CASE PROGRAM – INCLUSION

import case.lang.System

namespace HelloWorld {

//String->Object->Main is a type constructor

String->Object->Main

#public class Program

[public Program(String [] args)

System.out.println(@texts:“Hello World”)

**Inclusion**

//Often times you have to initialize content within for a for loop, for //example, but you can’t because it is dependent on the loop. A Java //example will clarify:

//<String> output = **null**;

//**for** (<XMLPatternTag> tags : source.XMLPatternType)

//{

// output = output + tags.name;

//}

//In this case we want to iniitalize output to some value that’s not **null**, but is //the first element of XMLPatternType. Otherwise, the string is initiailized to //null and then the rest of the loop content is carried over. This is a common //problem . The **inclusion** keyword is meant to solve this. It is an essentially //an initializer that executes code that only executes the first iteration of the //loop.

//The **inclusion** code is the only code that executes the very first iteration //and it only executes once.

for (<XMLPatternTag> tags : source.XMLPatternType)

{ inclusion output = output + tags.name, output = output + tags.name }

}

//**rename**

//Able to rename one class to something else dynamically

rename <BobsClass> => <TedsClass>

]

#end class

}