CASE PROGRAMS – Object Orientation

package orientation

namespace orientationnamespae

{

String->Object->Main

#public class Program

[public Program(String [] args)

[EntityPool Pool = EntityPool.getEntityPool]

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

//classes are for instantiating types

//CASE contains built in types like Int

//to use a non-built in type declare it as a a type

///then use it like Int for example

//**<**TypeName> ClassName myClass = new <>()

<ListDemo> AdvancedList = new <>()

stream (v) AdvancedList

MyList (v) AdvancedList

]

//class function

[public void meta myBooFunction()

Print “Boo!”

]

[public void inputFunction(function Program.myBooFunction())

myBooFunction()

]

//no boxing

[public void translateFunction(Int b)

//instead of Integer.parseInt(mystring) we can say…

//since we are using the root class

Int c = myString //casting myString down to Int

Int b = a //say we wanted to cast float down to Int

]

//An even better example is say I have an object Cat and an object Dog. We can do //the following: Cat = Dog or Dog = Cat , that is Dog is assigned to Cat and vice //versa.

//inner class

#public class HalloweenGear

[public void purchase()]

[public void sell()]

[public void wear()]

#end class

#end class

}