

## RAGE

## Input Devices / Modifier Actions

**this RAGE game has an Action that modifies another Action :***... imports same as before*

```
public class MyGame extends VariableFrameRateGame
{
    ... class variables as before, plus an additional Action class:

    private InputManager im;
    private Action quitGameAction, incrementCounterAction,
                                   incAmtModAct ;

    public MyGame()
    { super();
      System.out.println("press C or button 3 to increment counter");
      System.out.println("press button 4 to change increment amount");
      System.out.println("press ESC or button 9 to quit");
    }
}
```

*... setupWindows(), setupScene(), same as before*

```
protected void setupInputs()
{ im = new GenericInputManager();
  String kbName = im.getKeyboardName();
  String gpName = im.getFirstGamepadName();

  // build the action objects
  quitGameAction = new QuitGameAction(this);
  incAmtModAct = new IncrementAmountModifierAction(this);

  // the modifier action is now part of the constructor for the
  // increment action:
  incrementCounterAction = new IncrementCounterAction(this,
    (IncrementAmountModifierAction) incAmtModAct);
}
```

*... associateAction calls as before, plus the following additional:*

```
im.associateAction(gpName,
  net.java.games.input.Component.Identifier.Button._4,
  incAmtModAct,
  InputManager.INPUT_ACTION_TYPE.ON_PRESS_ONLY);
...
}
```

```
protected void incrementCounter(int amt) { counter += amt; }
... note this was slightly changed
```

*main() and update() are unchanged.***revised Action class – now includes a modifier action:**

```
import ray.input.action.AbstractInputAction;
import ray.rage.game.*;
import net.java.games.input.Event;

public class IncrementCounterAction extends AbstractInputAction
{
    private MyGame game;

    // note that this class now has a pointer to the modifier action:
    private IncrementAmountModifierAction incAmtModAct;

    public IncrementCounterAction(MyGame g,
                                   IncrementAmountModifierAction modAct)
    { game = g;
      incAmtModAct = modAct;
    }

    public void performAction(float time, Event e)
    { System.out.println("counter action initiated");
      int incAmt = incAmtModAct.getIncAmt(); // it uses the modifier here
      game.incrementCounter(incAmt);
    }
}
```

**a Modifier Action class used by the above Action class**

```
import ray.input.action.AbstractInputAction;
import ray.rage.game.*;
import net.java.games.input.Event;

public class IncrementAmountModifierAction extends AbstractInputAction
{
    private MyGame game;
    private int incAmt = 1;

    public IncrementAmountModifierAction(MyGame g)
    { game = g;
    }

    public void performAction(float time, Event e)
    { System.out.println("modifier action initiated");
      incAmt++;
      if (incAmt == 5) incAmt=1;
    }

    protected int getIncAmt() { return incAmt; }
}
```

*The QuitAction class is unchanged.*