

1.-

Crear un QPushButton que cambie su tamaño dependiendo de la longitud del texto que tenga. Utiliza el decorador @Property para gestionar la propiedad text y un slot que se ejecute cuando cambie el texto.

Pasos:

1. Crea un CustomButton que herede de QPushButton.
2. Implementa una propiedad text que emita una señal cuando cambie el texto.
3. Conecta la señal textoCambiado a un slot que cambie el tamaño del botón de acuerdo con la longitud del texto.
4. Asegúrate de que el tamaño del botón cambie de forma dinámica al modificar el texto.

2.-

Crear una interfaz con un QCheckBox que, cuando se marque o desmarque, cambie el texto de un QLabel. Utiliza el decorador @Slot para conectar la señal del QCheckBox al slot que modificará el texto del QLabel.

Pasos:

1. Crea un archivo de interfaz con Qt Designer que tenga un QCheckBox y un QLabel.
2. En el archivo Python, conecta la señal toggled del QCheckBox a un slot que cambie el texto del QLabel dependiendo de si el QCheckBox está marcado o no.
3. Cuando el QCheckBox esté marcado, cambia el texto del QLabel a "Activado", y cuando no lo esté, cambia el texto a "Desactivado".