

Guía Dolphin Smalltalk 7

UTN FRLP

¿Qué es Dolphin?

- Es un entorno de desarrollo integrado para smalltalk de código abierto y adaptado para sistemas Windows modernos tanto de 32 como 64 bits.

Ventajas frente a Smalltalk Express:

- Moderno, soporta hasta sistemas operativos Windows de 64bits.
- Interfaz amigable.
- Auto-completado de código.
- No precisa instalar una máquina virtual.
- Ventana de errores más amigable al usuario.



Desventajas:

- No tiene soporte para SO distintos de Windows.
- Algunas diferencias de sintaxis e impresión.

Diferencias con Smalltalk Express

Sintaxis:

- En cuanto a la escritura del lenguaje, se encuentran algunas diferencias, principalmente en los nombres de algunas clases, por lo tanto es necesario conocerlas para poder realizar correctamente las prácticas. Sin embargo es conveniente acordarse las expresiones de Smalltalk Express para los parciales y/o finales, ya que el mismo sigue siendo el oficial de la cátedra.

Express	Dolphin
MessageBox message:'x'.	MessageBox notify:'x'.
Prompter prompt: 'x' default:'x'.	Prompter prompt: 'x' .
'x' asInteger	'x' asNumber
val:=23 asString.	val:=23 displayString.
2/3 inspect.	(2/3) inspect. "fracciones"



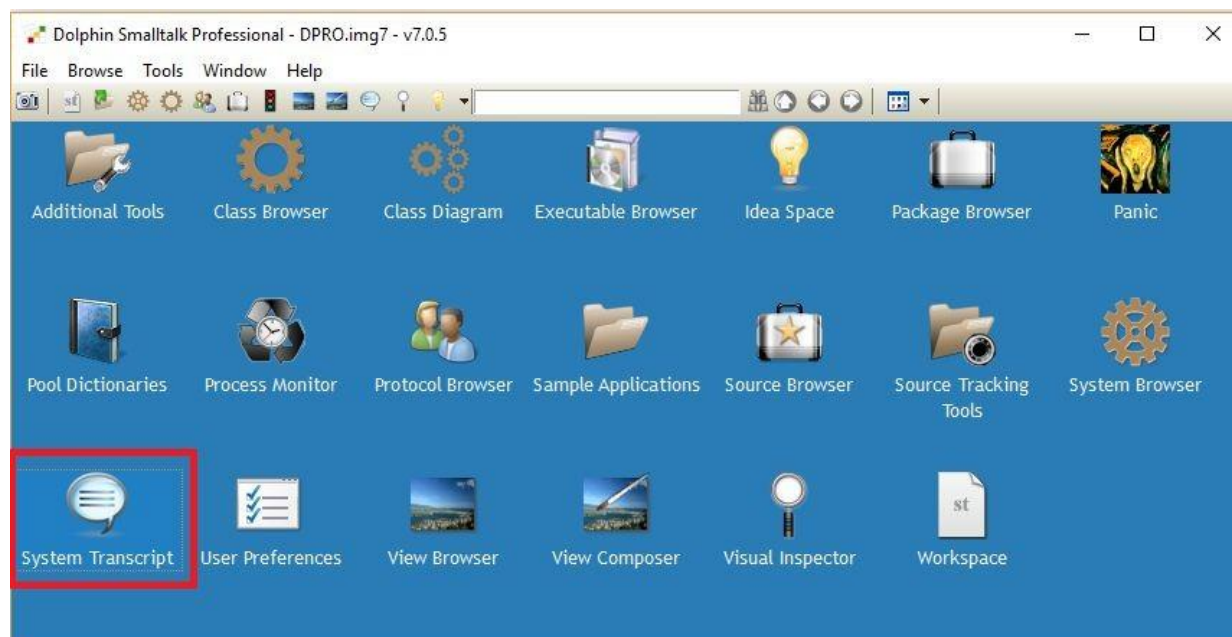
Impresión:

Para mostrar mensajes en pantalla de Transcript se utilizan métodos propios de la clase ventana. Sin embargo solo puede mostrar objetos de la clase String. Por lo tanto para imprimir cualquier objeto (incluyendo variables) que no sean de la clase String debemos agregar "printString" o "displayString" al final.

