

# Chapter 11 – MdiArea

---

## mdiDemo application

### Step 1 Open Qt Designer

- Open the QtDesigner by selecting the Qt Designer program icon



### Step 2 Open the Tab Widget application

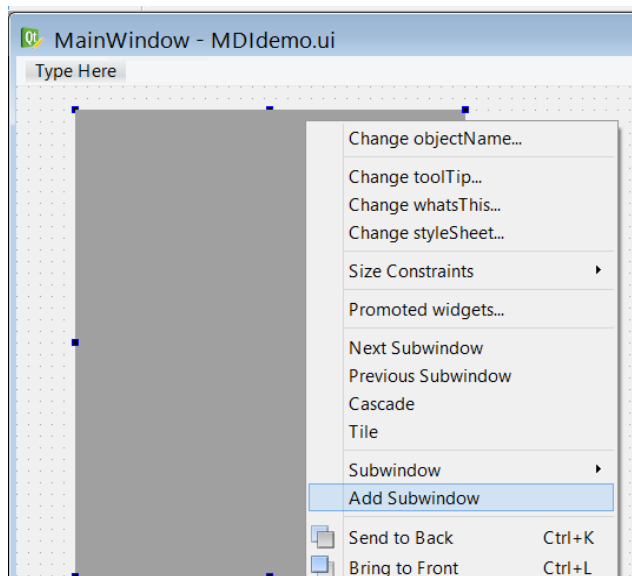
- Click on the “Main Window”
- Click the Create button
- A new form with the caption “untitled” is created with a menubar

### Step 3 Save the application as

- Click File > Save As
- Save the file as MDIdemo.ui

### Step 4 Add widgets to the form

- Drag and drop an MDIarea widget from the Containers section onto the form.
- Right click on the widget and select Add Subwindow



- A new window should be created within the bounds of the MDIarea widget.
- Repeat the process to add another subwindow
- Add the following widgets to the first subwindow

Widget	Property	Value
QLabel (Display widgets section)	objectName text	label Enter your views here
QTextEdit (Display widgets section)	objectName	textEdit

- Add the following widgets to the second subwindow

Widget	Property	Value
QLabel (Display widgets section)	objectName text	label_2 This is second Sub Window

- Add the following widgets to the main window

Widget	Property	Value
QPushButton (Buttons section)	objectName text	showNext Show Next
QPushButton (Buttons section)	objectName text	showPrevious Show Previous
QPushButton (Buttons section)	objectName text	closeAll Close All
QPushButton (Buttons section)	objectName text	cascadeButton Cascade
QPushButton (Buttons section)	objectName text	tileButton Tile
QPushButton (Buttons section)	objectName text	SubWindowViewButton Sub Window View
QPushButton (Buttons section)	objectName text	TabbedViewButton Tabbed View

- Change the Type Here place holder to Windows.
- Add the following entries to the Windows menu
  - First Window
  - Second Window

## Step 5 Save the form

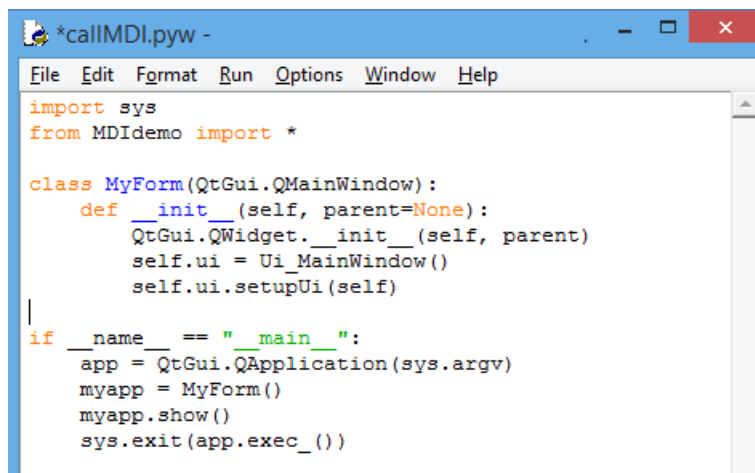
- Save the file as MDIdemo.ui **(note the case!! Python is case sensitive)**

