

# [笔记][LIKE-Python-GUI编程-PyQt5][13]

PyQt5

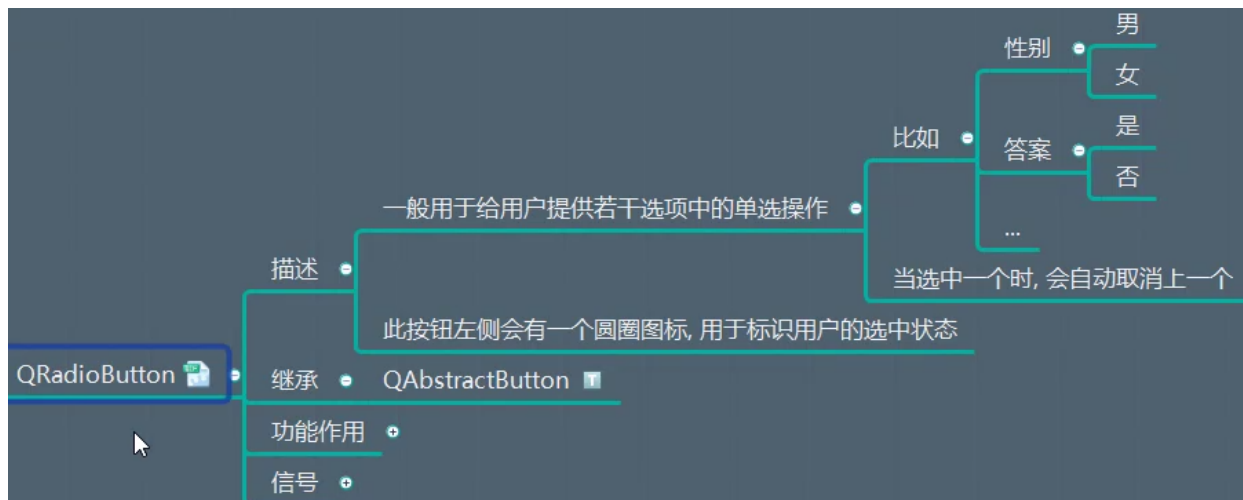
[笔记][LIKE-Python-GUI编程-PyQt5][13]

101. Python-GUI编程-PyQt5-QRadioButton-创建和基本设置

102. Python-GUI编程-PyQt5-QRadioButton-信号使用

103. Python-GUI编程-PyQt5-QRadioButton-多组互斥问题

## 101. Python-GUI编程-PyQt5-QRadioButton-创建和基本设置



```
# 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('QRadioButton功能测试')
window.resize(500, 500)

# QRadioButton 构造函数
# QRadioButton(parent: QWidget = None) 也可以不传父对象，默认没有父对象
# QRadioButton(str, parent: QWidget = None)

# 只有一个 QRadioButton 的时候
# 再点一下可以取消
# 有多个单选按钮则无法取消
# 快捷键方式1: 文本加 &
rb_male = QRadioButton('男-&Male', window)
rb_male.move(100, 100)
rb_male.setIcon(QIcon('img/male.png'))
rb_male.setIconSize(QSize(40, 40))
# 默认设置选中男性
rb_male.setChecked(True)

rb_female = QRadioButton('女-Female', window)
rb_female.move(100, 150)
rb_female.setIcon(QIcon('img/female.png'))
rb_female.setIconSize(QSize(40, 40))
# 快捷方式2: setShortcut
rb_female.setShortcut('Alt+F')

# 2.3 展示控件
window.show()

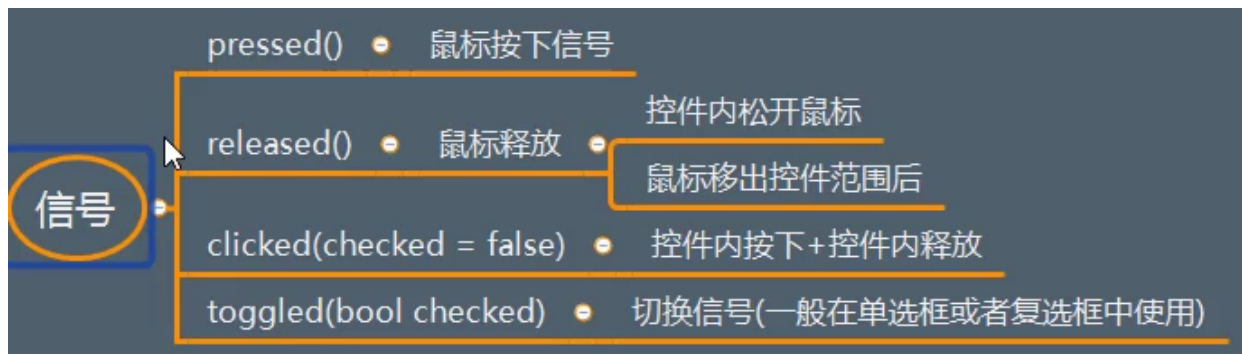
# 3. 应用程序的执行，进入到消息循环
sys.exit(app.exec_())