El método `unObj printOn: Transcript`, se puede usar pero no siempre imprime en la ventana del sistema (queda en buffer) . En general muestra los resultados en el workspace. Si se combina una impresión con `printOn:` con un `show:` muestra en la ventana del sistema. Recomendamos usar los siguientes métodos:

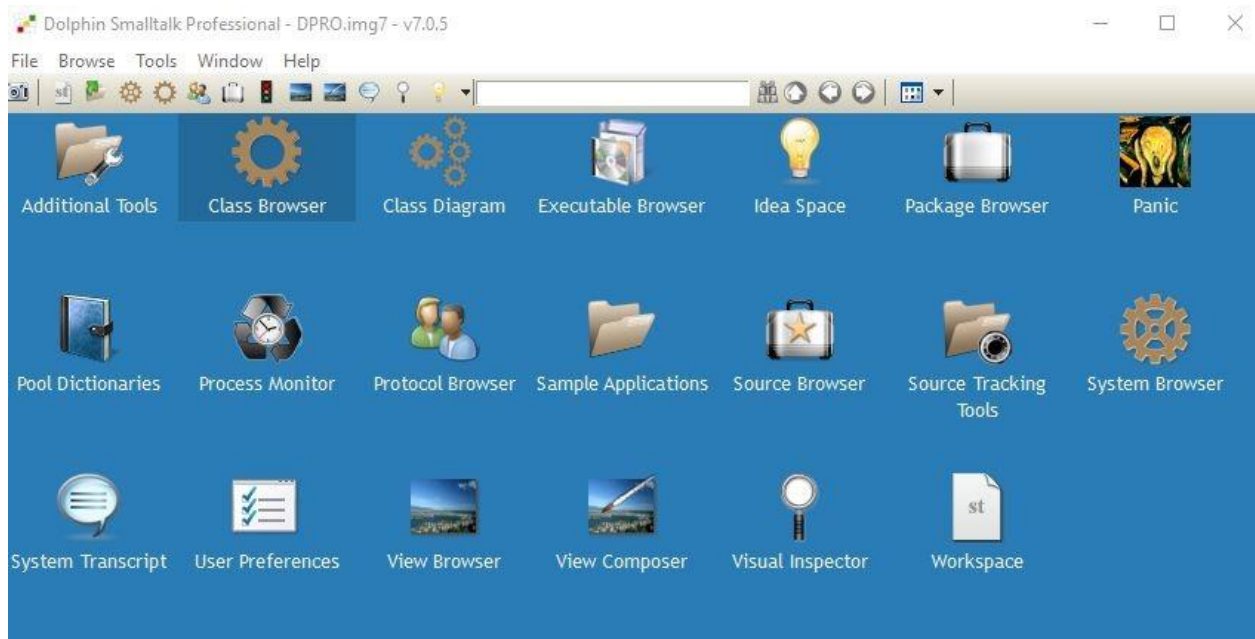
Ejemplos
Transcript show: 'hola'.
Transcript show: '47'.
Transcript show: 47 printString.
Transcript nextPutAll: ' La temperatura es de ', 25 displayString, ' grados'.
a a:= 47. Transcript show: a printString. Transcript show: a displayString.

Para abrir la ventana de Transcript clickeamos en el ícono o vamos a **Tools>System Transcript**

