

对话框是为了更好地实现人与程序的交互 对话框主要是完成特定场景下的功能,比如删除确认等.

QMessageBox

QMessageBox是普通的对话框



```
1
2 from PyQt5.Qtwidgets import *
3 from PyQt5.QtCore import *
4 from PyQt5.QtGui import *
5 import sys
6
7
8 def do_click():
9     # QMessageBox 对话框
10    # information: 信息对话框 parent, title, content 阻塞式
11    # QMessageBox.information(window, '我是对话框的title', '我是对话框的内容')
12
13    # question: 问答对话框
14    # result = QMessageBox.question(window, 'title', 'content')
15    # print(type(result))
16    # if result == QMessageBox.Yes:
17    #     print('yes')
18    # else:
19    #     print('no')
20
21    # warning
22    # QMessageBox.warning(window, 'title', 'content')
```

```

23
24     # critical
25     # QMessageBox.critical(window, 'title', 'content')
26
27     # about
28     QMessageBox.about(window, 'title', 'content')
29
30     print('对话框消失了')
31
32
33 if __name__ == '__main__':
34     # 创建app
35     app = QApplication(sys.argv)
36
37     # 窗体显示
38     window = QWidget()
39
40     # 按钮
41     btn = QPushButton('弹出对话框')
42     # 设置到界面
43     btn.setParent(window)
44
45     window.show()
46
47     # 事件绑定
48     btn.clicked.connect(do_click)
49
50     # 执行app
51     sys.exit(app.exec_())

```

对话框类型

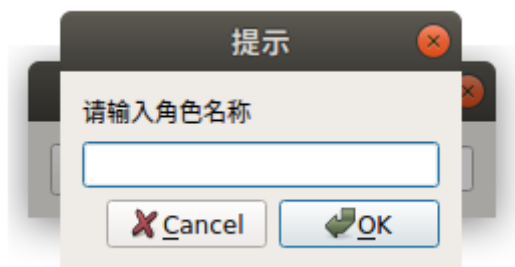
对话框类型	描述
QMessageBox.information(QWidget, title, text, buttons, defaultButton)	弹出消息对话框
QMessageBox.question(QWidget, title, text, buttons, defaultButton)	弹出问答对话框
QMessageBox.warning(QWidget, title, text, buttons, defaultButton)	弹出警告对话框
QMessageBox.ctitical(QWidget, title, text, buttons, defaultButton)	弹出严重错误对话框
about(QWidget, title, text)	弹出关于对话框

按钮类型

类型	描述
QMessageBox.Ok	确定
QMessageBox.Cancel	取消
QMessageBox.Yes	同意
QMessageBox.No	不同意
QMessageBox.Abort	终止
QMessageBox.Retry	重试
QMessageBox.Ignore	忽略

QInputDialog

QInputDialog是输入对话框,由一个文本框和两个按钮(OK按钮和Cancel按钮)组成



```
1  from PyQt5.Qtwidgets import *
2  from PyQt5.QtCore import *
3  from PyQt5.QtGui import QIcon
4  import sys
5
6  def func():
7      str,success = QInputDialog.getText(w,'提示','请输入角色名称')
8      if success:
9          edit.setText(str)
10
11
12  # 1.创建应用程序
13  app = QApplication(sys.argv)
14
15  # 2.创建窗口
16  w = QWidget()
17
18
19  # 修改窗口标题
20  w.setWindowTitle('输入提示框')
21  """----- 创建界面 -----"""
22  layout = QHBoxLayout()
23  # 添加布局
24  w.setLayout(layout)
25
```

```
26 # 控件
27 btn = QPushButton('创建角色')
28 edit = QLineEdit()
29 # 控件添加到布局中
30 layout.addWidget(btn)
31 layout.addWidget(edit)
32
33 # 信号和槽函数绑定
34 btn.clicked.connect(func)
35
36 # 3.显示窗口
37 w.show()
38
39 # 4.等待窗口停止
40 sys.exit(app.exec())
```