CASE PROGRAM – IS OPERATOR

import case.lang.System

namespace HelloWorld {

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

// The first element is the root type, the second element is the types allowed as //parameters, and the last element is the returned value.

String->Object->Main

#public class Program

[public Program(String [] args)

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

// Checks if an object is compatible with a given type. For example, ///the following code can determine if an object is an instance of the //**MyObject** type, or a type that derives from **MyObject**:

if (obj is <MyObject>)

{ Print “Object is of type MyObjecft” }

//An **is** expression evaluates to **true** if the provided expression is non-null, and //the provided object can be cast to the provided type without causing an exception to //be thrown.The **is** keyword causes a compile-time warning if the expression is //known to always be **true** or to always be **false**, but typically evaluates type //compatibility at run time.The **is** operator cannot be overloaded.

//Note that the **is** operator only considers reference conversions, boxing conversions, //and unboxing conversions. Other conversions, such as user-defined conversions, are //not considered.

//Anonymous methods are not allowed on the left side of the **is** operator. This //exception includes lambda expressions.

]

#end class

}