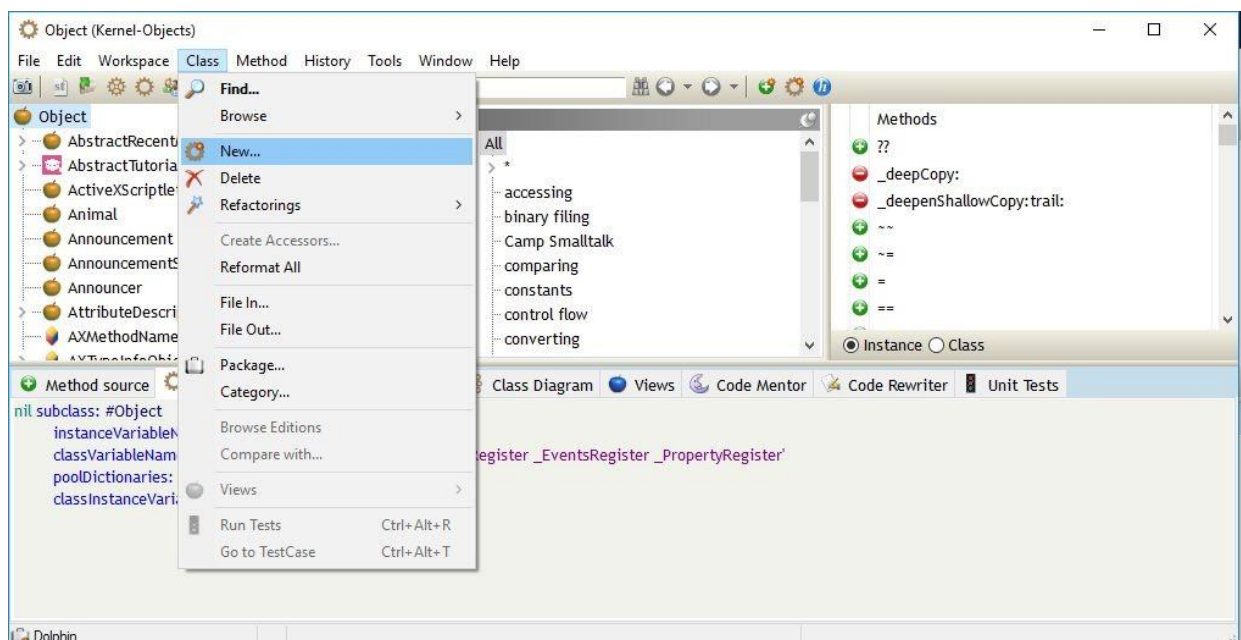


¿Cómo crear clases y métodos?

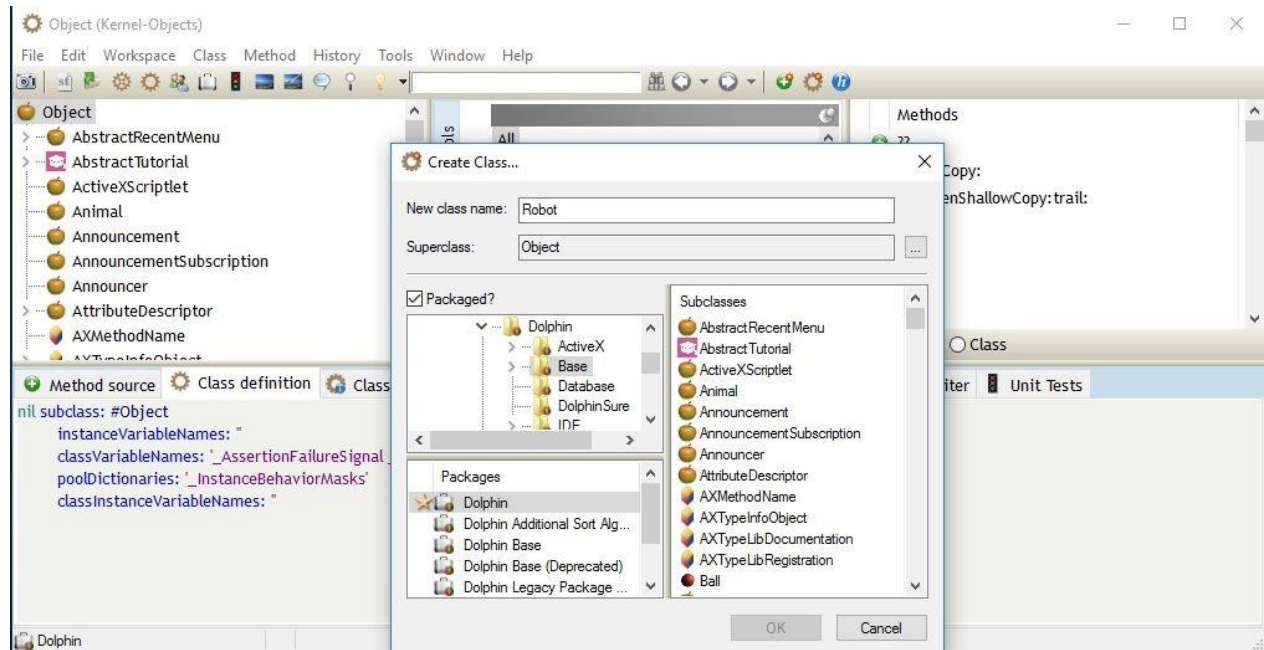
Para crear una clase en el Dolphin tenemos que ir al **Class Browser**.



