

Forum:[Hack Forums](#)

Thread: [Inner classes and anonymous classes](#)

Traitor-Panda - June 11, 2012, 3:01 pm

I'm having trouble understanding inner and anonymous classes.
Can anyone show me a simple example of it's usage?
Would appreciate it!

novados24 - June 11, 2012, 3:50 pm

A common inner class use is an actionlistener for a button. Most of the time you won't want to start another text file just for a simple listener on one button. Instead, at the bottom of your class you would make another class like:

```
Code private class MyListener implements ActionListener{
    public void actionPerformed(ActionEvent e){
        System.out.println("pressed");
    }
}
```

This goes within the brackets of the other class but not in a method. If you have very complex listeners that handle alot of buttons you'd probably be better off making a whole new text file but for very simple things like this is when it is commonly used

frodo - June 11, 2012, 4:27 pm

(Jun 11, 2012 03:50 PM)novados24 Wrote: [\[CITE\]](#) A common inner class use is an actionlistener for a button. Most of the time you won't want to start another text file just for a simple listener on one button. Instead, at the bottom of your class you would make another class like:

```
Code private class MyListener implements ActionListener{
    public void actionPerformed(ActionEvent e){
        System.out.println("pressed");
    }
}
```

This goes within the brackets of the other class but not in a method. If you have very complex listeners that handle alot of buttons you'd probably be better off making a whole new text file but for very simple things like this is when it is commonly used

expanding on this, anonymous classes can be used to accomplish the same goal in a similar way:

```
Code someButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        System.out.println("button pressed");
    }
});
```

irrationalByte - June 12, 2012, 12:43 am

How about if you have have been given a project to make an Employee class?
Its easy,isn't it?

```
Code public class Employee
{
    int sallary,DA,HRA,TA;
```

```
String name;  
int ID;  
//other stuff etc..  
//then you make your functions  
void input()  
{  
  //something  
}  
void show()  
{  
  //something  
}  
//Then your main function  
public static void main(String args[])  
{  
}  
}
```

I'll be willing to help you, if you can figure out the error in this program.
Hint: It's not a compiling error but it voids a definition.