Lec 3 Wed 1/24/01

- please read chap 1-4 & learn JBuilder by Mon.
- then chap 25 in Deitel.

assignment 1: 4.7 p 157.

start = ϕ

g.drawCircle(50, 50, 200, 200);
g.fillArc(50, 50, 200, 200, 90, 15);
start ↑ span ↑
" " (" " " " , start+90, 15);
g.drawRect(50, 50, 200, 200); // makes // rectangle.

start = start + 5
start = start/360;

Multithreading review in this exercise.

setStart
getStart

properties
start (T/F) boolean
T
F ← reverse (T/F) boolean
200 ← speed (int)
↳ sleep
jump

public class Fan
{

Lec 3 Wed 1/24/01

```
public class Fan
{

    public void run()
    {

    }
```

- make the fan work, then drop into a heap & make button work with it.

Required:
1) public class
2) parameterless constructor
3) serializable

Optional:
4) get/set methods
5) for an event source
   addXXXListener
   removeXXXListener

UI designer, events critical.

read p 50 in class.
JButton btl = new JButton();

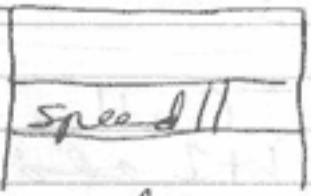
constructor rule: Specify all of the constructors or none of the constructors
→ if one specified then no default there.

Lec 3. Wed 1/24/01

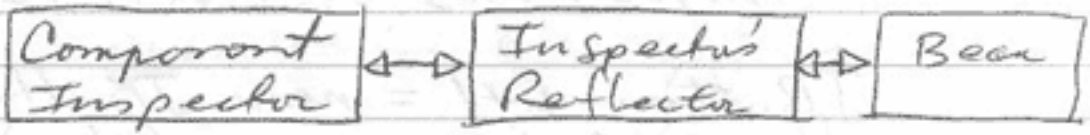
- public accessor methods.

```
public int getSpeed() { }
public void setSpeed(int s) { }
```

get —
set



JBuilder sees class and allows properties to be modified but not the java code.

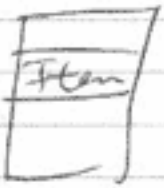
056. see 

- Bean is completely independent at runtime.

We must master events & listeners in chapter 3.

addActionListener()
removeActionListener()