Apretamos y nos va a aparecer una pantalla con todas las clases que actualmente tenemos. Para crear una clase vamos a **Class>New**.

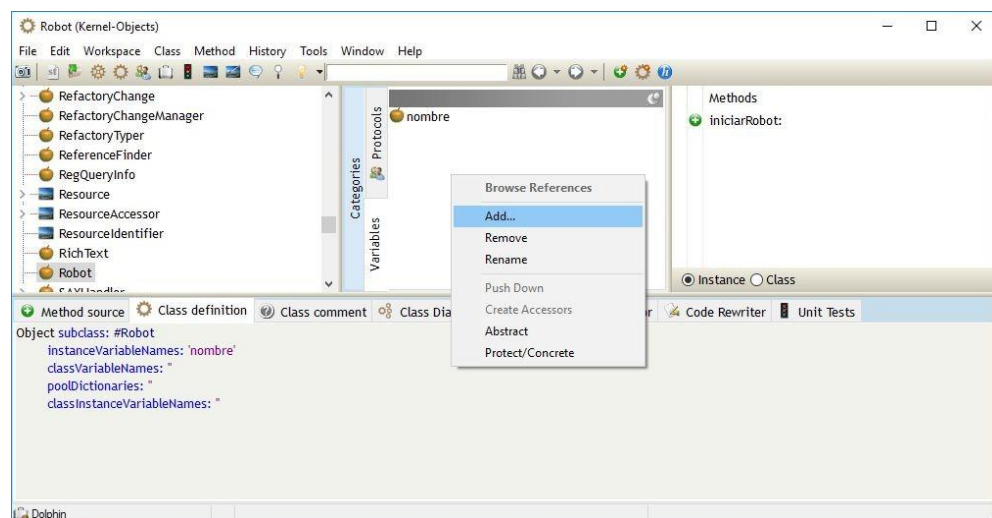


Ingresamos el nombre de la Clase y como **Superclass** seleccionamos **Object**. (En el caso de que se quiera hacer una clase hija se deberá seleccionar la superclase en la lista)

