CPS122 - OBJECT-ORIENTED SOFTWARE DEVELOPMENT

Excerpts from an Illustration of the Use of the Factory Pattern in Swing

```
class SwingDemo extends JFrame
{
     ...
     /** Constructor - create a frame containing several buttons, each with a
      *  different look and feel.  The name of the button will indicate what
      *  look and feel it exemplifies.
      */
     public SwingDemo() throws Exception
     {
          getContentPane().setLayout(new GridLayout(0, 1));

          // Each time through this loop, we specify a different look and feel

          for (int i = 0; i < 4; i ++)
          {
               String buttonName = "";

               try
               {
                    switch(i)
                    {
                         case 0:

                                 UIManager.setLookAndFeel(
                                        "apple.laf.AquaLookAndFeel");
                                 buttonName = "This is in the Macintosh look and feel";
                                 break;
                                 break;

                         case 1:

                                 UIManager.setLookAndFeel(
                                        "com.sun.java.swing.plaf.motif.MotifLookAndFeel");
                                 buttonName = "This is in the Motif look and feel";
                                 break;

                         ...
                    }

                    // Create and add a new button with the current look and feel.
                    // The button's name will indicate which look and feel it
                    // exemplifies

                    getContentPane().add(new JButton(buttonName));
               }
               catch(Exception e)
               {
                    System.out.println("Exception " + e);
               }
          }

          pack();
     }
}
```