

```
//Author:      Furrukh Khan  
//Company:    OSU
```

```
package beantest5;
```

```
import java.util.*;
```

```
public class PressureEvent extends EventObject {
```

```
    public PressureEvent(Object source) {  
        super(source);  
    }
```

```
    private double pressure;
```

```
    public double getPressure() {  
        return pressure;  
    }
```

```
    public void setPressure(double newPressure) {  
        pressure = newPressure;  
    }  
}
```

```
//Author:      Furrukh Khan  
//Company:    OSU
```

```
package beantest5;
```

```
import java.util.*;
```

```
public interface PressureListener extends EventListener {
```

```
    public void pressureChanged(PressureEvent e);  
}
```

```
//Author:      Furrukh Khan
//Company:     OSU
```

```
package beantest5;
```

```
import java.io.*;
import java.util.*;
```

```
public class Boiler implements Serializable {
```

```
    public Boiler() {
    }
```

```
    void readObject(ObjectInputStream ois) throws ClassNotFoundException, IOException {
        ois.defaultReadObject();
    }
```

```
    void writeObject(ObjectOutputStream oos) throws IOException {
        oos.defaultWriteObject();
    }
```

```
    public void run() {
        double pressure;
        for(int i = 0; i < 100; ++i) {
            pressure = (double)i;
            if( pressure > getThreshold() ) {
                PressureEvent ev = new PressureEvent(this);
                ev.setPressure(pressure);
                firePressureChanged(ev);
            }
        }
    }
```

```
    public void setThreshold(double newThreshold) {
        threshold = newThreshold;
    }
```

```
    public double getThreshold() {
        return threshold;
    }
```

```
    public synchronized void removePressureListener(PressureListener l) {
        if(pressureListeners != null && pressureListeners.contains(l)) {
            Vector v = (Vector) pressureListeners.clone();
            v.removeElement(l);
            pressureListeners = v;
        }
    }
```

```
    public synchronized void addPressureListener(PressureListener l) {
        Vector v = pressureListeners == null ? new Vector(2) : (Vector) pressureListeners.clone();
        if(!v.contains(l)) {
            v.addElement(l);
            pressureListeners = v;
        }
    }
```

```
}  
private double threshold;  
private transient Vector pressureListeners;  
  
protected void firePressureChanged(PressureEvent e) {  
    if(pressureListeners != null) {  
        Vector listeners = pressureListeners;  
        int count = listeners.size();  
        for (int i = 0; i < count; i++) {  
            ((PressureListener) listeners.elementAt(i)).pressureChanged(e);  
        }  
    }  
}  
}
```

```
//Author:      Furrukh Khan
//Company:     OSU
```

```
package beantest5;
```

```
import java.io.*;
```

```
public class ControlBox implements Serializable {
```

```
    private beantest5.Boiler itsBoiler;
```

```
    public ControlBox() {
        super();
    }
```

```
    void readObject(ObjectInputStream ois) throws ClassNotFoundException, IOException {
        ois.defaultReadObject();
    }
```

```
    void writeObject(ObjectOutputStream oos) throws IOException {
        oos.defaultWriteObject();
    }
```

```
    public void setItsBoiler(beantest5.Boiler newItsBoiler) {
        itsBoiler = newItsBoiler;
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
```

```
    public beantest5.Boiler getItsBoiler() {
        return itsBoiler;
    }
```

```
    private void jbInit() throws Exception {
        itsBoiler.addPressureListener(new beantest5.PressureListener() {
```

```
            public void pressureChanged(PressureEvent e) {
                itsBoiler_pressureChanged(e);
            }
        });
    }
```

```
    void itsBoiler_pressureChanged(PressureEvent e) {
        System.out.println("Threshold Exceeded! Pressure is: "+
            e.getPressure());
    }
}
```

```
//Author:      Furrukh Khan
//Company:     OSU
```

```
package beantest5;
```

```
public class Client {
    Boiler mainBoiler = new Boiler();
    ControlBox mainControlBox = new ControlBox();

    public Client() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }

    public static void main(String[] args) {
        Client client = new Client();
        client.mainBoiler.run();

    }

    private void jbInit() throws Exception {
        mainControlBox.setItsBoiler(mainBoiler);
        mainBoiler.setThreshold(90.0);
    }
}
```