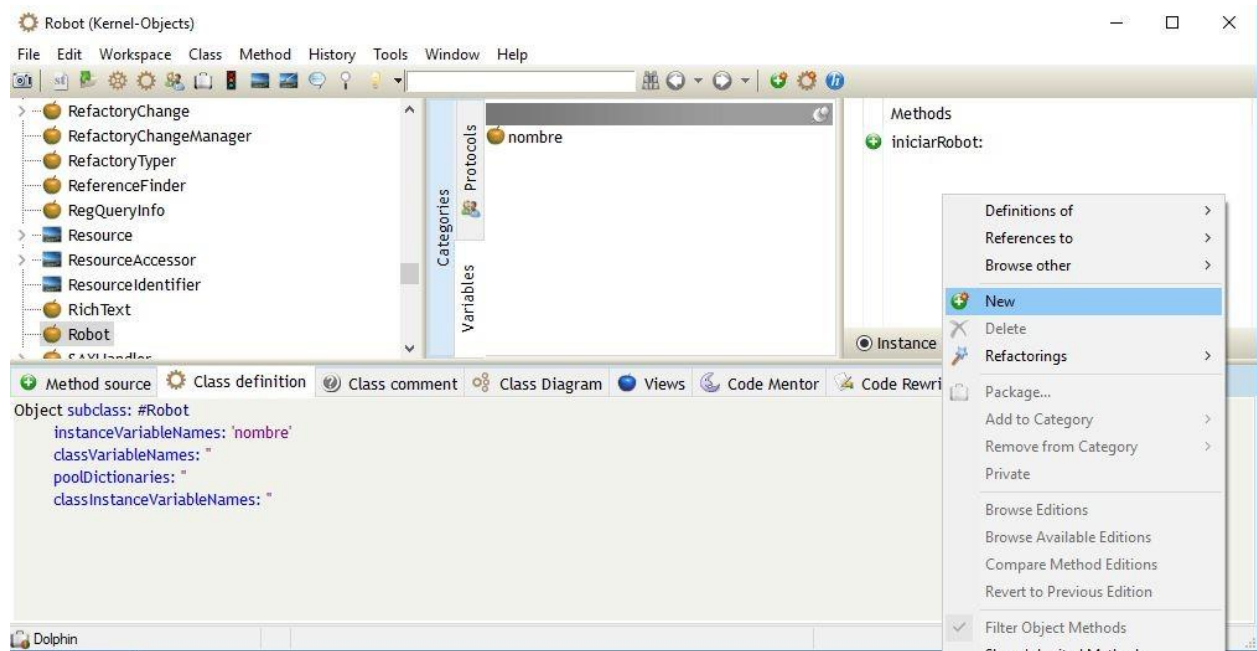


Una vez creada la buscamos en la lista de la izquierda, o en el buscador de la parte superior.

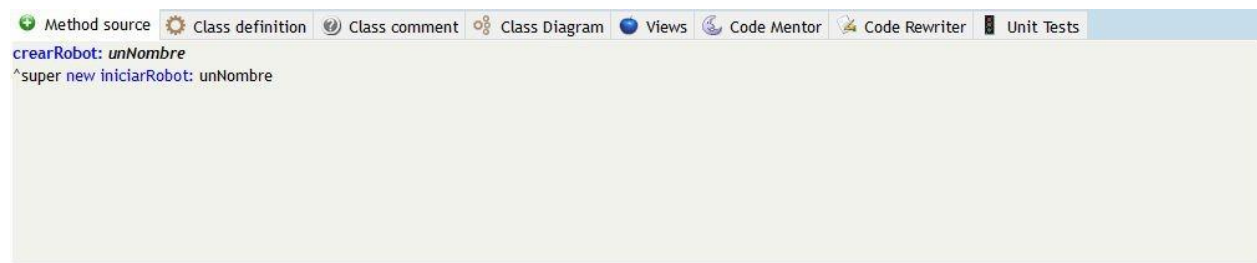
Ahora, para crear los atributos de la clase. Seleccionamos **Variables** en el panel central, **click derecho>Add**.



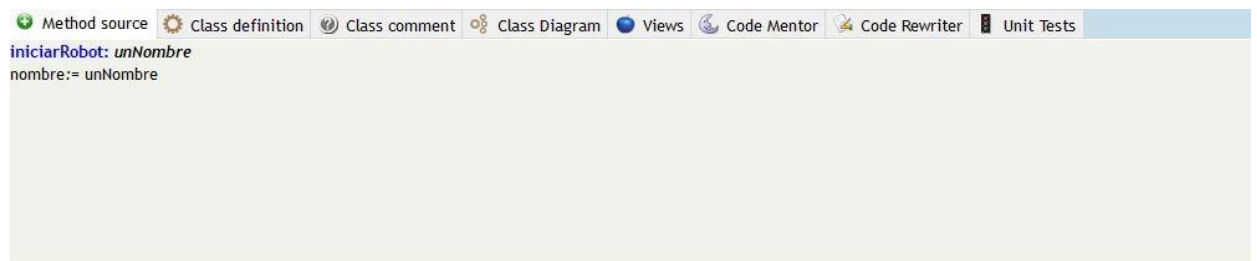
Para crear métodos, vamos al panel de la derecha. Previamente seleccionamos en la parte inferior si va a ser un **método de Clase o Instancia**.



