CASE – SYNOPERATOR

package myentitites

import case.lang.System

namespace EntitiesNameSpace {

String->Object->Main

#public class Program

[public Program(String [] args)

[EntityPool Pool = EntityPool.getEntityPool]

assert(Pool) //asserts that Pool exists and has a value

Stream (n) String

Int MyInt = EntityPool.getStreamMemory() //retrieve mem from pool

Int GetInt = EntityPool.get(“MyInt”) //pointer to MyInt using pool get

//get pointer to CurrentLocationInList from the pool

Int ListStatus = n.get(“CurrentLocationInList”)

//output

//Often times in some part of the code an event will occur. Reactions //allow for a callback function for custom eventes across the board. //Instead of wiring an event to an event queue, etc. **reactions** provide //us with an automatic callback function when the event occuring //doesn’t even know about the reactions.

]

//Sample Code:

//somewhere embedded in the code the following runs:

[public void function(Integer ScreenResolution\_x,

Integer ScreenResolution\_y)

…some code…

//new screen resolution

ScreenResolution\_x = 800

ScreenResolution\_y = 600

]

[**callbyreaction** ScreenResolution\_x && **callbyreaction** ScreenReseolution\_y

//activity has been performed on the screen resolution

//we want to see what the new resolution is

Print ScreenResolution\_x

Print ScreenResolution\_y

//maybe change them for some reason

//inside callbyreaction they are in scope

ScreenResolution\_x = 1080

ScreenResolution\_y = 780

]

]

[ virtual public pickFruit()

….

]

#endclass