```

## 102. Python-GUI编程-PyQt5-QRadioButton-信号使用



## QPushButton 的信号



最常用为 `toggled()`

```
# 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('QPushButton功能测试')
window.resize(500, 500)

# QPushButton 构造函数
# QPushButton(parent: QWidget = None) 也可以不传父对象, 默认没有父对象
# QPushButton(str, parent: QWidget = None)

# 只有一个 QPushButton 的时候
# 再点一下可以取消
# 有多个单选按钮则无法取消
# 快捷键方式1: 文本加 &
rb_male = QPushButton('男-&Male', window)
rb_male.move(100, 100)
rb_male.setIcon(QIcon('img/male.png'))
rb_male.setIconSize(QSize(40, 40))
# 默认设置选中男性
rb_male.setChecked(True)

rb_female = QPushButton('女-Female', window)
rb_female.move(100, 150)
rb_female.setIcon(QIcon('img/female.png'))
rb_female.setIconSize(QSize(40, 40))
# 快捷方式2: setShortcut
rb_female.setShortcut('Alt+F')
# 状态发生改变就会发射信号
rb_female.toggled.connect(lambda is_checked: print('女', is_checked))
# 设置为非独占, 即不排他! 这样就有点类似复选框了!
```

```
rb_female.setAutoExclusive(False)
```

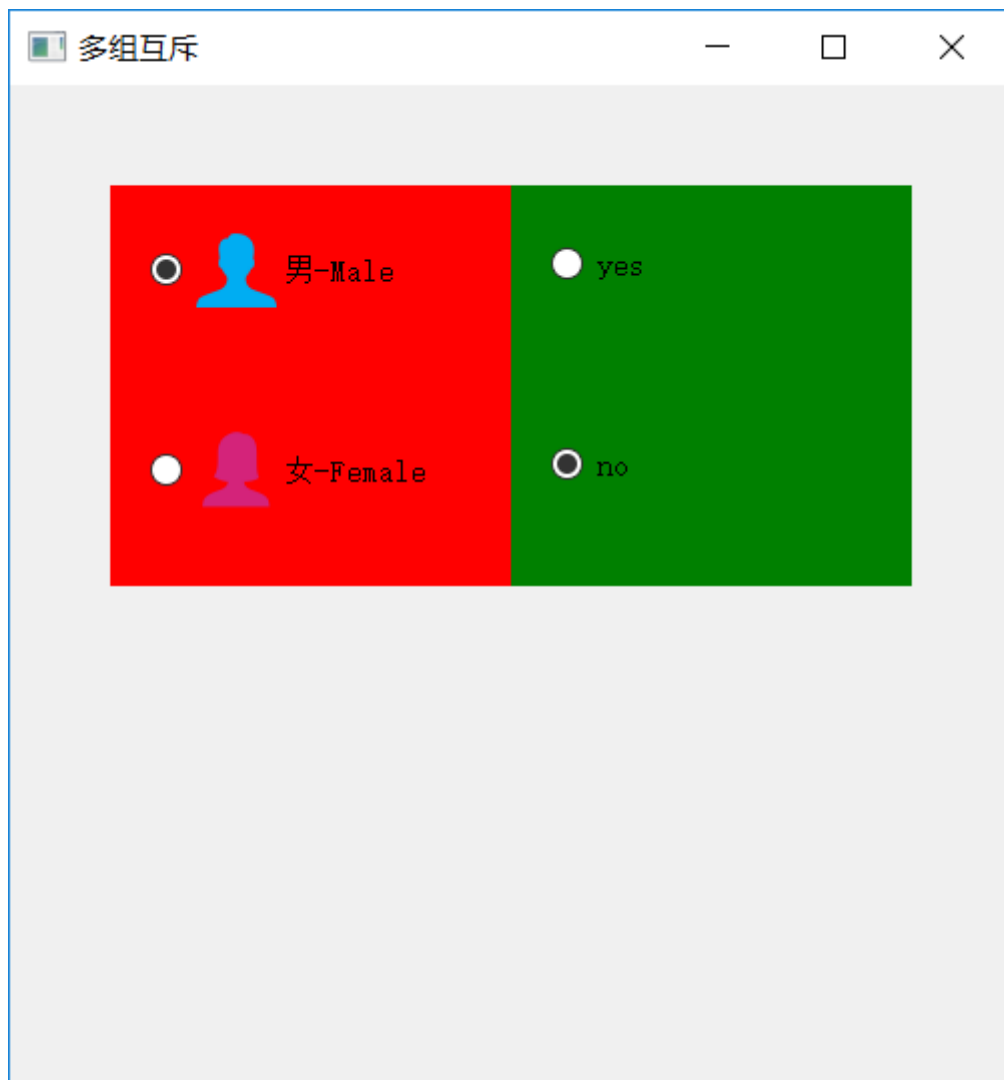
```
# 2.3 展示控件
```

```
window.show()
```

```
# 3. 应用程序的执行，进入到消息循环
```

```
sys.exit(app.exec_())
```

## 103. Python-GUI编程-PyQt5-QRadioButton-多组互斥问题



```
# 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('多组互斥')
window.resize(500, 500)

# 多组互斥解决方案1: 放在两个父控件里面
red = QWidget(window)
red.resize(200, 200)
red.setStyleSheet('background-color: red;')
red.move(50, 50)

green = QWidget(window)
green.resize(200, 200)
green.setStyleSheet('background-color: green;')
green.move(250, 50)

rb_male = QRadioButton('男-&Male', red)
rb_male.move(20, 20)
rb_male.setIcon(QIcon('img/male.png'))
rb_male.setIconSize(QSize(40, 40))
rb_male.setChecked(True)

rb_female = QRadioButton('女-Female', red)
rb_female.move(20, 120)
rb_female.setIcon(QIcon('img/female.png'))
rb_female.setIconSize(QSize(40, 40))
rb_female.setShortcut('Alt+F')

rb_yes = QRadioButton('yes', green)
rb_yes.move(20, 30)

rb_no = QRadioButton('no', green)
rb_no.move(20, 130)

# 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_())
```