Applet

Example applet can be browsed at https://github.com/apache/tomee/tree/master/examples/applet

Help us document this example! Click the blue pencil icon in the upper right to edit this page.

Calculator

```java
package org.superbiz.applet;

import javax.ejb.Remote;

@Remote
public interface Calculator {
    public double add(double x, double y);
}
```

CalculatorApplet

```java
package org.superbiz.applet;

import javax.naming.Context;
import javax.naming.InitialContext;
import javax.naming.NamingException;
import javax.rmi.PortableRemoteObject;
import javax.swing.JApplet;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JTextArea;
import javax.swing.JTextField;
import javax.swing.SwingUtilities;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.Properties;

public class CalculatorApplet extends JApplet {
    JTextArea area;

    JTextField field1;
    JTextField field2;
    JLabel label1;
    JLabel label2;
    JButton button;
    JLabel label3;
    Context ctx;
```

```java
public void init() {
    try {
        SwingUtilities.invokeAndWait(new Runnable() {
            public void run() {
                createUI();
            }
        });
    } catch (Exception e) {
        System.err.println("createGUI didn't successfully complete");
    }
}

private void createUI() {
    field1 = new JTextField();
    field2 = new JTextField();
    label1 = new JLabel("Enter first number");
    label2 = new JLabel("Enter second number");
    label3 = new JLabel("RESULT=");
    button = new JButton("Add");

    setLayout(new GridLayout(3, 2));
    add(label1);
    add(field1);
    add(label2);
    add(field2);
    add(button);
    add(label3);
    Properties props = new Properties();
    props.put(Context.INITIAL_CONTEXT_FACTORY,
            "org.apache.openejb.client.RemoteInitialContextFactory");
    props.put(Context.PROVIDER_URL, "http://127.0.0.1:8080/applet/ejb");
    try {
        ctx = new InitialContext(props);
    } catch (NamingException e) {
        throw new RuntimeException(e);
    }
    button.addActionListener(new ActionListener() {

        public void actionPerformed(ActionEvent e) {

            try {
                final Object ref = ctx.lookup("CalculatorImplRemote");
                Calculator calc = (Calculator) PortableRemoteObject.narrow(
                        ref, Calculator.class);
                String text1 = field1.getText();
                String text2 = field2.getText();
                int num1 = Integer.parseInt(text1);
                int num2 = Integer.parseInt(text2);
                double result = calc.add(num1, num2);
                label3.setText("RESULT=" + result);
```

```
                } catch (NamingException ex) {
                    throw new RuntimeException(ex);
                }
            }
        });
    }
}
```

CalculatorImpl

```java
package org.superbiz.applet;

import javax.ejb.Stateless;

@Stateless
public class CalculatorImpl implements Calculator {

    public double add(double x, double y) {
        return x + y;
    }
}
```

web.xml

```xml
<web-app>
  <servlet>
    <servlet-name>ServerServlet</servlet-name>
    <servlet-class>org.apache.openejb.server.httpd.ServerServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>ServerServlet</servlet-name>
    <url-pattern>/ejb/*</url-pattern>
  </servlet-mapping>
</web-app>
```

JNDILookupTest

```java
package org.superbiz;

import org.junit.Assert;
import org.junit.Test;
import org.superbiz.applet.Calculator;

import javax.naming.Context;
import javax.naming.InitialContext;
import javax.rmi.PortableRemoteObject;
import java.util.Properties;


public class JNDILookupTest {

    @Test
    public void test() {
        Properties props = new Properties();
        props.put(Context.INITIAL_CONTEXT_FACTORY,
"org.apache.openejb.client.RemoteInitialContextFactory");
        props.put(Context.PROVIDER_URL, "http://127.0.0.1:8080/tomee/ejb");
        try {
            Context ctx = new InitialContext(props);
            System.out.println("Found context " + ctx);
            final Object ref = ctx.lookup("CalculatorImplRemote");
            Calculator calc = (Calculator) PortableRemoteObject.narrow(ref,
Calculator.class);
            double result = calc.add(10, 30);
            Assert.assertEquals(40, result, 0.5);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```