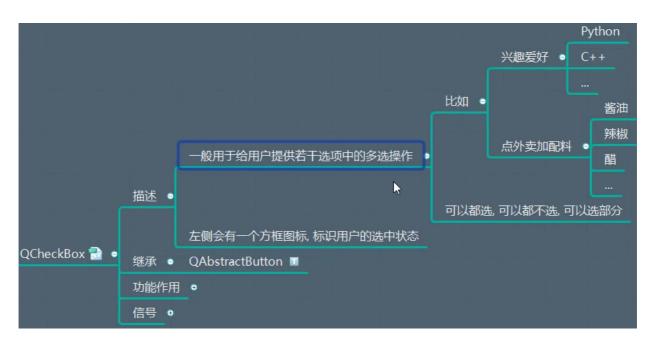
[笔记][LIKE-Python-GUI编程-PyQt5][15]

Pv0t5

[笔记][LIKE-Python-GUI编程-PyQt5][15]
110. Python-GUI编程-PyQt5-QCheckBox-功能使用
111. Python-GUI编程-PyQt5-QCheckBox-信号

110. Python-GUI编程-PyQt5-QCheckBox-功能使用



```
QCheckBox(parent=None)
                        创建复选框按钮 🏻
                                       QCheckBox(text, parent=None)
                                         图标 • setIcon(QIcon)
                        常用继承父类操作
                                                 文本加&
                                         快捷键
                                                 setShortcut()
                                            setTristate(bool=True)
QCheckBox 📆 🏮
                                      API •
                                            isTristate()
              功能作用
                        设置是否三态 •
                                      应用场景 • 是否支持复选框展示三种状态
                             R
                                              setCheckState(Qt.CheckState)
                                       API •
                                              checkState()
                        设置复选框状态
                                       应用场景 ● 想给用户多一种选择状态
                                       Qt.Unchecked • 该项目未选中
                        补充 ● 状态 ●
                                       Qt.PartiallyChecked • 部分选中
                                       Qt::Checked • 真的被选中
```

```
# 0. 导入需要的包和模块
import sys
from PyQt5.Qt import *
# 1. 创建一个应用程序对象
app = QApplication(sys.argv)
# 2. 控件的操作
# 2.1 创建控件
window = QWidget()
# 2.2 设置控件
window.setWindowTitle('QCheckBox功能测试')
window.resize(500, 500)
# 通过打印属性看父类
print('QCheckBox直接继承的父类: ', QCheckBox.__bases__)
# 新建 QCheckbox
# QCheckBox(parent: QWidget = None)
# QCheckBox(str, parent: QWidget = None)
cb = QCheckBox('&Python', window)
# 设置图标
cb.setIcon(QIcon('img/Python.png'))
# 设置图标大小
cb.setIconSize(QSize(30, 30))
# 设置快捷键 &Python 或者 setShortcut
```

```
# 设置是否三态
 # 点一下: 小方框
 # 再点一下: 对勾
 # 再点一下: 空格子
 cb.setTristate(True)
 # 设置复选框状态
 # 注意: 因为有三态, 要用 setCheckState(Qt.CheckState)
 # Qt. Unchecked 未被选中
 # Qt.PartiallyChecked 部分选中
 # Qt.Checked 全部选中
 cb.setCheckState(Qt.PartiallyChecked)
 # 当然 setChecked(bool) 也可以,只不过没法表示部分选中的状态
 cb.setChecked(True)
 # 2.3 展示控件
 window.show()
 # 3. 应用程序的执行,进入到消息循环
 sys.exit(app.exec_())
```

111. Python-GUI编程-PyQt5-QCheckBox-信号

stateChanged(int state) • 选中或清除选中时,发射此信号信号 • 其他都继承

注意: QCheckBox 从 QAbstractButton 继承来的信号 toggled , 只能检测到未被选中与部分 被选中的状态 , 不能检测到完全选中的状态。

```
# 0. 导入需要的包和模块
import sys
from PyQt5.Qt import *

# 1. 创建一个应用程序对象
app = QApplication(sys.argv)

# 2. 控件的操作
# 2.1 创建控件
window = QWidget()
# 2.2 设置控件
window.setWindowTitle('QCheckBox信号')
window.resize(500, 500)
```

```
cb = QCheckBox('&Python', window)
cb.setIcon(QIcon('img/Python.png'))
cb.setIconSize(QSize(30, 30))
cb.setTristate(True)

# cb.stateChanged.connect(lambda state: print('状态是: ', state))
cb.toggled.connect(lambda is_checked: print('是否被选中: ', is_checked))

# 2.3 展示控件
window.show()

# 3. 应用程序的执行,进入到消息循环
sys.exit(app.exec_())
```

完成于 201810242046