); wedTF.addKeyListener(dg); thurTF.addKeyListener(dg); friTF.addKeyListener(dg); } //Below the Frame class or in another file import java.awt.event.*; class DigitKeyListener extends KeyAdapter { //consumes all values except numbers period, delete and backspace</pre>
---	---	--

		<pre> public void keyTyped(KeyEvent f) { char ch = f.getKeyChar(); //If it's not a period, backspace, digit or delete... if (!((ch == KeyEvent.VK_PERIOD) (Character.isDigit(ch)) (ch == KeyEvent.VK_BACK_SPACE) (ch == KeyEvent.VK_DELETE) ch == KeyEvent.VK_MINUS))) { f.consume(); //prevent from appearing in the textfield } } </pre>
<p>Making a field uneditable or read-only</p>	<p>setEditable</p>	<pre> scoreFieldTF.setEditable(false); </pre>
<p>Change the background color of a button</p>	<p>setBackground</p>	<pre> b1.setBackground(Color.blue); //NOTE: b1 is an initialized Button reference </pre>

<p>Change the foreground color of a button</p>	<p>setForeground</p>	<p>b1.setForeground(Color.yellow); //NOTE: b1 is an initialized Button reference</p>
<p>Popping up a message using a dialog box (Props to Craig Ceremuga for this info)</p>	<p>JOptionPane</p>	<p>import javax.swing.*;</p> <p>JOptionPane.showMessageDialog(null, "There is not enough \$\$\$ for this purchase", "Change calculation error", JOptionPane.ERROR_MESSAGE);</p> <p>the general format is</p> <p>JOptionPane.showMessageDialog(null, "Main message in the pop-up", "Title bar of the window", type of dialog desired);</p> <p>//You can also use the following parameters to vary the style of the dialog box that you end up with...</p> <p>PLAIN_MESSAGE – plain dialog box without any icon in it INFORMATION_MESSAGE – icon denoting information QUESTION_MESSAGE – question mark WARNING_MESSAGE – warning icon</p>
<p>Using combo boxes</p>	<p>getSelectedIndex(); getItemAt();</p>	<p>public void actionPerformed(ActionEvent e) { //was this the combo box? if(e.getSource() == topicCB) { //Figure out which topic they selected int a = topicCB.getSelectedIndex();</p>

		<pre> String choice = (String)topicCB.getItemAt(a); } //now choice contains the phrase from the combo box }</pre>
Getting data from a JTextField	getText()	<pre> public class RegisterFrame extends JFrame implements ActionListener { JTextField searchTF = new JTextField(); public void actionPerformed(ActionEvent e) { String temp = searchTF.getText(); //temp now contains whatever was typed into //the text field } }</pre>
Pausing a program for a while(Props to Jameson Ma)	Thread.sleep	<pre> try { Thread.sleep(10); //This is supposed to be for 10 milliseconds but mine is not //working this way } catch(InterruptedException e) { Thread.currentThread().interrupt();} }</pre>
Making a frame not resizable	setResizable()	<pre> //This can be done using the properties on the frame public static void main(String[] args) { ComputerQuiz computerQuiz1 = new ComputerQuiz(); computerQuiz1.setSize(400,400); computerQuiz1.setResizable(false); computerQuiz1.setVisible(true); }</pre>

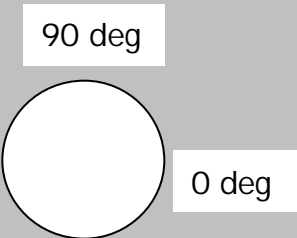
		<pre> }</pre>
Getting in data from a pop up dialog box	showInputDialog	<pre> import javax.swing.*; //to get a String String input; input = JOptionPane.showInputDialog("Please enter a word"); //to get an integer String input = JOptionPane.showInputDialog("Please enter an int); int x = Integer.parseInt(input); //to get a double String input = JOptionPane.showInputDialog("Please enter a double); double y = Double.parseDouble(input);</pre>
Playing wav files from within an application or applet NOTE: wav, midi and au files work fine, mp3's do not with this approach	play, loop and stop	<pre> import java.applet.Applet; import java.applet.AudioClip; public class ConsoleSound { //declaration static AudioClip clip1,clip2; public static void main(String[] args) { //one time to load in from disk clip1=Applet.newAudioClip(ClassLoader.getResource("sounds/move.wav"));</pre>


