

PYQT6 Course

3. Creating QLineEdit in PyQt6

A line edit allows the user to enter and edit a single line of plain text with a useful collection of editing functions, including undo and redo, cut and paste, and drag and drop. For creating line edit we need to create the object of QLineEdit, there are different methods that you can use in QLineEdit.

- **setEchoMode():** It sets the echo mode of the Line Edit widget. it determines how the contents of the Line Edit widget are to be displayed. The available options are as follows:
 1. **Normal:** This is the default mode and it displays characters the way they are entered
 2. **NoEcho:** It switches off the Line Edit echo, that is, it doesn't display anything
 3. **Password:** This option is used for password fields, no text will be displayed; instead, asterisks appear for the text entered by the user
 4. **PasswordEchoOnEdit:** It displays the actual text while editing the password fields, otherwise it will display the asterisks for the text

- `maxLength()`: This method is used to specify the maximum length of text that can be entered in the Line Edit widget.
- `setText()`: This method is used for assigning text to the Line Edit widget.
- `text()`: This method accesses the text entered in the Line Edit widget.
- `clear()`: This method clears or deletes the complete content of the Line Edit widget.
- `setReadOnly()`: When the Boolean value `true` is passed to this method, it will make the Line Edit widget read-only, that is, non-editable. The user cannot make any changes to the contents displayed through the Line Edit widget, but can only copy.
- `setEnabled()`: By default, the Line Edit widget is enabled, that is, the user can make changes to it. But if the Boolean value `false` is passed to this method, it will disable the Line Edit widget so the user cannot edit its content, but can only assign text via the `setText()` method.
- `setFocus()`: This method positions the cursor on the specified Line Edit widget.
- `setPlaceholderText()`: This method is used for adding placeholder text to the line edit.

For creating of the line edit, you need to create the object of QLineEdit class, QLineEdit class is related to QtWidgets module.

```
line_edit = QLineEdit(self)
```

In here we have set the default text for the QLineEdit.

```
line_edit.setText("Default Text")
```

In here we have added a placeholder text to the line edit.

```
line_edit.setPlaceholderText("Please enter your  
username")
```

Using this method we can disable the line edit.

```
line_edit.setEnabled(False)
```

If you want to create a password line edit than you can use this code.

```
line_edit.setEchoMode(QLineEdit.EchoMode.Password)
```