

```
//Title:      EE/CIS 694T
//Version:
//Copyright:  Copyright (c) 2000
//Author:     Furrukh Khan
//Company:    OSU
//Description: Final Solution
```

```
package FinalSolution;
```

```
public class Experiment {
```

```
    public static void main(String[] args) {
```

```
        Tank mainTank = new Tank();
        mainTank.setMinVolume(10.0D);
        mainTank.setMaxVolume(100.0D);
        mainTank.setFullVolume(110.0D);
```

```
        ValveController mainValveController =
            new ValveController();
        mainValveController.setItsTank(mainTank);
```

```
        for(int i = 0; i < 150; ++i) {
            try {
                mainTank.addLiquid();
            }
            catch(TankFullException e) {
                System.out.println(e);
            }
        }
```

```
    }
```

```
}
```

```
/* --- output ---  
VALVE CONTROLLER LOG
```

```
New Volume: 1.0  
New Volume: 2.0  
New Volume: 3.0  
New Volume: 4.0  
New Volume: 5.0  
New Volume: 6.0  
New Volume: 7.0  
New Volume: 8.0  
New Volume: 9.0  
New Volume: 10.0  
New Volume: 11.0  
New Volume: 12.0  
New Volume: 13.0  
New Volume: 14.0  
New Volume: 15.0  
New Volume: 16.0  
New Volume: 17.0  
New Volume: 18.0  
New Volume: 19.0  
New Volume: 20.0  
New Volume: 21.0  
New Volume: 22.0  
New Volume: 23.0  
New Volume: 24.0  
New Volume: 25.0  
New Volume: 26.0  
New Volume: 27.0  
New Volume: 28.0  
New Volume: 29.0  
New Volume: 30.0  
New Volume: 31.0  
New Volume: 32.0  
New Volume: 33.0  
New Volume: 34.0  
New Volume: 35.0  
New Volume: 36.0  
New Volume: 37.0  
New Volume: 38.0  
New Volume: 39.0  
New Volume: 40.0  
New Volume: 41.0  
New Volume: 42.0  
New Volume: 43.0  
New Volume: 44.0  
New Volume: 45.0  
New Volume: 46.0  
New Volume: 47.0  
New Volume: 48.0  
New Volume: 49.0  
New Volume: 50.0  
New Volume: 51.0  
New Volume: 52.0  
New Volume: 53.0  
New Volume: 54.0  
New Volume: 55.0  
New Volume: 56.0  
New Volume: 57.0  
New Volume: 58.0  
New Volume: 59.0  
New Volume: 60.0  
New Volume: 61.0  
New Volume: 62.0  
New Volume: 63.0  
New Volume: 64.0  
New Volume: 65.0  
New Volume: 66.0  
New Volume: 67.0  
New Volume: 68.0
```

New Volume: 69.0  
New Volume: 70.0  
New Volume: 71.0  
New Volume: 72.0  
New Volume: 73.0  
New Volume: 74.0  
New Volume: 75.0  
New Volume: 76.0  
New Volume: 77.0  
New Volume: 78.0  
New Volume: 79.0  
New Volume: 80.0  
New Volume: 81.0  
New Volume: 82.0  
New Volume: 83.0  
New Volume: 84.0  
New Volume: 85.0  
New Volume: 86.0  
New Volume: 87.0  
New Volume: 88.0  
New Volume: 89.0  
New Volume: 90.0  
New Volume: 91.0  
New Volume: 92.0  
New Volume: 93.0  
New Volume: 94.0  
New Volume: 95.0  
New Volume: 96.0  
New Volume: 97.0  
New Volume: 98.0  
New Volume: 99.0  
New Volume: 100.0  
Draining Tank ...  
New Volume: 10.0  
New Volume: 11.0  
New Volume: 12.0  
New Volume: 13.0  
New Volume: 14.0  
New Volume: 15.0  
New Volume: 16.0  
New Volume: 17.0  
New Volume: 18.0  
New Volume: 19.0  
New Volume: 20.0  
New Volume: 21.0  
New Volume: 22.0  
New Volume: 23.0  
New Volume: 24.0  
New Volume: 25.0  
New Volume: 26.0  
New Volume: 27.0  
New Volume: 28.0  
New Volume: 29.0  
New Volume: 30.0  
New Volume: 31.0  
New Volume: 32.0  
New Volume: 33.0  
New Volume: 34.0  
New Volume: 35.0  
New Volume: 36.0  
New Volume: 37.0  
New Volume: 38.0  
New Volume: 39.0  
New Volume: 40.0  
New Volume: 41.0  
New Volume: 42.0  
New Volume: 43.0  
New Volume: 44.0  
New Volume: 45.0  
New Volume: 46.0  
New Volume: 47.0  
New Volume: 48.0  
New Volume: 49.0  
New Volume: 50.0

```
New Volume: 51.0
New Volume: 52.0
New Volume: 53.0
New Volume: 54.0
New Volume: 55.0
New Volume: 56.0
New Volume: 57.0
New Volume: 58.0
New Volume: 59.0
New Volume: 60.0
--- end output --- */
```

```
//Title:      EE/CIS 694T
//Version:
//Copyright:  Copyright (c) 2000
//Author:     Furrukh Khan
//Company:    OSU
//Description: Final Solution
```

```
package FinalSolution;
```