		<pre> clip2=Applet.newAudioClip(ClassLoader.getResource("sounds/die.wav")); playSound(); //calls the playSound method } public void playSound() { //to play one time clip1.play(); //to loop clip2.loop(); //to stop clip2.stop(); } </pre>
Setting the value of a JTextField with a String variable or double or int variable	setText()	<pre> public class RegisterFrame extends JFrame implements ActionListener { JTextField inputaTF = new JTextField(); JTextField inputbTF = new JTextField(); JTextField sumTF = new JTextField(); public void actionPerformed(ActionEvent e) { String tempa = inputaTF.getText(); //tempa now contains whatever was typed into //the text field String tempb = inputbTF.getText(); //tempb now contains whatever was typed //into the text field double a = Double.parseDouble(tempa); //a now contains whatever number was //typed into the field inputaTF double b = Double.parseDouble(tempb); //a now contains whatever number was //typed into the field inputbTF double sum = a+b; //Now to set the sumTF to the variable sum Double sumText = new Double(sum); sumTF.setText(sumText.toString()); } </pre>

		<pre> } } </pre>
Using ImageIcon	new ImageIcon and setIcon	<pre> public class TestFrame extends JFrame { ImageIcon red = new ImageIcon(ClassLoader.getResource("images/redblock.png")); JLabel myLabel = new JLabel(); public void actionPerformed(ActionEvent e) { myLabel.setIcon(red); } } </pre>
Using checkboxes	isSelected()	<pre> public class TestFrame extends JFrame { JCheckBox includeDepCBX = new JCheckBox(); public void actionPerformed(ActionEvent e) { if (includeDepCBX.isSelected() == true) { //stuff here is } } } </pre>
Using radio buttons and button groups	isSelected(), setSelected(), use JBuilder to add a radio button group	<p>Make sure you use JBuilder to add a button group to your frame, I don't know if it will be listed in the treelist for the frame, but for buttons to act in an exclusive way, they should be part of a button group. Use the property inspector to associate the radio buttons with the group.</p> <p>//see above for isSelected() to test to see if a radio button is selected</p>

<p>How to make a custom icon for your application</p>		<pre>import java.awt.Image; public class StudentInfoFrame extends JFrame implements ActionListener { Image icon = Toolkit.getDefaultToolkit(). getImage (java.net.URLClassLoader.getResource("images/eaglesmini.jpg")); //in the classes directory, create an images directory private void jbInit() throws Exception { //Set the icon image for this application this.setIconImage(icon); ...other stuff } }</pre>
<p>Trimming data and checking for empty</p>		<pre>public void actionPerformed(ActionEvent e) { String temp = bagsTF.getText(); if (temp.trim().equals("")) { JOptionPane.showMessageDialog(null, "Need Data", "Error", JOptionPane.ERROR_MESSAGE); } }</pre>

<p>Set up a frame to use the enter key as well as click a button</p>	<p>Add a keylistener</p>	<pre>//Inside your frame public class JavaOutletFrame extends JFrame implements ActionListener, KeyListener { //Need the following three methods public void keyTyped(KeyEvent e) { if(e.getKeyCode()==KeyEvent.VK_ENTER) processInput(); //design another method you can call from actionPerformed } public void keyPressed(KeyEvent e) { } public void keyReleased(KeyEvent e) { } //method to do logic desired by both clicking a button AND pressing return private void processInput() { //Grab the bags String temp = bagsTF.getText(); if (temp.trim().equals("")) { JOptionPane.showMessageDialog(null, "Need Data", "Error", JOptionPane.ERROR_MESSAGE); more stuff } public void actionPerformed(ActionEvent e) {</pre>
--	--------------------------	--

		<pre> processInput(); //call the method so clicking does some thing as enter key } private void jbInit() throws Exception { //Add keylisteners findBUT.addKeyListener(this); //add for everywhere that might have the focus bagsTF.addKeyListener(this); } </pre>
<p>Section IV: Painting</p>		
<p>Drawing an arc in a graphics context</p>	<p><code>g.fillArc()</code> or <code>g.drawArc</code></p>	<pre> import java.awt.*; public void paint(Graphics g) {  90 deg 0 deg } //This diagram will help you with the starting angle parameter </pre>

		<p>//g.fillArc(x,y,(this is the upper left point of the bounding rectangle, width, height (of bounding rectangle, startAngle, arcAngle (both integers, expressed in degrees)) //The start angle works like this; 0 represents 3 o'clock as shown on the graph above, 90 would be 12 o'clock as shown on the graph above, etc. //The arcAngle is what determines what size arc you get. negative for clockwise, + for counter-clockwise. So if you go -270, you'll go cw 3/4 of the way around g.drawArc(50,50,100,100,0,-90); //gives you something like this</p> 
<p>Drawing text to a graphics window(Props to Sunil Ga\nesh)</p>	<p>drawString</p>	<pre>public void paint(Graphics g) { g.drawString("Hello World",10,100); //10 is x and 100 is y g.drawString(Message + " " + MessageNote,10,200); //Can use String variables also }</pre>
<p>Loading an image for use in an applet (Props to Steve Bozak when painting</p>	<p>getImage</p>	<p>//usually done as a global</p> <p>Image i1 = getImage(getDocumentBase(),"../images/splash.gif"); //NOTE: this assumes that a directory one level up from the class file has been created called images and that the splash.gif file is stored in that directory</p>