Some of the Jcomponents

Java Swing

- Use JScrollBar to create a scroller area.
- Use JToolBar to create a toolbar
- Use JDialog to create a dialog
- Use JEditorPane to create an editable area e.g. text

Scroll Bar

- ScrollBarExample(){

```
    JFrame f= new JFrame("Scrollbar Example");
    JScrollBar s=new JScrollBar();
    s.setBounds(100,100, 50,100);
    f.add(s);
    f.setSize(400,400);
    f.setLayout(null);
    f.setVisible(true);
  }
  public static void main(String args[])
  {
    new ScrollBarExample();
  }}
```

Tool Bar

- JToolBar toolbar = new JToolBar();
  toolbar.setRollover(true);
-
  JButton button = new JButton("File");
  toolbar.add(button);
-
  toolbar.addSeparator();
-
  toolbar.add(new JButton("Edit"));
  toolbar.add(new JComboBox(new String[] { "Opt-1", "Opt-2", "Opt-3", "Opt-4" }));
  Container contentPane = myframe.getContentPane();
  contentPane.add(toolbar, BorderLayout.*NORTH*);

Dialog

- DialogExample() {

```
JFrame f= new JFrame();
d = new JDialog(f , "Dialog Example", true);
d.setLayout( new FlowLayout() );
JButton b = new JButton ("get details");
b.addActionListener ( new ActionListener()
{
    public void actionPerformed( ActionEvent e )
    {
        d.setVisible(false);
    }
});
d.add( new JLabel ("Click for more details"));
d.add(b);
d.setSize(300,300);
d.setVisible(true);
}
```

Edit Pane

```
private void test() {
    myFrame = new JFrame("JEditorPane Test");
    myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    myFrame.setSize(400, 200);

    JEditorPane myPane = new JEditorPane();
    myPane.setContentType("text/plain");
    myPane.setText(“ ");

    myFrame.setContentPane(myPane);
    myFrame.setVisible(true);
}
```

Java JProgressBar

- The JProgressBar class is used to display the progress of the task. It inherits JComponent class.

Commonly used Constructors

Constructor	Description
JProgressBar()	It is used to create a horizontal progress bar but no string text.
JProgressBar(int min, int max)	It is used to create a horizontal progress bar with the specified minimum and maximum value.
JProgressBar(int orient)	It is used to create a progress bar with the specified orientation, it can be either Vertical or Horizontal by using SwingConstants.VERTICAL and SwingConstants.HORIZONTAL constants.
JProgressBar(int orient, int min, int max)	It is used to create a progress bar with the specified orientation, minimum and maximum value.

Commonly used Methods:

Method	Description
void setStringPainted(boolean b)	It is used to determine whether string should be displayed.
void setString(String s)	It is used to set value to the progress string.
void setOrientation(int orientation)	It is used to set the orientation, it may be either vertical or horizontal by using SwingConstants.VERTICAL and SwingConstants.HORIZONTAL constants.
void setValue(int value)	It is used to set the current value on the progress bar.

JFileChooser class

- JFileChooser is a simple and successful method for inciting the client to pick a file or a directory. JFileChooser inherited the properties of JComponent class and implemented them with an Accessible interface.

- Present constructers:
- **1. JFileChooser():**
- Constructs a JFileChooser highlighting the client's default directory.

```
JFileChooser jf = new JFileChooser(); // default constructor JFileChooser is called.
jf.showSaveDialog(null);
```

- **2. JFileChooser(File currentDirectory):**

Constructs a JFileChooser involving the given File as the way.

```
JFileChooser jf = new JFileChooser("c:"); // parameterised constructor JFileChooser( File currentDirectory) is called.
jf.showSaveDialog(null); // opening the saved dialogue
```

Activity

- Use the knowledge you learnt throughout the lectures to Implement the UIs of an app for selling stationaries (2-3 interfaces).
- Hint: use ArrayList to add to the cart.

Thank you