# PySide6 – “Qt for Python”

## A Python wrapper for Qt version 6

Hamilton Python Users Group
Ian Stewart
10 Oct 2022

# PySide6

Presentation Overview:

- GUI Toolkits and their Python wrappers.
- History of Qt and PySide
- Installation of PySide/Qt6
- Review of what is installed.
- Utilities
- Introduce pyside6-assistant.py program
- Demo pyside6-assistant.py.
- Code highlights / bugs in pyside6-assistant.py

# Open Source – GUI Tookits.

| GUI Toolkit | Language | Version | Python Bindings |
|---|---|---|---|
| Tk/Tcl | C | 8.6 | Tkinter |
| Gtk | C | 4 | PyGObject Introspection – GIR - GTK4 |
| Qt | C++ | 6.4 | 1. PyQt6 – Riverbank Computing.<br>2. PySide6 – The Qt Company. |

# Qt History

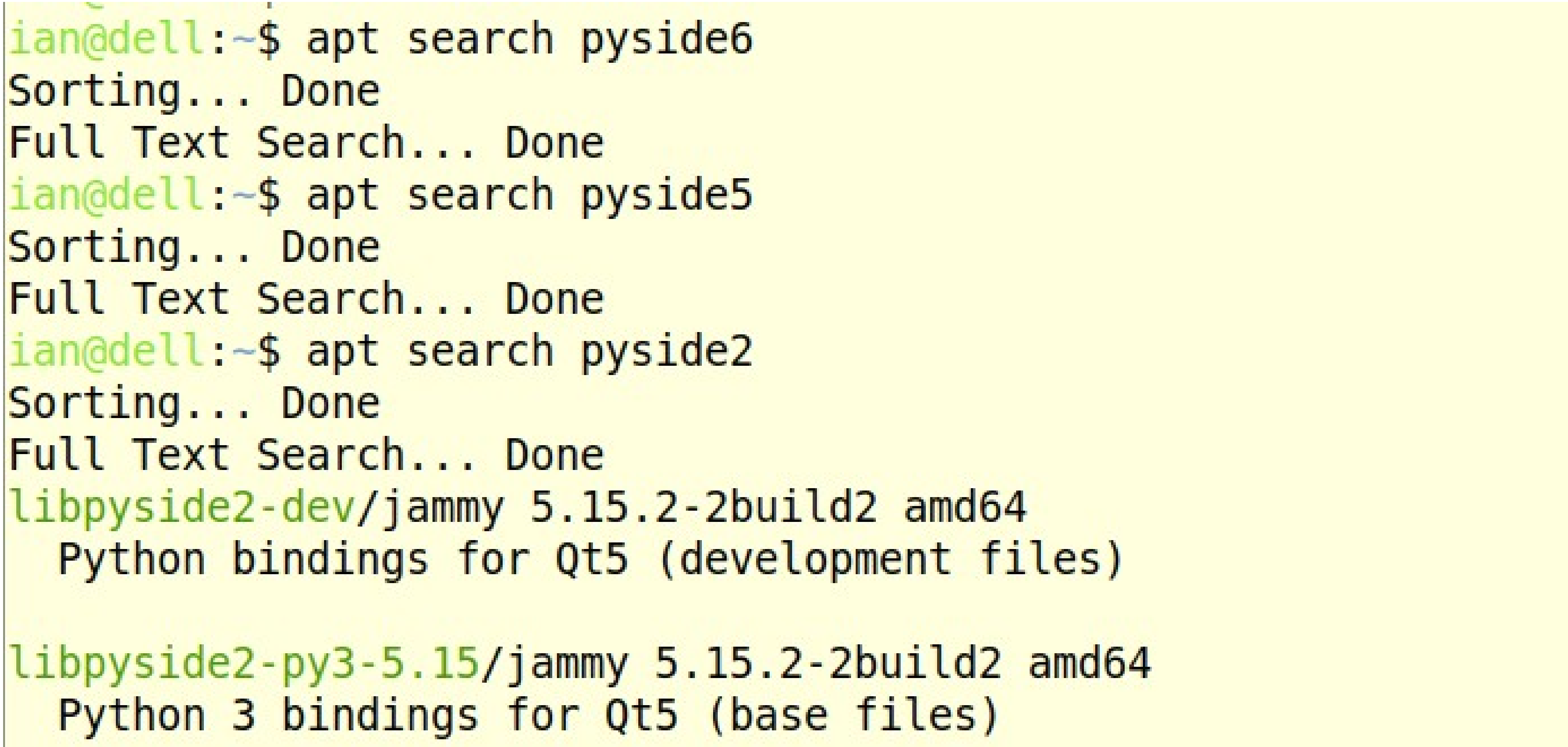
- 1991 – Haavard Nord and Eirik Chambe-Eng from Norway, started writing Qt.
- 1994 – founded Trolltech
- 2006 – IPO of Trolltech
- 2008 – Nokia Corporation acquire Trolltech
- 2008 – Trolltech renamed Qt Software
- 2009 – Renamed Qt Development Frameworks
- 2012 – Nokia sold to Digia
- 2014 – Digia formed “The Qt Company” wholly owned subsidiary.
- 2016 – The Qt Company went public on NASDAQ Helsinki.
- Headquarters – Espoo Finland.
- 445 Employees world wide.
- Website: www.qt.io
- Repository: https://github.com/qt/

# PySide History.

- 2009 – Initial Release by Qt.
- 'Side' is Finnish for binding.
- 2022-09-12 – PySide6 – V6.3.2 currently available on PyPI
- PySide supports Linux, macOS, and Microsoft Windows.
- “PySide” supports Qt 4
- 2015 - “PySide2” to support Qt 5.
- 2020 – Released “PySide6” supports Qt 6 – for Python V3.6+
- “PySide” marketed as “Qt for Python”.
- https://wiki.qt.io/Qt_for_Python
- Tutorials: https://doc.qt.io/qtforpython/tutorials/index.html

