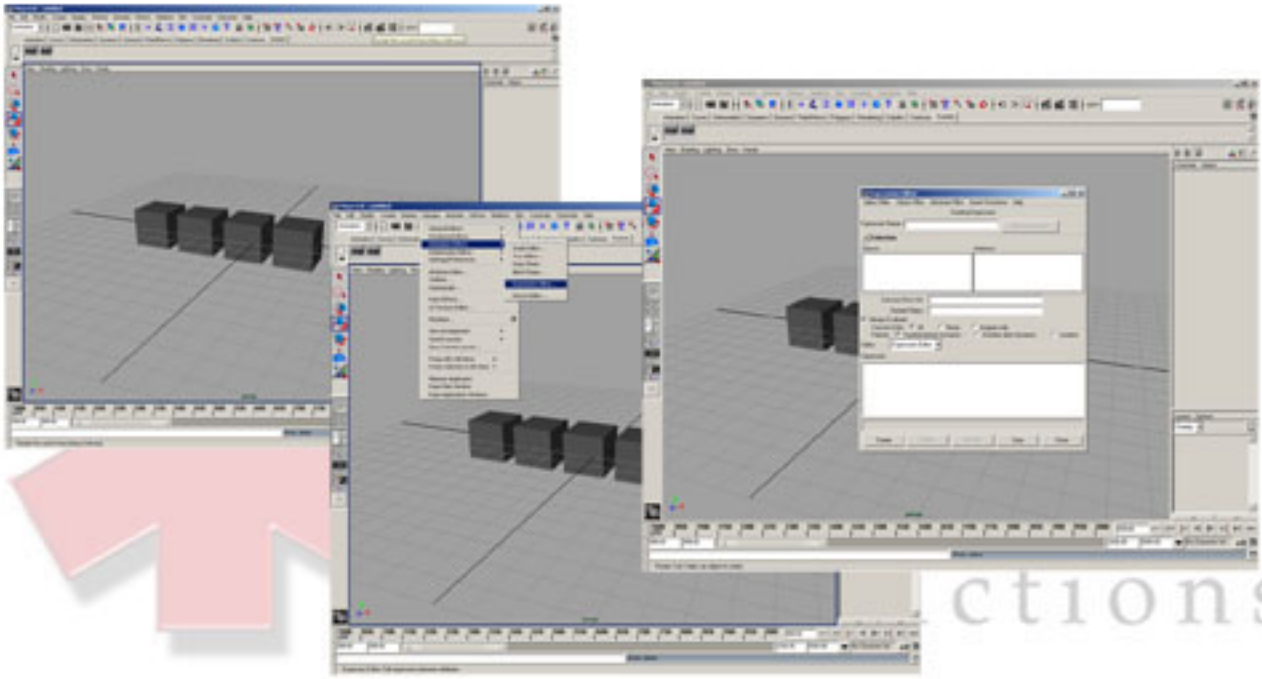
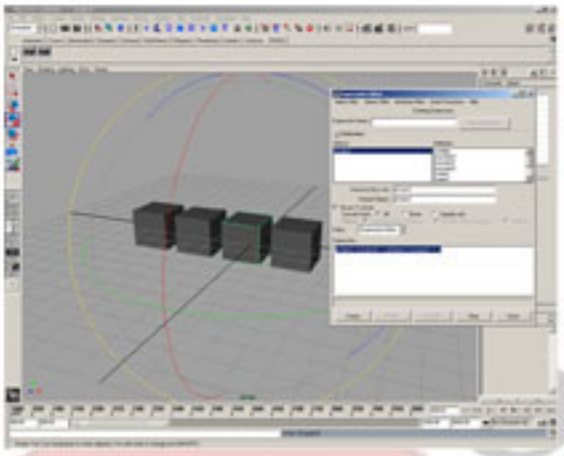
A Very simple Mel Script



Mel scripting...ahh I can hear the cry of babies at the mere sound of the work. Luckily for you it is not as bad as most people make it out to be. It is acctually quite simple. Given a little time, polish up those old math skills and learn a little programing and you will be a pro.