addItemListener
remove



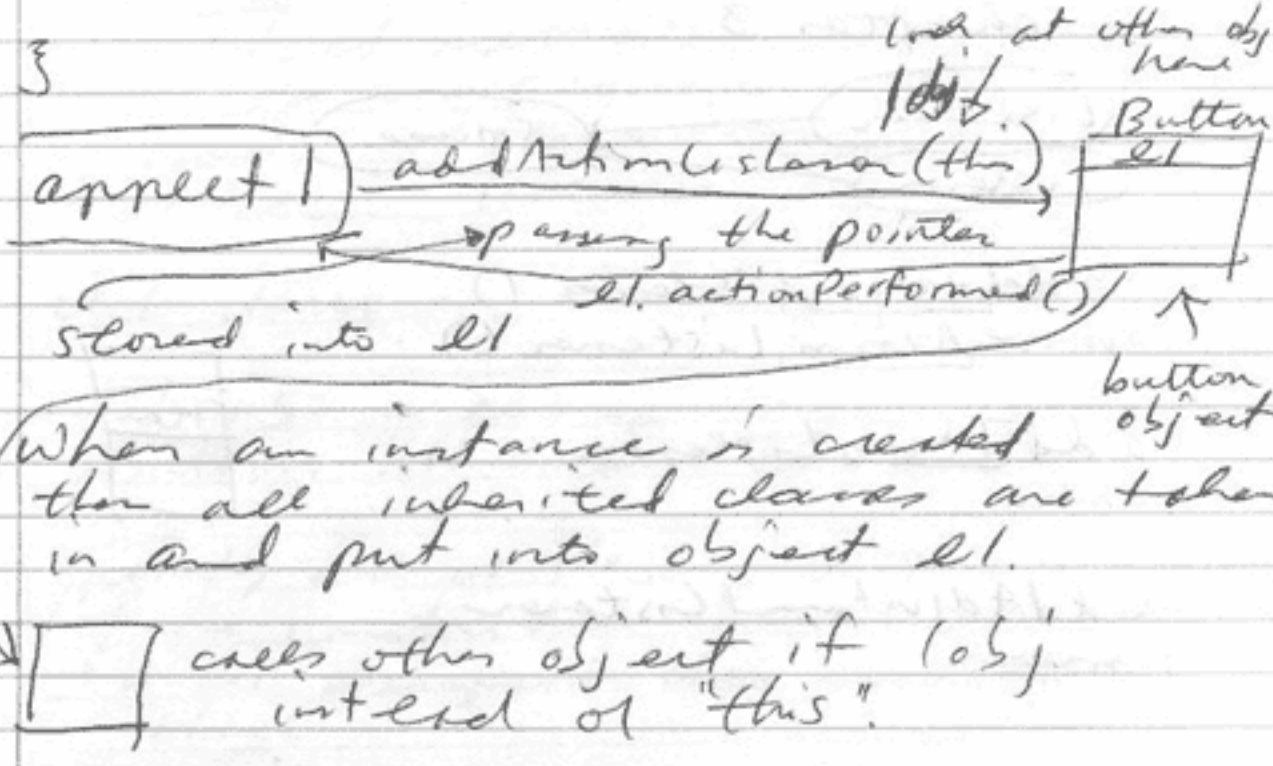
addAdjustmentListener
remove

Lec 3 Wed 1/24/01

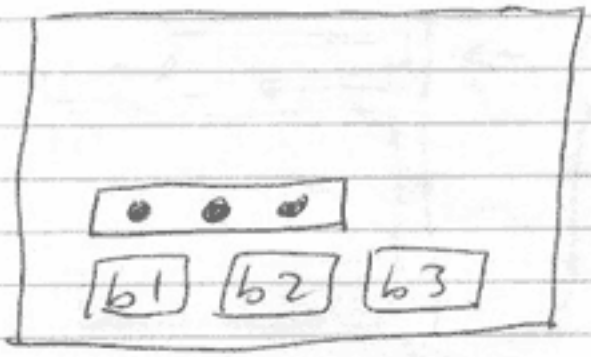
```
public void addActionListener
                (ActionListener obj)
{
}
```

```
JButton btn = new JButton("click");
bt1.addActionListener(this);
```

```
public class MyApplet extends
                JApplet
    implements ActionListener
{
    JButton bt1 = new JButton("click");
    bt1.addActionListener(this);
}
public void ActionPerformed(ActionEvent
                e)
{
}
```



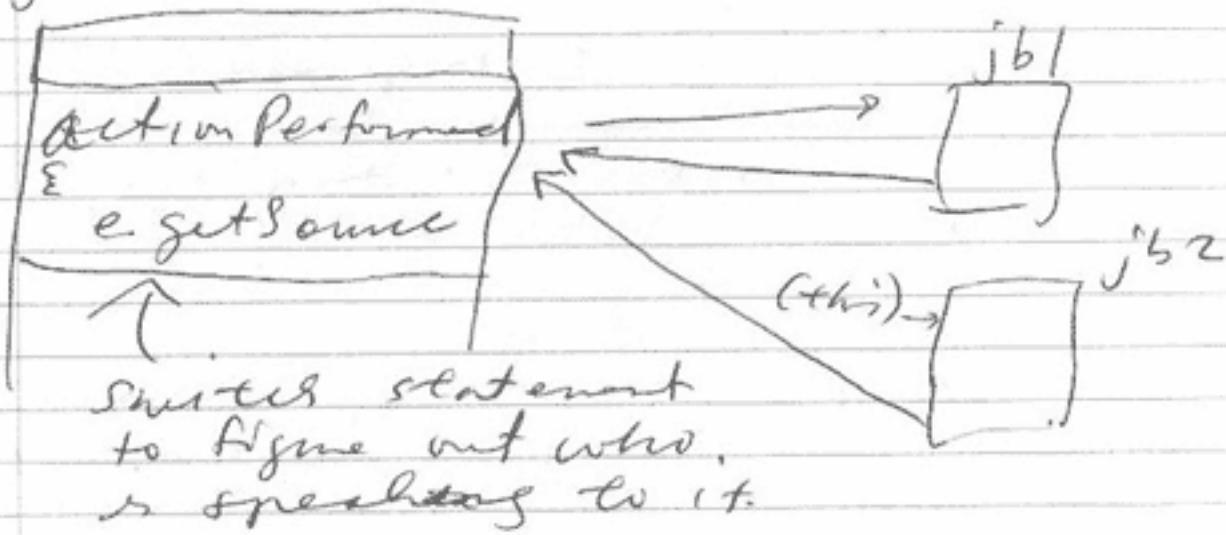
Lec 3 wed 1/24/01




```
JButton jb1 = new JButton ("b1");
        jb2                ("b2");
        jb3                ("b3");

getContentPane().add(jb1);
                    (jb2)
                    (jb3)

// register
jb1.addActionListener(this);   // the object that has called back function.
jb2.    "          "    "
jb3.    "          "    "
```



hello to have separate listeners