## Step 6 Convert the .ui file to a .py file

- Convert the MDIdemo.ui file to MDIdemo.py using pyuic4. **(note the case!! Python is case sensitive, so even on file names the case must be the same throughout)**

## Step 7 Create a source file (.pyw) that imports the .py file

- Create a source file that will import the .py file created in step above and from which we will invoke the user interface
- Use the following code **(note the indentation and case!!)**



```
File Edit Format Run Options Window Help
import sys
from MDIdemo import *

class MyForm(QtGui.QMainWindow):
    def __init__(self, parent=None):
        QtGui.QWidget.__init__(self, parent)
        self.ui = Ui_MainWindow()
        self.ui.setupUi(self)

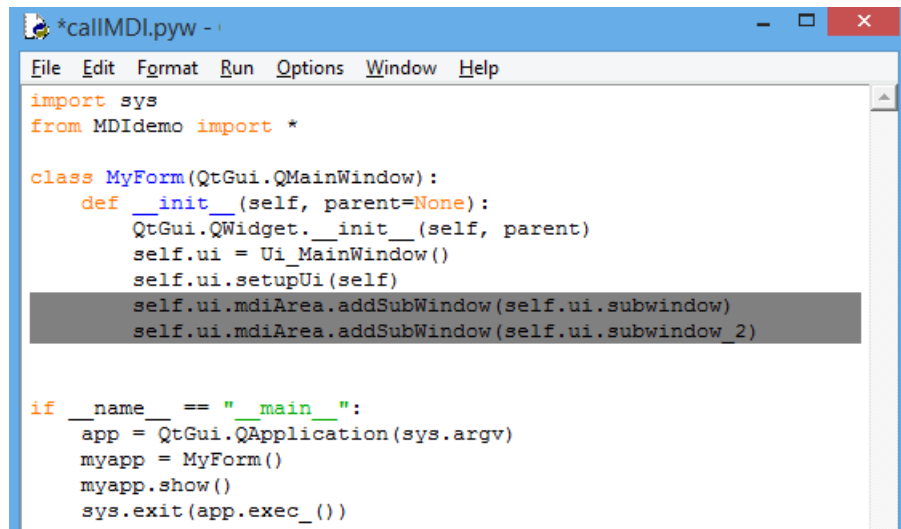
if __name__ == "__main__":
    app = QtGui.QApplication(sys.argv)
    myapp = MyForm()
    myapp.show()
    sys.exit(app.exec_())
```

- Save the file as callMDI.pyw
- Run and test the application upto this point.

## Step 7 Add code to create the combobox items and signal/slots

- Add the following code to create the subwindows: **(Note the case and indentation)**

```
self.ui.mdiArea.addSubWindow(self.ui.subwindow)
self.ui.mdiArea.addSubWindow(self.ui.subwindow_2)
```



```
*callMDI.pyw -
File Edit Format Run Options Window Help
import sys
from MDIdemo import *

class MyForm(QtGui.QMainWindow):
    def __init__(self, parent=None):
        QtGui.QWidget.__init__(self, parent)
        self.ui = Ui_MainWindow()
        self.ui.setupUi(self)
        self.ui.mdiArea.addSubWindow(self.ui.subwindow)
        self.ui.mdiArea.addSubWindow(self.ui.subwindow_2)

if __name__ == "__main__":
    app = QtGui.QApplication(sys.argv)
    myapp = MyForm()
    myapp.show()
    sys.exit(app.exec_())
```

