


INSTRUMENTACION VIRTUAL

SPIN BOX

El Spin Box es un widget usado para desplegar valores enteros, valores de punto flotante y texto. Los valores del Spin Box son valores constates y el usuario no podrá introducir un valor aleatorio, podrá solo seleccionar el valor disponible en el Spin Box. El Spin Box puede mostrar un valor por default que puede ser incrementado y decrementado por los botones  y con las flechas arriba y abajo del teclado.


METODOS

El Spin Box puede ser creado usando dos clases, QSpinBox y QDoubleSpinBox, donde QSpinBox despliega solo valores enteros, y el QDoubleSpinBox desplegara valores de punto flotante. Los métodos del Spin Box son los siguientes.

- **value()**. Este método retorna el valor actual del Spin Box.
- **text()**. Este método retorna el texto desplegado en el Spin Box.
- **setValue()**. Este método asigna el valor al Spin Box.
- **setMinimum()**. Este método establece el valor mínimo del Spin Box.
- **setMaximum()**. Este método establece el valor máximo del Spin Box.

SEÑALES

Las señales emitidas por la clase QSpinBox son las siguientes.

- **valueChanged()**. Esta señal es emitida cuando el valor del Spin Box es cambiado ya se por los botones  o con las teclas de flecha hacia arriba o hacia abajo.
- **editingFinished()**. Esta señal es emitida cuando el foco en el Spin Box se pierde.

La clase usada para el Spin Box para números flotantes es QDoubleSpinBox. Todos los métodos vistos anteriormente son soportados por la clase QDoubleSpinBox. Este widget despliega hasta dos números decimales por default. Para cambiar la precisión, usa round().

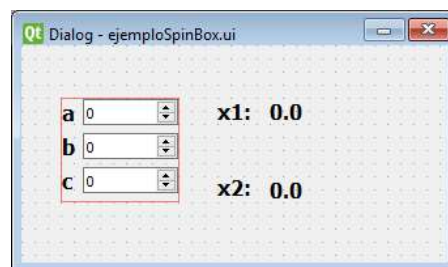
Vamos a hacer un ejemplo usando este widget, el siguiente ejemplo calcula la formular general el valor de las variables serán ingresados con el Spin Box. La fórmula es la siguiente.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

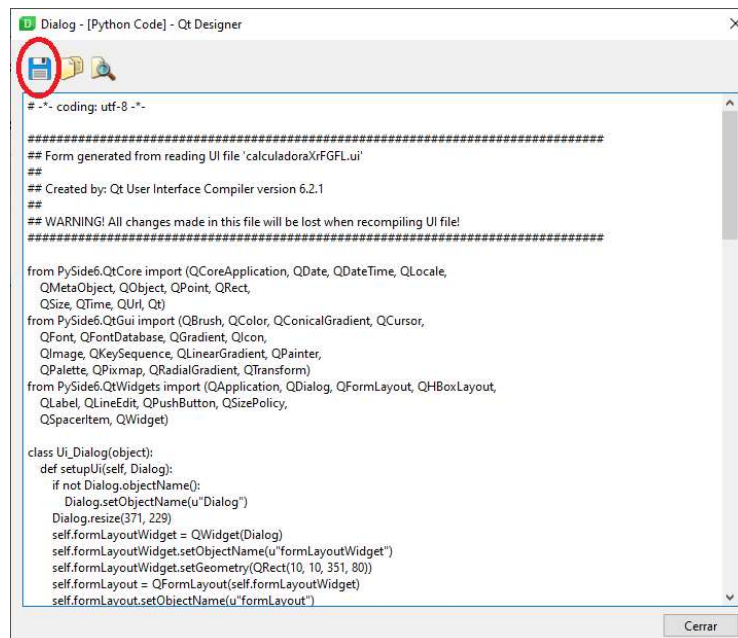
1. Vamos a crear un MainWindow y agregamos 7 Labels, 3 Spin Box y un Form Layout. Dentro del FormLayout agregamos tres Labels y tres SpinBox.
2. Los tres Labels que están dentro del FormLayout modificamos su propiedad *text* y ponemos a, b y c.

3. Los otros Labels que están fuera del FormLayout a dos les modificamos su propiedad *text* por x1 y x2 y los otros dos Labels cambiamos también su propiedad *text* y borramos su contenido.
4. Los tres SpinBox tenemos que modificar su propiedad *objectName* para asignarles estos nombres spValA, spValB, spValC respectivamente y los dos Labels sin texto también cambiamos su nombre en *objectName* por lblResultX1 y lblResultX2.
5. Guardamos el archivo con el nombre spinBox.ui.

La interfaz quedaría de la siguiente forma.



9. Después de crear la interfaz convertimos el archivo "ui" en código Python. Primer vamos al menú Formulario/View Python Code...



El programa quedaría de la siguiente forma.

```
import sys
from PySide6.QtWidgets import QApplication, QMainWindow
from spinBox import Ui_MainWindow
from math import *

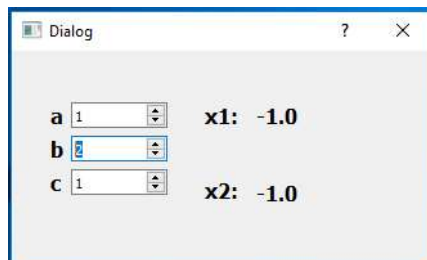
class MiFormulario(QMainWindow, Ui_MainWindow):
    def __init__(self):
        super().__init__()
        self.setupUi(self)
        self.spValA.valueChanged.connect(self.calcularRaices)
        self.spValB.valueChanged.connect(self.calcularRaices)
        self.spValC.valueChanged.connect(self.calcularRaices)

    def calcularRaices(self):
        a = self.spValA.value()
        b = self.spValB.value()
        c = self.spValC.value()

        try:
            raiz = sqrt((b ** 2) - (4 * a * c))
            aux = 2 * a
            x1 = ((-1 * b) + raiz) / aux
            x2 = ((-1 * b) - raiz) / aux
            self.lblResultX1.setText(str(x1))
            self.lblResultX2.setText(str(x2))
        except ZeroDivisionError:
            self.lblResultX1.setText("NaN")
            self.lblResultX2.setText("NaN")
        except ValueError:
            self.lblResultX1.setText("NaN")
            self.lblResultX2.setText("NaN")

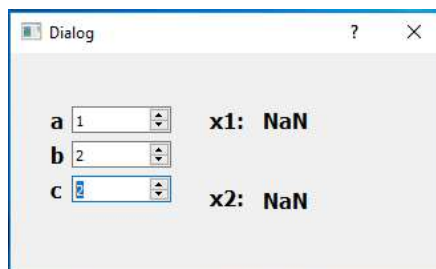
if __name__ == "__main__":
    app = QApplication(sys.argv)
    w = MiFormulario()
    w.show()
    sys.exit(app.exec())
```

Como resultado tenemos.



Dialog

a	1	x1:	-1.0
b	2		
c	1	x2:	-1.0



Dialog

a	1	x1:	NaN
b	2		
c	2	x2:	NaN