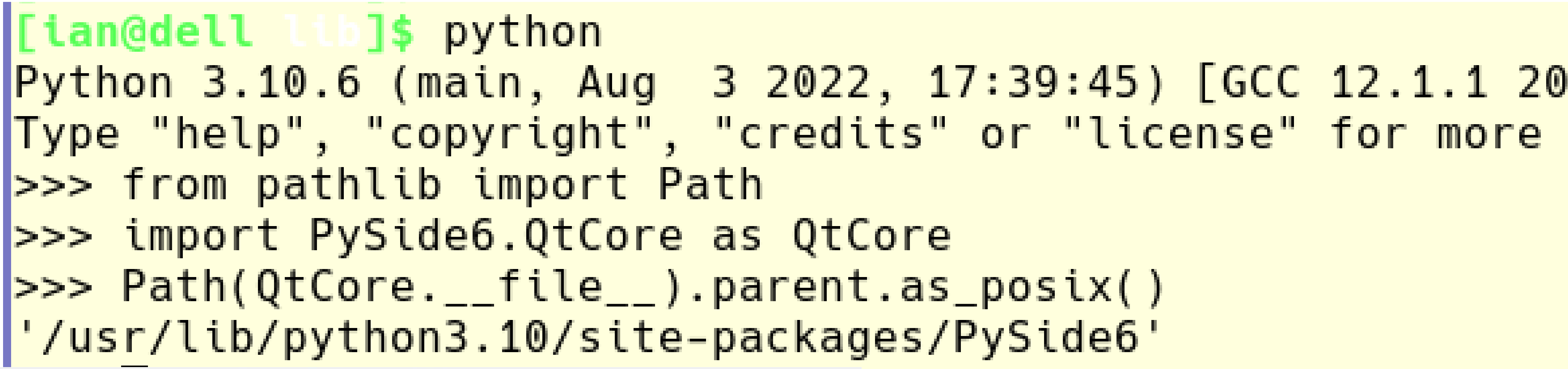
# PySide6/Qt  Installation

# PySide6 Install – Is Pyside6 in distro's repository? E.g. Not in 22.04 Ubuntu...

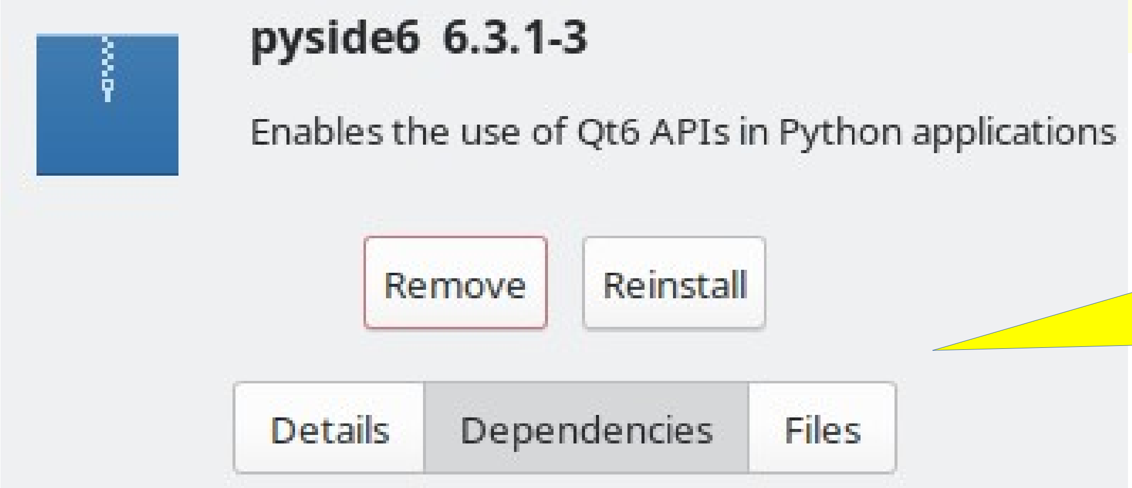


```
ian@dell:~$ apt search pyside6
Sorting... Done
Full Text Search... Done
ian@dell:~$ apt search pyside5
Sorting... Done
Full Text Search... Done
ian@dell:~$ apt search pyside2
Sorting... Done
Full Text Search... Done
libpyside2-dev/jammy 5.15.2-2build2 amd64
  Python bindings for Qt5 (development files)

libpyside2-py3-5.15/jammy 5.15.2-2build2 amd64
  Python 3 bindings for Qt5 (base files)
```

# PySide6 Install – Is Pyside6 in distro’s repository? E.g. Mangaro KDE includes PySide6 in /usr/lib...



```
[ian@dell lib]$ python
Python 3.10.6 (main, Aug  3 2022, 17:39:45) [GCC 12.1.1 20
Type "help", "copyright", "credits" or "license" for more
>>> from pathlib import Path
>>> import PySide6.QtCore as QtCore
>>> Path(QtCore.__file__).parent.as_posix()
'/usr/lib/python3.10/site-packages/PySide6'
```

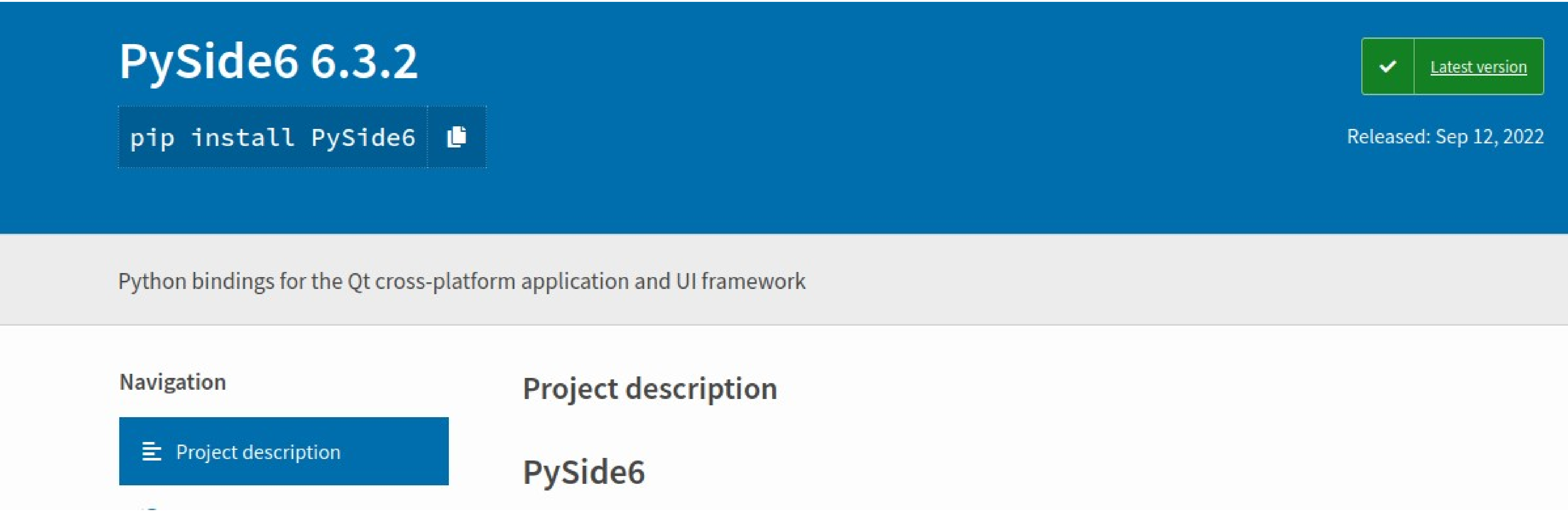


**pyside6 6.3.1-3**

Enables the use of Qt6 APIs in Python applications

Pacman: About half of the PySide6 modules installed by default.