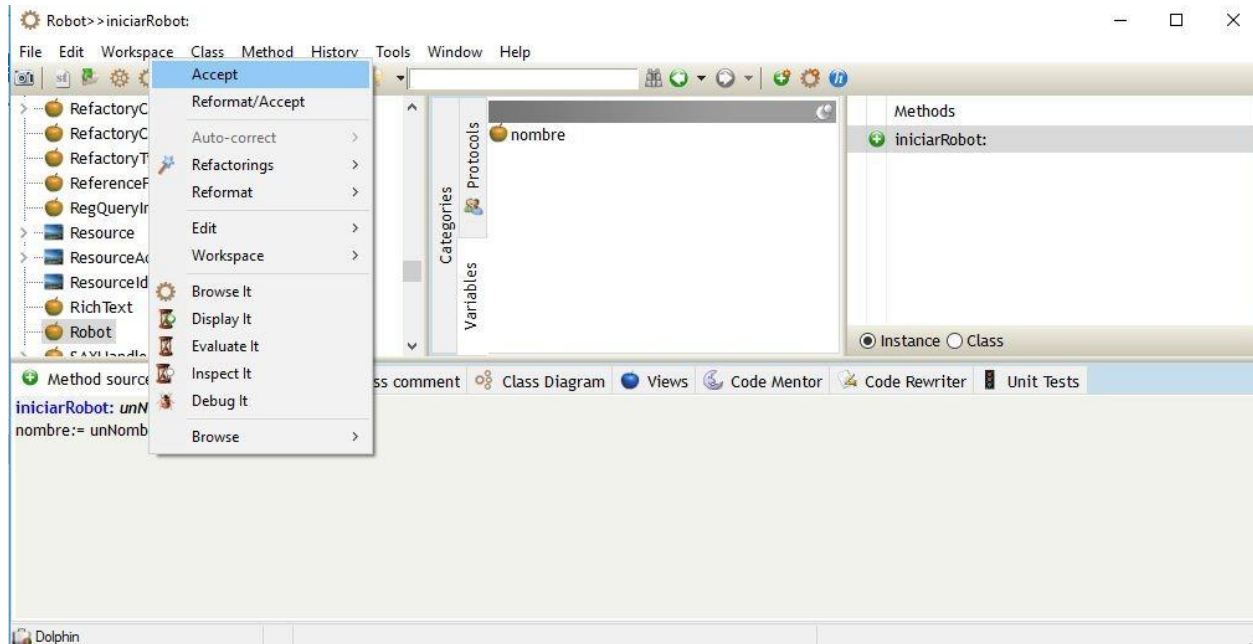
Al darle en **New** podremos escribir el método en la parte inferior de la pantalla.



Creamos el método de clase *crearRobot: unNombre* que va a utilizar el método de instancia *iniciarRobot: unNombre*.

