JInput
Getting Controller & Component Information

```
// uses JInput library
import net.java.games.input.Controller;
import net.java.games.input.ControllerEnvironment;
import net.java.games.input.Version;
import net.java.games.input.Component;

public class FindComponents
{
  public void listControllers()
  {
    System.out.println("JInput version: " + Version.getVersion());
    ControllerEnvironment ce =
                          ControllerEnvironment.getDefaultEnvironment();

    // get the set of controllers from the controller environment

    Controller[] cs = ce.getControllers();

    // print details and sub-controllers for each of the controllers

    for (int i=0; i < cs.length; i++)
    {
      System.out.println("\nController #" + i);
      listComponents(cs[i]);
    }
  }

  // Report the component information for a controller.
  // Recursively visit any subcontrollers and report their details as well.

  private void listComponents(Controller contr)
  {
    System.out.println ("Name: '" + contr.getName()
                            + "'.  Type ID:" + contr.getType());

    // get the components in the controller, and list their details

    Component [] comps = contr.getComponents();
    for (int i=0; i < comps.length; i++)
      System.out.println ("  name: " + comps[i].getName()
                              + "  ID: " + comps[i].getIdentifier() );

    // find subcontrollers, if any, and recursively list their details too

    Controller[] subCtrls = contr.getControllers();
    for (int j=0; j < subCtrls.length; j++)
    {
      System.out.println("  " + contr.getName() + " subcontroller #" + j);
      listComponents(subCtrls[j]);
    }
  }

  public static void main(String[] args)
  {  FindComponents f = new FindComponents();
     f.listControllers();
  }
}
```