简述

QRadioButton部件提供了一个带有文本标签的单选框（单选按钮）。

QRadioButton是一个可以切换选中（checked）或未选中（unchecked）状态的选项按钮。单选框通常呈现给用户一个“多选一”的选择。也就是说，在一组单选框中，一次只能选中一个单选框。

详细描述

单选框默认开启自动互斥（autoExclusive）。如果启用了自动互斥，属于同一个父部件的单选框的行为就和属于一个互斥按钮组的一样。如果你需要为属于同一父部件的单选框设置多个互斥按钮组，把它们加入QButtonGroup中。

每当一个按钮切换选中或未选中状态时，会发出的toggled()信号。如果希望每个按钮切换状态时触发一个动作，连接到这个信号。使用isChecked()来查看特定按钮是否被选中。

就像QPushButton一样，单选框可以显示文本，以及可选的小图标。图标使用setIcon()来设置，文本可以在构造函数或通过setText()来设置。可以指定快捷键，通过在文本中的特定字符前指定一个&。

例如：

QRadioButton \*button = new QRadioButton("Search from the &cursor", this);

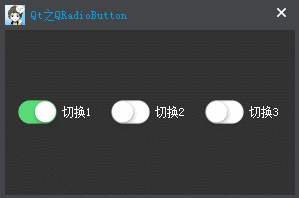
1

这个示例中，快捷键为Alt+c。关于更多快捷键的内容请参考：QShortcut 。如果要显示一个“&”，请使用’&&’。

示例

我们来实现一个iphone中常见的开关效果 - 单选。

效果



源码

构建单选框QRadioButton，然后将它们添加至按钮组QButtonGroup中。

QHBoxLayout \*pLayout = new QHBoxLayout();

m\_pButtonGroup = new QButtonGroup(this);

// 设置互斥

m\_pButtonGroup->setExclusive(true);

for (int i = 0; i < 3; ++i)

{

QRadioButton \*pButton = new QRadioButton(this);

// 设置文本

pButton->setText(QString::fromLocal8Bit("切换%1").arg(i + 1));

pLayout->addWidget(pButton);

m\_pButtonGroup->addButton(pButton);

}

pLayout->setSpacing(10);

pLayout->setContentsMargins(10, 10, 10, 10);

setLayout(pLayout);

// 连接信号槽

connect(m\_pButtonGroup, SIGNAL(buttonClicked(QAbstractButton\*)), this, SLOT(onButtonClicked(QAbstractButton\*)));

槽函数，用来判断当前点击的按钮，以及获取按钮组中各个按钮的选中状态。

void MainWindow::onButtonClicked(QAbstractButton \*button)

{

// 当前点击的按钮

qDebug() << QString("Clicked Button : %1").arg(button->text());

// 遍历按钮，获取选中状态

QList<QAbstractButton\*> list = m\_pButtonGroup->buttons();

foreach (QAbstractButton \*pButton, list)

{

QString strStatus = pButton->isChecked() ? "Checked" : "Unchecked";

qDebug() << QString("Button : %1 is %2").arg(button->text()).arg(strStatus);

}

}

样式

单选框样式

QRadioButton{

spacing: 2px;

color: white;

}

QRadioButton::indicator {

width: 45px;

height: 30px;

}

QRadioButton::indicator:unchecked {

image: url(:/Images/switchOff);

}

QRadioButton::indicator:unchecked:hover {

image: url(:/Images/switchOffHover);

}

QRadioButton::indicator:unchecked:pressed {

image: url(:/Images/switchOffPressed);

}

QRadioButton::indicator:checked {

image: url(:/Images/switchOn);

}

QRadioButton::indicator:checked:hover {

image: url(:/Images/switchOnHover);

}

QRadioButton::indicator:checked:pressed {

image: url(:/Images/switchOnPressed);

}

上面，我们通过调用QButtonGroup的setExclusive(true)来设置按钮组中的单选框互斥。

当然，也可以设置setExclusive(false)来关闭互斥，从而实现多选功能。即使这样，如之前所言 - 强烈建议使用众所周知的约定。所以，如果要实现多选功能，建议选择QCheckBox。