# PySide6 Install – Availble in PyPI



## PySide6 6.3.2

```
pip install PySide6
```

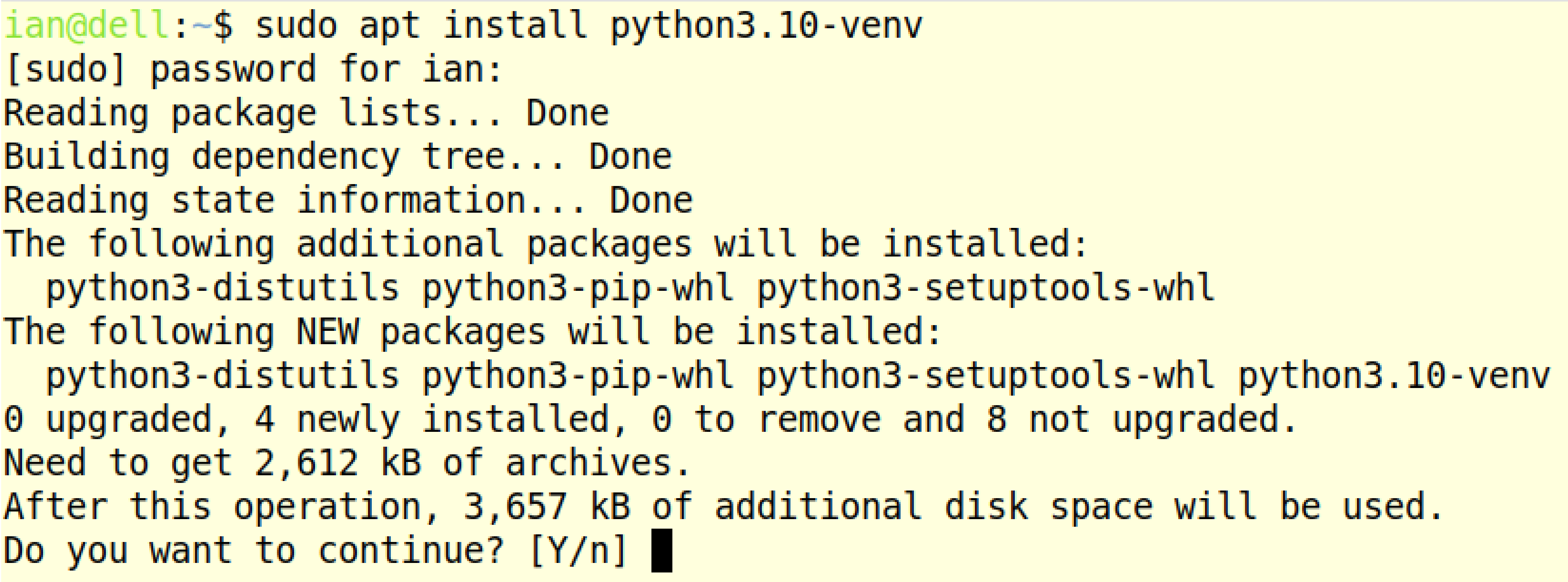
Released: Sep 12, 2022

Python bindings for the Qt cross-platform application and UI framework

### Project description

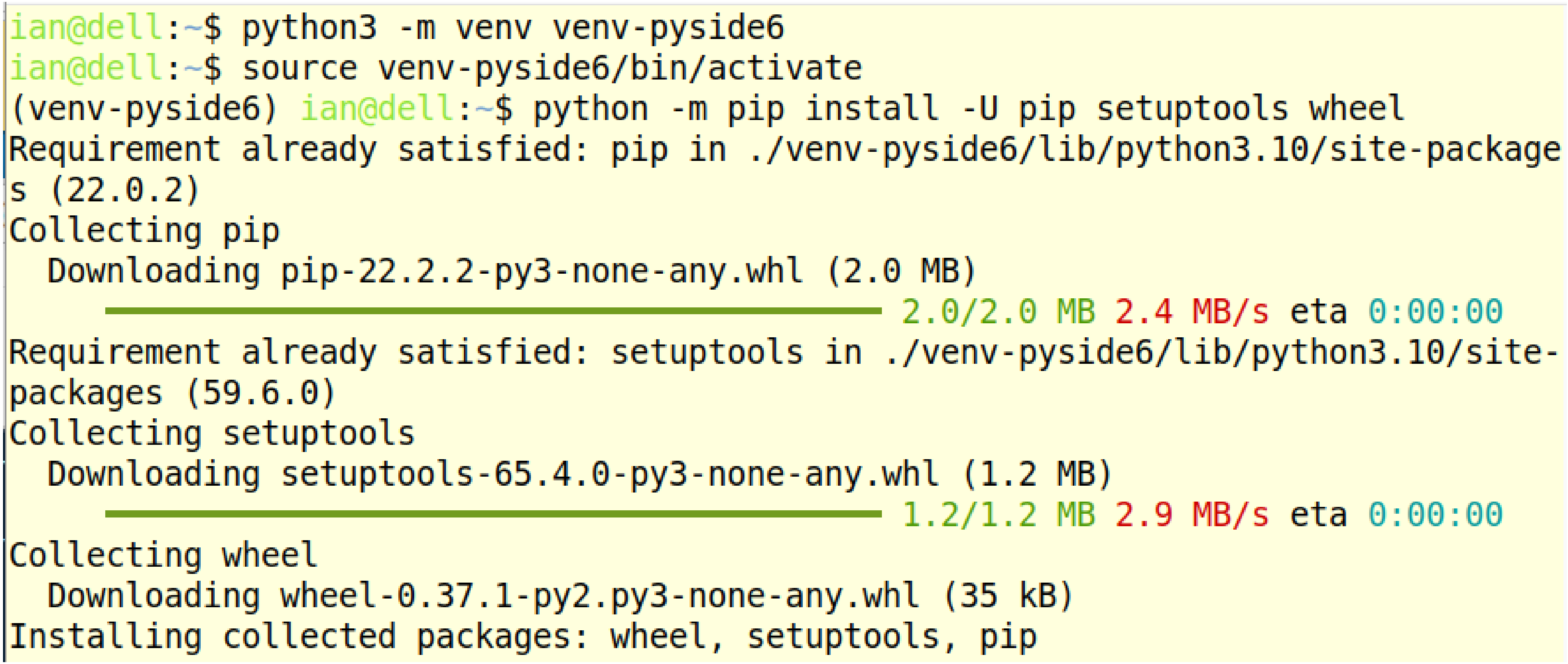
### PySide6

# PySide6 Install – Create Virtual Environment



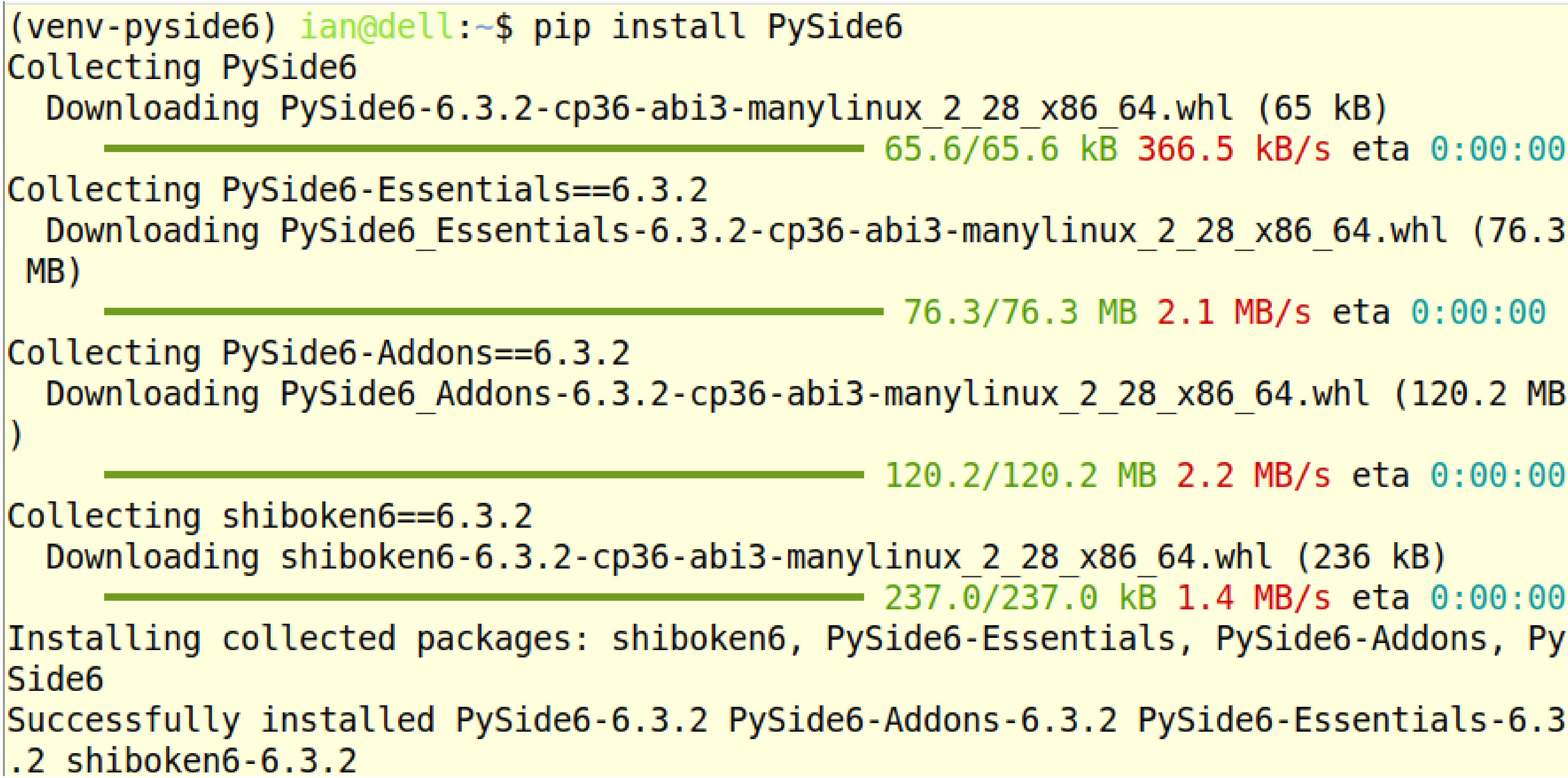
```
ian@dell:~$ sudo apt install python3.10-venv
[sudo] password for ian:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-distutils python3-pip-whl python3-setuptools-whl
The following NEW packages will be installed:
  python3-distutils python3-pip-whl python3-setuptools-whl python3.10-venv
0 upgraded, 4 newly installed, 0 to remove and 8 not upgraded.
Need to get 2,612 kB of archives.
After this operation, 3,657 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

# PySide6 Install – Create Virtual Environment



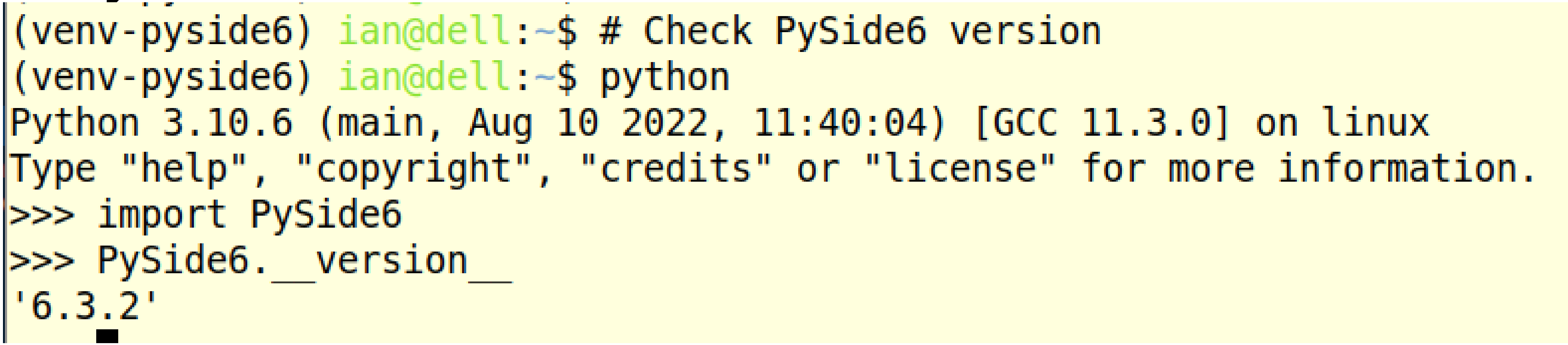
```
ian@dell:~$ python3 -m venv venv-pyside6
ian@dell:~$ source venv-pyside6/bin/activate
(venv-pyside6) ian@dell:~$ python -m pip install -U pip setuptools wheel
Requirement already satisfied: pip in ./venv-pyside6/lib/python3.10/site-package
s (22.0.2)
Collecting pip
  Downloading pip-22.2.2-py3-none-any.whl (2.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.0/2.0 MB 2.4 MB/s eta 0:00:00
Requirement already satisfied: setuptools in ./venv-pyside6/lib/python3.10/site-
packages (59.6.0)
Collecting setuptools
  Downloading setuptools-65.4.0-py3-none-any.whl (1.2 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.2/1.2 MB 2.9 MB/s eta 0:00:00