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

```
public class ValveController implements Serializable {
    Tank itsTank = new Tank();
```

```
    public ValveController() {
        super();
        System.out.println("VALVE CONTROLLER LOG\n");
    }
```

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

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

```
    private void jbInit() throws Exception {
        itsTank.addtankListener(new FinalSolution.tankListener() {
```

```
            public void volumeChanges(tankEvent e) {
            }
```

```
            public void maxVolumeExceeded(tankEvent e) {
                itsTank_maxVolumeExceeded(e);
            }
        });
```

```
        itsTank.addtankListener(new FinalSolution.tankListener() {
```

```
            public void volumeChanges(tankEvent e) {
                itsTank_volumeChanges(e);
            }
```

```
            public void maxVolumeExceeded(tankEvent e) {
            }
        });
    }
```

```
    void itsTank_volumeChanges(tankEvent e) {
        double newVolume = e.getVolume();
        System.out.println("New Volume: " + newVolume );
    }
```

```
    void itsTank_maxVolumeExceeded(tankEvent e) {
        Tank tank = (Tank)e.getSource();
        System.out.println("Draining Tank ... ");
        tank.drain();
    }
```

```
    public void setItsTank(FinalSolution.Tank newItsTank) {
        itsTank = newItsTank;
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
```

```
    public FinalSolution.Tank getItsTank() {
        return itsTank; }
}
```

```
//Title:      EE/CIS 694T
//Version:
//Copyright:  Copyright (c) 2000
//Author:     Furrukh Khan
//Company:    OSU
//Description: Final Solution
```

```
package FinalSolution;
```

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

```
public class Tank implements Serializable {
```

```
    private transient Vector tankListeners;
    private double minVolume;
    private double maxVolume;
    private double fullVolume;
    private double volume;
```

```
    public Tank() {
        volume = 0.0D;
    }
```

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

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

```
    // add one gallon
```

```
    public void addLiquid() throws TankFullException{
```

```
        if( volume < fullVolume ) {
            volume = volume + 1.0D;
            tankEvent ev = new tankEvent(this);
            ev.setVolume(volume);
            fireVolumeChanges(ev);
            if(volume >= maxVolume ) {
                fireMaxVolumeExceeded(ev);
            }
        }
        else {
            throw new TankFullException("Tank is full, and shut down!");
        }
    }
```

```
}
```

```
    public void drain() {
```

```
        volume = minVolume;
        tankEvent ev = new tankEvent(this);
        ev.setVolume(volume);
        fireVolumeChanges(ev);
    }
```

```
}
```

```
    public synchronized void removetankListener(tankListener l) {
```

```
        if(tankListeners != null && tankListeners.contains(l)) {
            Vector v = (Vector) tankListeners.clone();
            v.removeElement(l);
            tankListeners = v;
        }
    }
```

```
}
```

```
    public synchronized void addtankListener(tankListener l) {
```

```
        Vector v = tankListeners == null ? new Vector(2) : (Vector) tankListeners.clone();
        if(!v.contains(l)) {
            v.addElement(l);
            tankListeners = v;
        }
    }
```

```
}
```

```

}

protected void fireVolumeChanges(tankEvent e) {
    if(tankListeners != null) {
        Vector listeners = tankListeners;
        int count = listeners.size();
        for (int i = 0; i < count; i++) {
            ((tankListener) listeners.elementAt(i)).volumeChanges(e);
        }
    }
}

protected void fireMaxVolumeExceeded(tankEvent e) {
    if(tankListeners != null) {
        Vector listeners = tankListeners;
        int count = listeners.size();
        for (int i = 0; i < count; i++) {
            ((tankListener) listeners.elementAt(i)).maxVolumeExceeded(e);
        }
    }
}

public void setMinVolume(double newMinVolume) {
    minVolume = newMinVolume;
}

public double getMinVolume() {
    return minVolume;
}

public void setMaxVolume(double newMaxVolume) {
    maxVolume = newMaxVolume;
}

public double getMaxVolume() {
    return maxVolume;
}

public void setFullVolume(double newFullVolume) {
    fullVolume = newFullVolume;
}

public double getFullVolume() {
    return fullVolume;
}

public double getVolume() {
    return volume;
}
}

```

```
//Title:      EE/CIS 694T
//Version:
//Copyright:  Copyright (c) 2000
//Author:     Furrukh Khan
//Company:    OSU
//Description: Final Solution
```

```
package FinalSolution;
```

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

```
public interface tankListener extends EventListener {
```

```
    public void volumeChanges(tankEvent e);
```

```
    public void maxVolumeExceeded(tankEvent e);
```

```
}
```

```
//Title:      EE/CIS 694T
//Version:
//Copyright:  Copyright (c) 2000
//Author:     Furrukh Khan
//Company:    OSU
//Description: Final Solution
```

```
package FinalSolution;
```

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

```
public class tankEvent extends EventObject {
```

```
    private double volume;
```

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

```
    public double getVolume() {
        return volume;
    }
```

```
    public void setVolume(double newVolume) {
        volume = newVolume;
    }
```

```
}
```

```
//Title:      EE/CIS 694T
//Version:
//Copyright:  Copyright (c) 2000
//Author:     Furrukh Khan
//Company:    OSU
//Description: Final Solution
```

```
package FinalSolution;
```

```
public class TankFullException extends Exception {
```

```
    public TankFullException(String str) {
        super(str);
    }
```

```
}
```