- Add the following code to create the functions and the signal/slot combinations (Note the case and indentation)

```
QtCore.QObject.connect(self.ui.showNext, QtCore.SIGNAL('clicked()'),
self.displayNext)
QtCore.QObject.connect(self.ui.showPrevious,
QtCore.SIGNAL('clicked()'), self.displayPrevious)
QtCore.QObject.connect(self.ui.closeAll, QtCore.SIGNAL('clicked()'),
self.closeAll)
QtCore.QObject.connect(self.ui.cascadeButton,
QtCore.SIGNAL('clicked()'), self.cascadeArrange)
QtCore.QObject.connect(self.ui.tileButton, QtCore.SIGNAL('clicked()'),
self.tileArrange)
QtCore.QObject.connect(self.ui.SubWindowViewButton,
QtCore.SIGNAL('clicked()'), self.SubWindowView)
QtCore.QObject.connect(self.ui.TabbedViewButton,
QtCore.SIGNAL('clicked()'), self.TabbedView)
self.connect(self.ui.actionFirst_Window, QtCore.SIGNAL('triggered()'),
self.displayNext)
self.connect(self.ui.actionSecond_Window, QtCore.SIGNAL('triggered()'),
self.displayPrevious)

def displayNext(self):
    self.ui.mdiArea.activateNextSubWindow()

def displayPrevious(self):
    self.ui.mdiArea.activatePreviousSubWindow()

def closeAll(self):
    self.ui.mdiArea.closeAllSubWindows()

def cascadeArrange(self):
    self.ui.mdiArea.cascadeSubWindows()

def tileArrange(self):
    self.ui.mdiArea.tileSubWindows()

def SubWindowView(self):
    self.ui.mdiArea.setViewMode(0)

def TabbedView(self):
    self.ui.mdiArea.setViewMode(1)
```

```
*callMDI.pyw -
File Edit Format Run Options Window Help
import sys
from MDIdemo import *

class MyForm(QtGui.QMainWindow):
    def __init__(self, parent=None):
        QtGui.QWidget.__init__(self, parent)
        self.ui = Ui_MainWindow()
        self.ui.setupUi(self)
        self.ui.mdiArea.addSubWindow(self.ui.subwindow)
        self.ui.mdiArea.addSubWindow(self.ui.subwindow_2)

        QtCore.QObject.connect(self.ui.showNext, QtCore.SIGNAL('clicked()'), self.displayNext)
        QtCore.QObject.connect(self.ui.showPrevious, QtCore.SIGNAL('clicked()'), self.displayPrevious)
        QtCore.QObject.connect(self.ui.closeAll, QtCore.SIGNAL('clicked()'), self.closeAll)
        QtCore.QObject.connect(self.ui.cascadeButton, QtCore.SIGNAL('clicked()'), self.cascadeArrange)
        QtCore.QObject.connect(self.ui.tileButton, QtCore.SIGNAL('clicked()'), self.tileArrange)
        QtCore.QObject.connect(self.ui.SubWindowViewButton, QtCore.SIGNAL('clicked()'), self.SubWindowView)
        QtCore.QObject.connect(self.ui.TabbedViewButton, QtCore.SIGNAL('clicked()'), self.TabbedView)
        self.connect(self.ui.actionFirst_Window, QtCore.SIGNAL('triggered()'), self.displayNext)
        self.connect(self.ui.actionSecond_Window, QtCore.SIGNAL('triggered()'), self.displayPrevious)

    def displayNext(self):
        self.ui.mdiArea.activateNextSubWindow()

    def displayPrevious(self):
        self.ui.mdiArea.activatePreviousSubWindow()

    def closeAll(self):
        self.ui.mdiArea.closeAllSubWindows()

    def cascadeArrange(self):
        self.ui.mdiArea.cascadeSubWindows()

    def tileArrange(self):
        self.ui.mdiArea.tileSubWindows()

    def SubWindowView(self):
        self.ui.mdiArea.setViewMode(0)

    def TabbedView(self):
        self.ui.mdiArea.setViewMode(1)

if __name__ == "__main__":
    app = QtGui.QApplication(sys.argv)
    myapp = MyForm()
    myapp.show()
    sys.exit(app.exec_())
```

- Save the file as callMDI.pyw
- Run and test the application.