Collecting wheel
  Downloading wheel-0.37.1-py2.py3-none-any.whl (35 kB)
Installing collected packages: wheel, setuptools, pip
```

# PySide6 Install – Create Virtual Environment



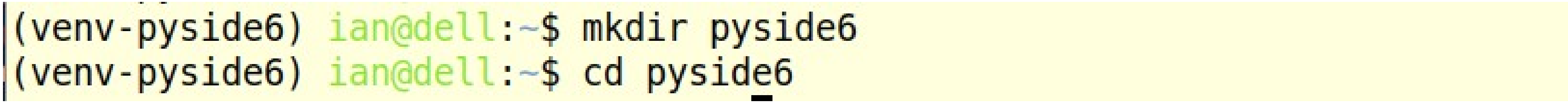
```
(venv-pyside6) ian@dell:~$ pip install PySide6
Collecting PySide6
  Downloading PySide6-6.3.2-cp36-abi3-manylinux_2_28_x86_64.whl (65 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 65.6/65.6 kB 366.5 kB/s eta 0:00:00
Collecting PySide6-Essentials==6.3.2
  Downloading PySide6_Essentials-6.3.2-cp36-abi3-manylinux_2_28_x86_64.whl (76.3
 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 76.3/76.3 MB 2.1 MB/s eta 0:00:00
Collecting PySide6-Addons==6.3.2
  Downloading PySide6_Addons-6.3.2-cp36-abi3-manylinux_2_28_x86_64.whl (120.2 MB
)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 120.2/120.2 MB 2.2 MB/s eta 0:00:00
Collecting shiboken6==6.3.2
  Downloading shiboken6-6.3.2-cp36-abi3-manylinux_2_28_x86_64.whl (236 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 237.0/237.0 kB 1.4 MB/s eta 0:00:00
Installing collected packages: shiboken6, PySide6-Essentials, PySide6-Addons, Py
Side6
Successfully installed PySide6-6.3.2 PySide6-Addons-6.3.2 PySide6-Essentials-6.3
.2 shiboken6-6.3.2
```

# PySide6 Install – Check Version



```
(venv-pyside6) ian@dell:~$ # Check PySide6 version
(venv-pyside6) ian@dell:~$ python
Python 3.10.6 (main, Aug 10 2022, 11:40:04) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import PySide6
>>> PySide6.__version__
'6.3.2'
```

Make a working directory...