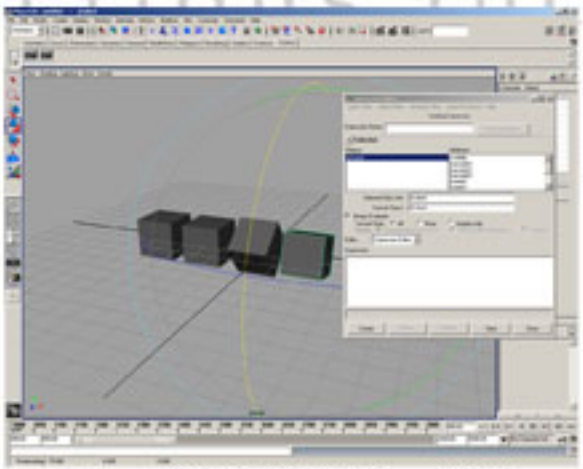
First off we will start with a one line Mell command. When we turn our main block the block next to it will turn half the ammount of the first.

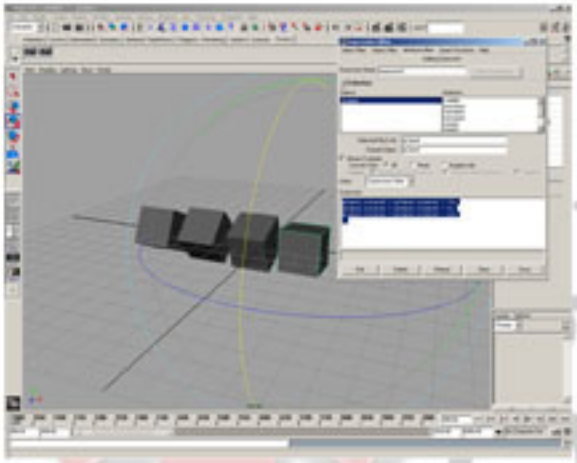
```
pCube3.rotateX = pCube4.rotateX *.5;
```

Now for those of you out there that are thinking WTF, well tough $hit. Naw I'm just joshin ya.

Here is the step by step process. Not only the step by step but a descrition of what each step does.

1. Go to Window/AnimationEditors/ExpressionEditors
2. Select the object that is going to be driven. Mine was pCube3. I was using pCube4 as the driver.
3. We wanted the pCube 3 to move half the distance on the X acess then pCube4.
4. Once you have your driven object selected in the Expression editor window select rotateX.
5. In the "Selected Object and Attribute" pCube3.rotateX should be displayed. This describes what you have selected. The X rotate on the pCube3.
6. Type or Copy and Paste it into the Expression window. This gives you your first half of the equation.
7. Now we need to look at what we want it to do. We want it to move half of the amount of pCube4. Now you can do this one of two ways. We can divide (/) it by 2 (/2) or we can multiply (*) it by .5 (*.5). Either way works just fine.
8. So....we need pCube3's value to equal half of pCube4's value. We look at this sentance and the equation is right there in front of us.
   "pCube3.rotateX = pCube.rotatex *.5;" or
   "pCube3.rotateX = pCube4.rotateX /2;"
9. Now give your driver, pCube4 a turn on the X axis. What should happen is as you turn pCube4, pCube3 turns half the distance.





Now you can layer this tecnique and use it to control a whole array of objects. I took all four cubes and gave them falling off values. Each one turns a little les then the one before it.

```
pCube1.rotateX = pCube4.rotateX *.75;
pCube2.rotateX = pCube4.rotateX *.5;
pCube3.rotateX = pCube4.rotateX *.25;
```

This should turn pCube3 25% of pCube4, pCube2 at 50% and so and so forth. This simple mel command can be used to create complex things.

In the skeleton Rig I used it to allow the two forarm bones to move as they would in reality. In order for the forearm to turn at the wrist the bones need to almost overlap. This was accomplished by setting up four bones in the forearm to the wrist and writing in the mel command above. Then I skinned the two arm bones to just the four rig bones to give them the ability of twisting over eachother.