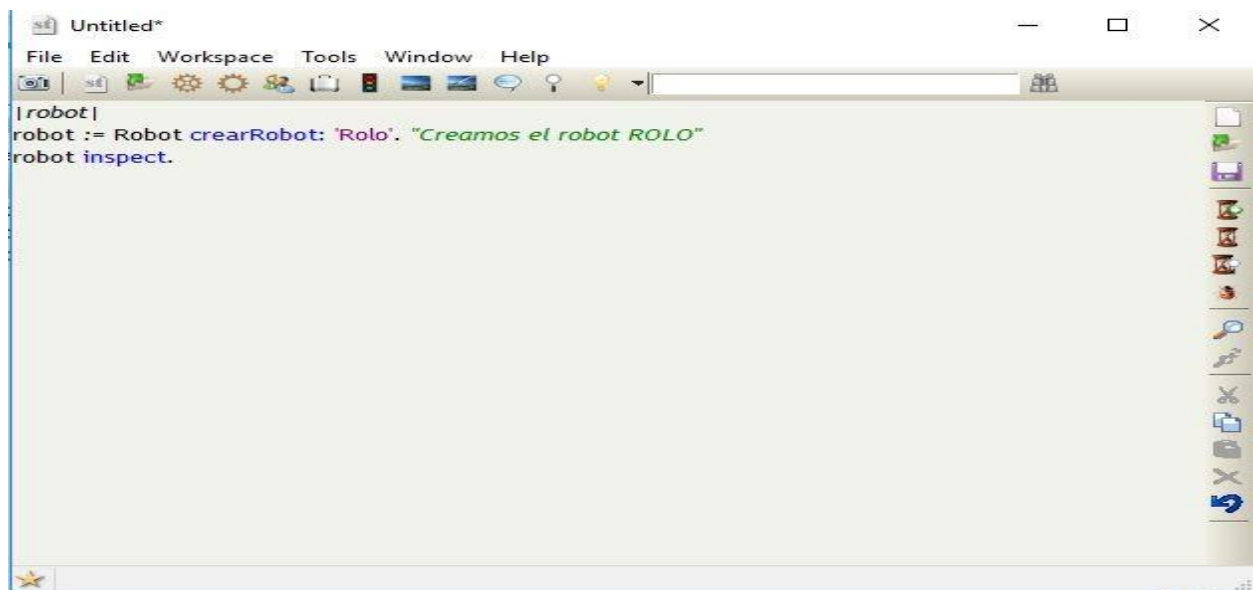


Es importante que después de escribir el método se haga **click derecho>Accept** para guardarlo.

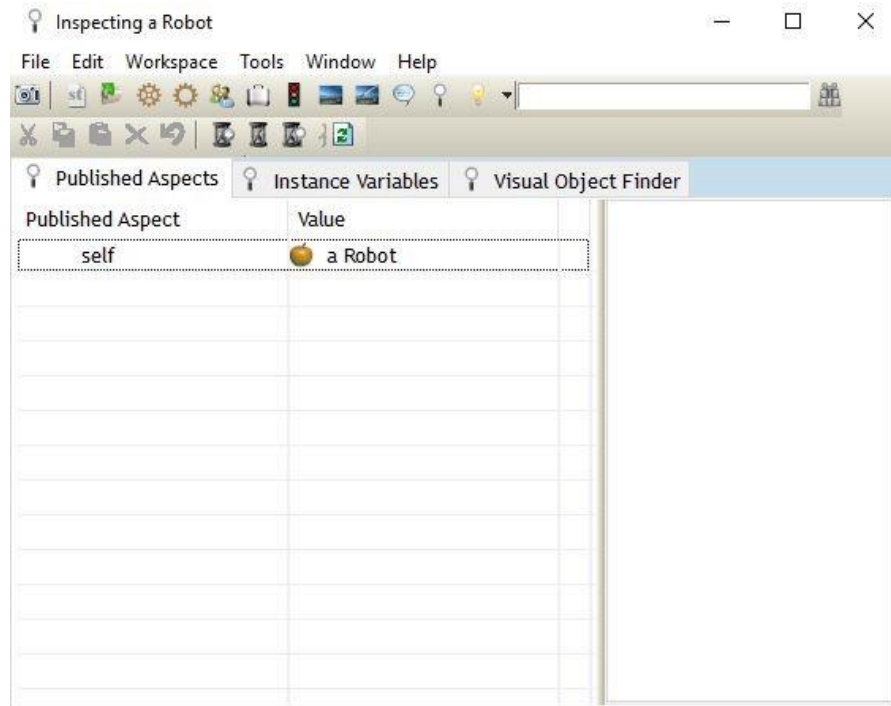


Una vez creados todos los atributos y métodos que necesitamos ya los podemos utilizar en nuestro programa.

Vamos al Workspace y los usamos.



Le hacemos un **inspect** a Robot para ver que todo se haya creado y cargado bien.



Se creó el objeto Robot. Si vamos a **Instance Variables** vemos la variable nombre ya cargada correctamente.