```
(venv-pyside6) ian@dell:~$ mkdir pyside6
(venv-pyside6) ian@dell:~$ cd pyside6
```

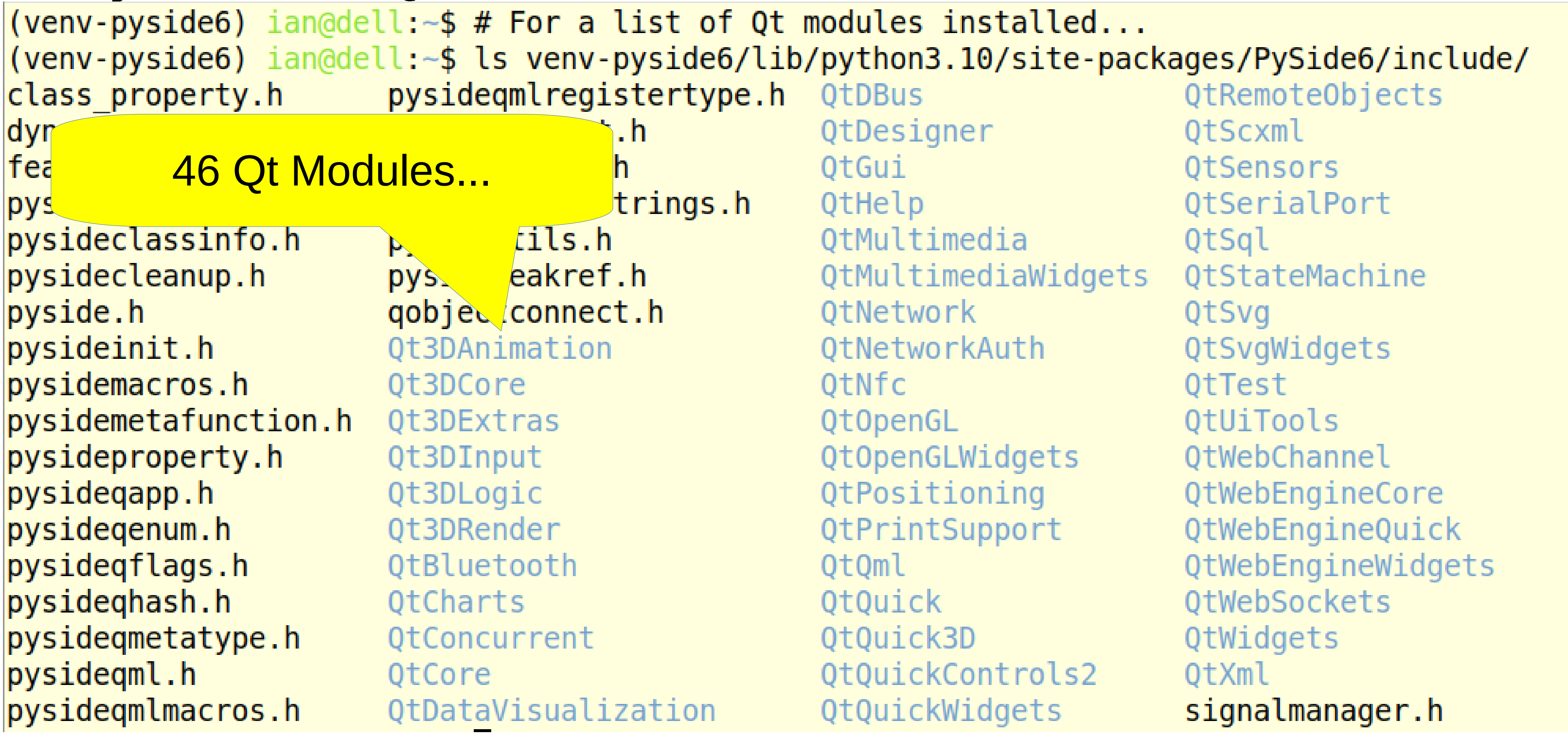
# PySide6 Install – What’s downloaded?...

- PySide6 Library containing 46 Python modules.
- Help on all Modules, Classes, attributes.
- Icons.
- Example Python/PySide6 programs.
- Utilities. Including:
    - pyside6-designer. Use “ui loader”.
    - pyside6-uic. User Interface Compiler.
    - pyside6-rcc. Resource Compiler.
    - ...more...

# PySide6 – Naming conventions...

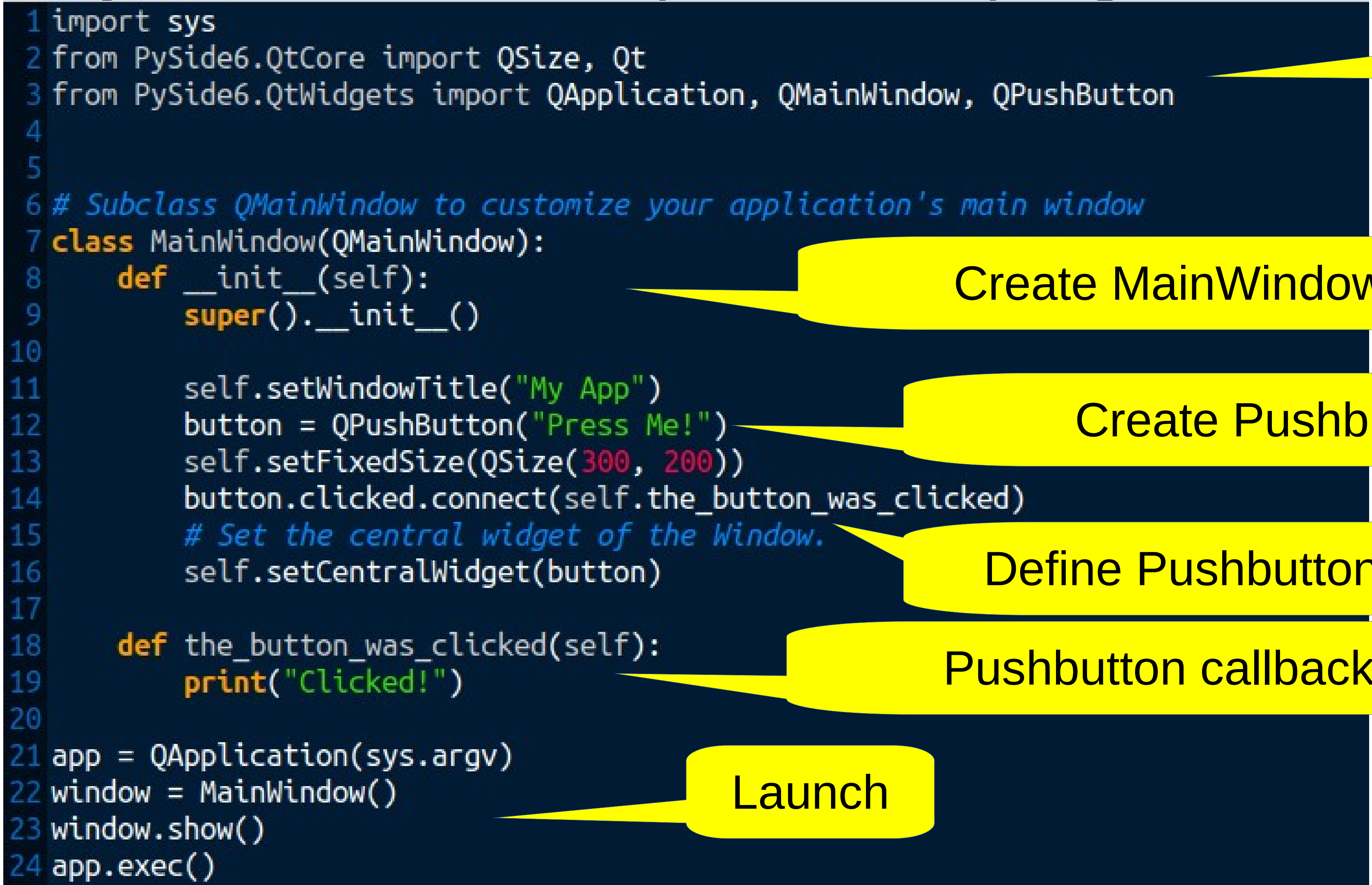
- PySide6 Library.
- “Qt” – Prefix for a Module. E.g. QtWidgets
- “Q” – Prefix for a Class. E.g. QPushButton
- Attributes – lower camel case. E.g. .setText()

# PySide6 – Qt Modules installed



