PYQT6 Course

13. Working with QCheckBox in PyQt6

A QCheckBox is an option button that can be switched on (checked) or off (unchecked). Checkboxes are typically used to represent features in an application that can be enabled or disabled without affecting others. Whenever a checkbox is checked, it emits the signal stateChanged(). Connect to this signal if you want to trigger an action each time the checkbox changes state. also you can use isChecked() method to query whether or not a checkbox is checked. mostly checkbox is used when you want the user to select one or more than one option from a set of options. In such situations, you need to make use of checkboxes

The QCheckBox class provides the following methods:

- isChecked(): This method returns the Boolean value true if the button is in the selected state.
- setIcon(): This method displays an icon with the checkbox.
- setText(): This method assigns the text to the checkbox.

- setChecked(): To make any checkbox appear selected by default, pass the Boolean value true to this method.

QCheckBox class is related to QtWidgets module, first we need to create the object of QCheckBox class.

```
self.check1 = QCheckBox("Python")
```

In here we have added an icon to the CheckBox, make sure that you have already added some icons to the project.

```
self.check1.setIcon(QIcon("images/py.png"))
```

With setIconSize() method we can set the icon size, it expects a QSize and you need to add the width and height of the icon.

```
self.check1.setIconSize(QSize(40,40))
```

With this method we can change the font of the QCheckBox.

```
self.check1.setFont(QFont("Sanserif", 13))
```

As I have already said that we can use stateChanged() signal in QCheckBox. In here we have connected the stateChanged signal to the item_selected() method or slot.

```
self.check1.stateChanged.connect(self.item_selected)
```

And this is the method that we have already connected to our signal, basically in this method we want to check the state of the QCheckBox using isChecked() method and according to the state we want to change the text of the label.

```
def item_selected(self):
    value = ""

    if self.check1.isChecked():
        value = self.check1.text()

    if self.check2.isChecked():
        value = self.check2.text()
    if self.check3.isChecked():
        value = self.check3.text()

    self.label.setText("You have selected : " + value)
```