```
(venv-pyside6) ian@dell:~$ # For a list of Qt modules installed...
(venv-pyside6) ian@dell:~$ ls venv-pyside6/lib/python3.10/site-packages/PySide6/include/
class_property.h      pysideqmlregistertype.h  QtDBus               QtRemoteObjects
dyn                                .h          QtDesigner           QtScxml
fea                                 h          QtGui                QtSensors
pys                                trings.h    QtHelp               QtSerialPort
pysideclassinfo.h     p      tils.h            QtMultimedia         QtSql
pysidecleanup.h       pys   eakref.h           QtMultimediaWidgets  QtStateMachine
pyside.h              qobje  connect.h         QtNetwork            QtSvg
pysideinit.h          Qt3DAnimation            QtNetworkAuth        QtSvgWidgets
pysidemacros.h        Qt3DCore                 QtNfc                QtTest
pysidemetafunction.h  Qt3DExtras               QtOpenGL             QtUiTools
pysideproperty.h      Qt3DInput                QtOpenGLWidgets      QtWebChannel
pysideqapp.h          Qt3DLogic                QtPositioning        QtWebEngineCore
pysideqenum.h         Qt3DRender               QtPrintSupport       QtWebEngineQuick
pysideqflags.h        QtBluetooth              QtQml                QtWebEngineWidgets
pysideqhash.h         QtCharts                 QtQuick              QtWebSockets
pysideqmetatype.h     QtConcurrent             QtQuick3D            QtWidgets
pysideqml.h           QtCore                   QtQuickControls2     QtXml
pysideqmlmacros.h     QtDataVisualization      QtQuickWidgets       signalmanager.h
```

# PySide6 – Write a “press me” program



```python
import sys
from PySide6.QtCore import QSize, Qt
from PySide6.QtWidgets import QApplication, QMainWindow, QPushButton


# Subclass QMainWindow to customize your application's main window
class MainWindow(QMainWindow):
    def __init__(self):
        super().__init__()

        self.setWindowTitle("My App")
        button = QPushButton("Press Me!")
        self.setFixedSize(QSize(300, 200))
        button.clicked.connect(self.the_button_was_clicked)
        # Set the central widget of the Window.
        self.setCentralWidget(button)

    def the_button_was_clicked(self):
        print("Clicked!")

app = QApplication(sys.argv)
window = MainWindow()
window.show()
app.exec()
```

Importing

Create MainWindow

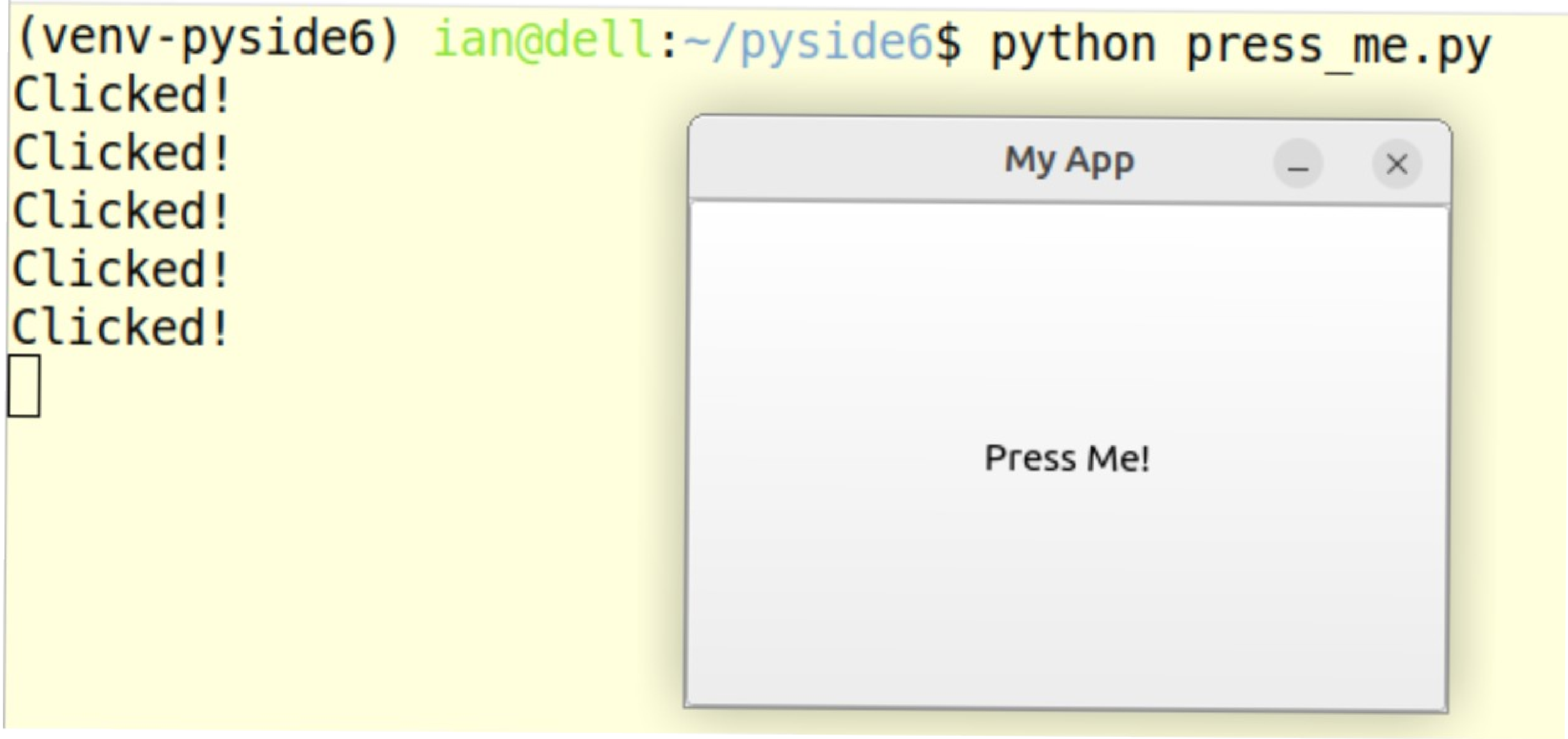
Create Pushbutton

Define Pushbutton callback

Pushbutton callback

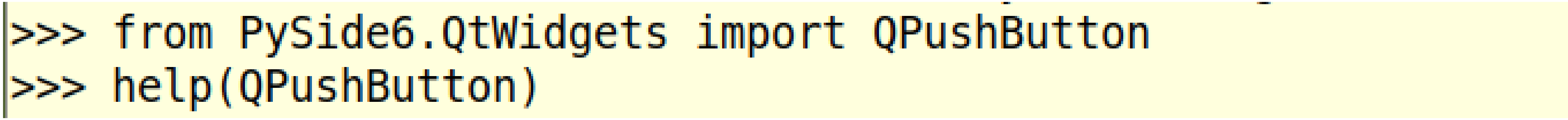
Launch

# PySide6 – “Press Me” program...

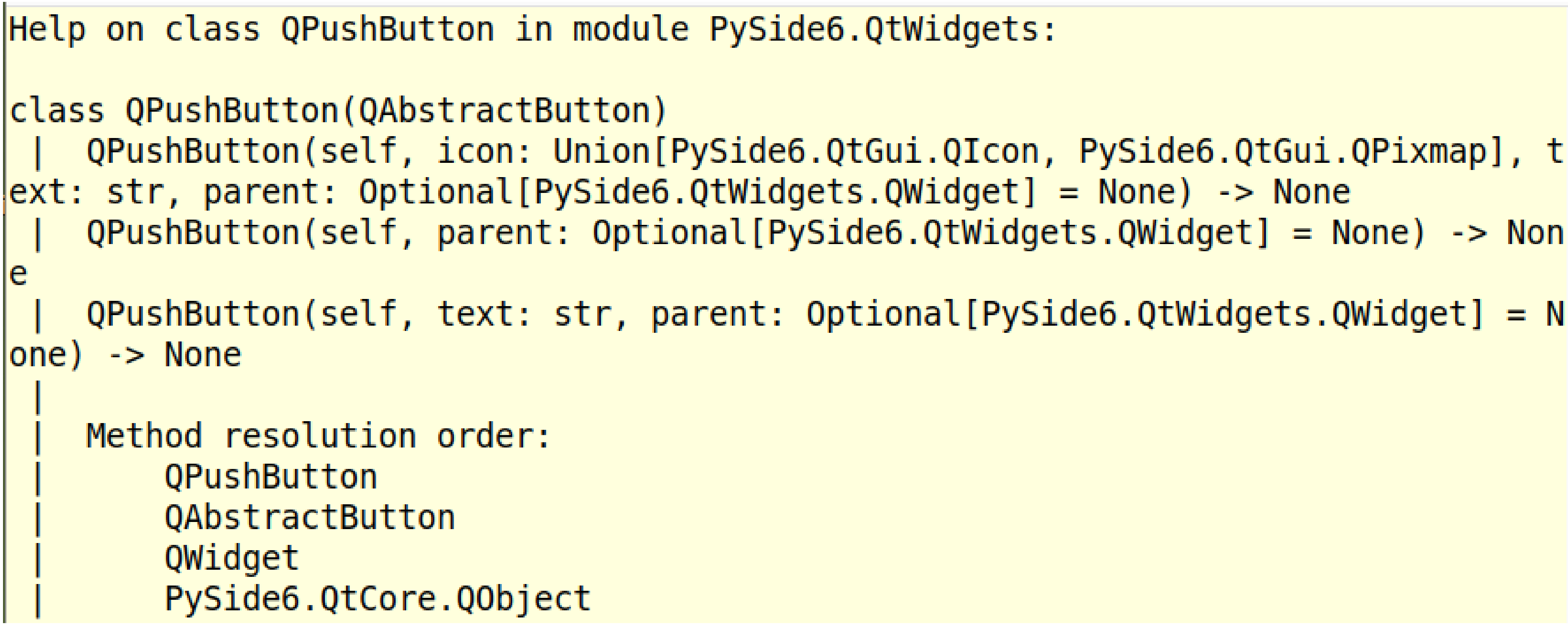


```
(venv-pyside6) ian@dell:~/pyside6$ python press_me.py
Clicked!
Clicked!
Clicked!
Clicked!
Clicked!
```

# PySide6 – help()...



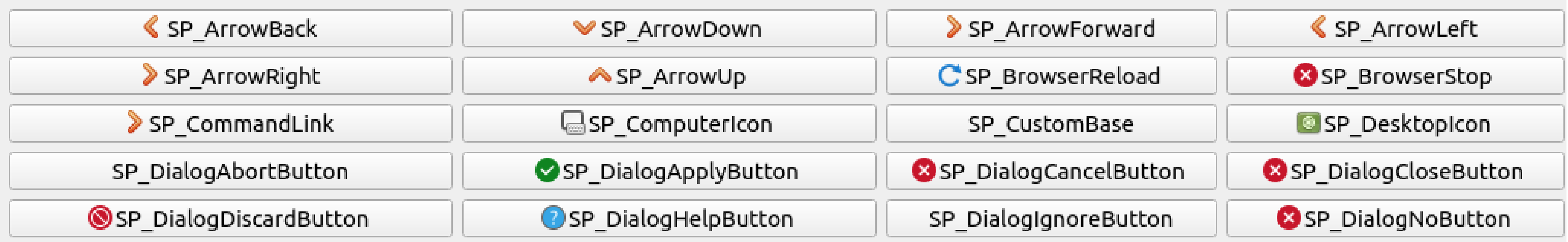
```
>>> from PySide6.QtWidgets import QPushButton
>>> help(QPushButton)
```



```
Help on class QPushButton in module PySide6.QtWidgets:

class QPushButton(QAbstractButton)
 |  QPushButton(self, icon: Union[PySide6.QtGui.QIcon, PySide6.QtGui.QPixmap], t
ext: str, parent: Optional[PySide6.QtWidgets.QWidget] = None) -> None
 |  QPushButton(self, parent: Optional[PySide6.QtWidgets.QWidget] = None) -> Non
e
 |  QPushButton(self, text: str, parent: Optional[PySide6.QtWidgets.QWidget] = N
one) -> None
 |
 |  Method resolution order:
 |      QPushButton
 |      QAbstractButton
 |      QWidget
 |      PySide6.QtCore.QObject
```

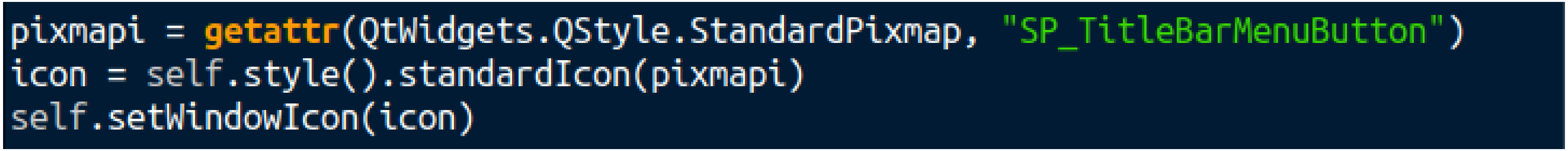
# PySide6 – Icons...

## About 80 icons are supplied. E.g.



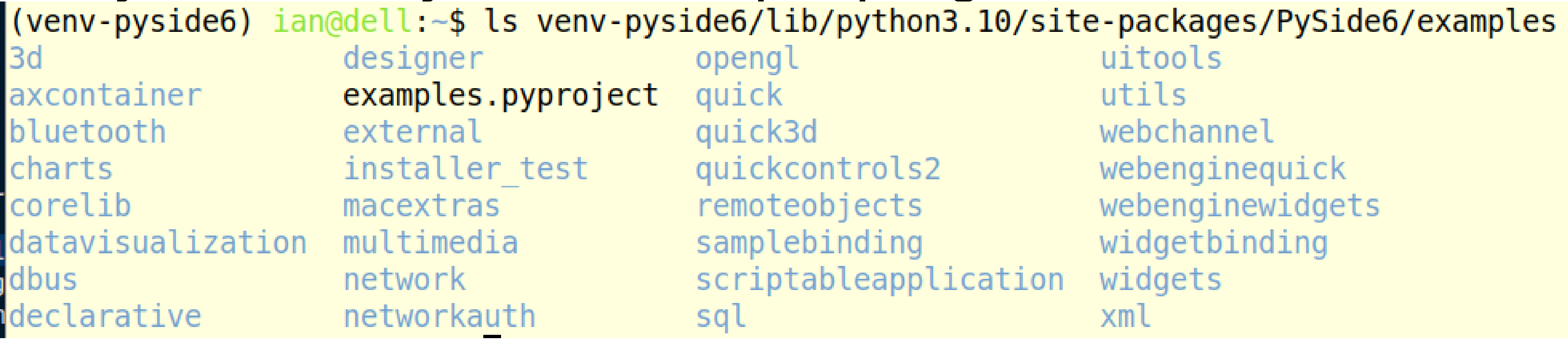
| | | | |
|---|---|---|---|
| SP_ArrowBack | SP_ArrowDown | SP_ArrowForward | SP_ArrowLeft |
| SP_ArrowRight | SP_ArrowUp | SP_BrowserReload | SP_BrowserStop |
| SP_CommandLink | SP_ComputerIcon | SP_CustomBase | SP_DesktopIcon |
| SP_DialogAbortButton | SP_DialogApplyButton | SP_DialogCancelButton | SP_DialogCloseButton |
| SP_DialogDiscardButton | SP_DialogHelpButton | SP_DialogIgnoreButton | SP_DialogNoButton |

## Example code for adding supplied icon...

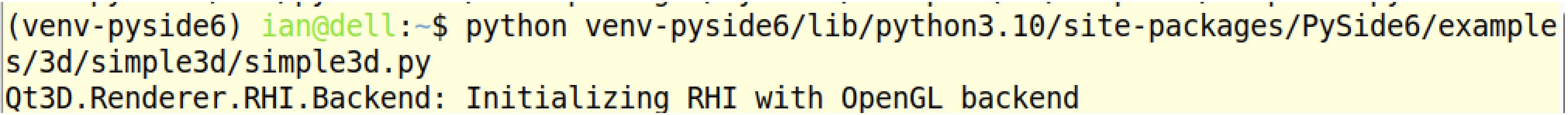


```
pixmapi = getattr(QtWidgets.QStyle.StandardPixmap, "SP_TitleBarMenuButton")
icon = self.style().standardIcon(pixmapi)
self.setWindowIcon(icon)
```

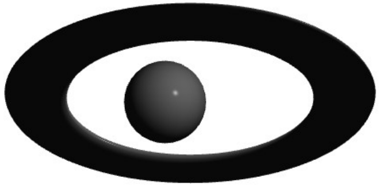
# PySide6 – Python Example programs...



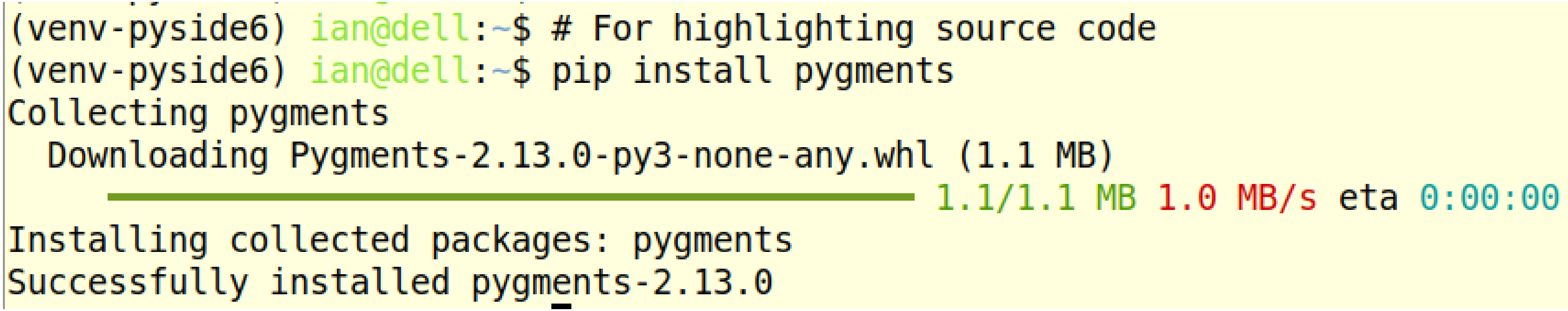
```
(venv-pyside6) ian@dell:~$ ls venv-pyside6/lib/python3.10/site-packages/PySide6/examples
3d                 designer            opengl                 uitools
axcontainer        examples.pyproject  quick                  utils
bluetooth          external            quick3d                webchannel
charts             installer_test      quickcontrols2         webenginequick
corelib            macextras           remoteobjects          webenginewidgets
datavisualization  multimedia          samplebinding          widgetbinding
dbus               network             scriptableapplication  widgets
declarative        networkauth         sql                    xml
```



```
(venv-pyside6) ian@dell:~$ python venv-pyside6/lib/python3.10/site-packages/PySide6/example
s/3d/simple3d/simple3d.py
Qt3D.Renderer.RHI.Backend: Initializing RHI with OpenGL backend
```

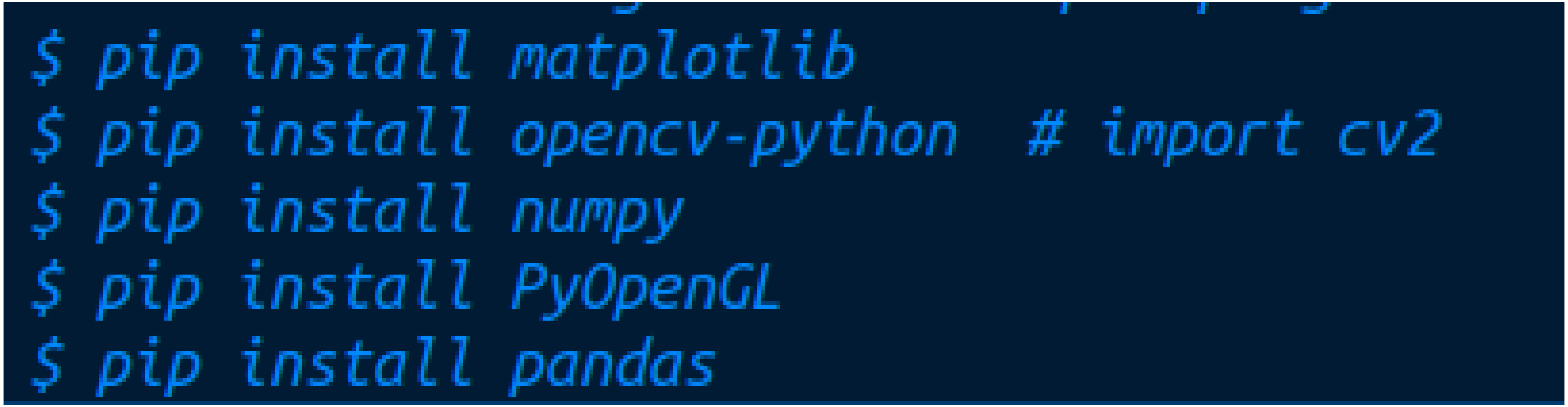


# PySide6 Install – Additional Python modules...



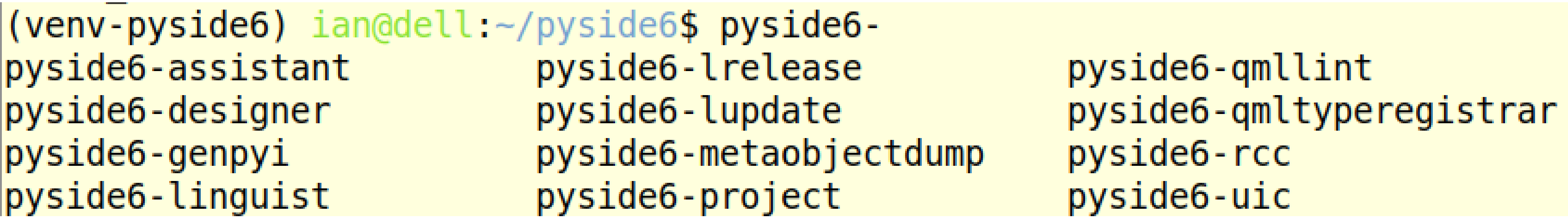
```
(venv-pyside6) ian@dell:~$ # For highlighting source code
(venv-pyside6) ian@dell:~$ pip install pygments
Collecting pygments
  Downloading Pygments-2.13.0-py3-none-any.whl (1.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.1/1.1 MB 1.0 MB/s eta 0:00:00
Installing collected packages: pygments
Successfully installed pygments-2.13.0
```

## Additional Python modules to run examples...



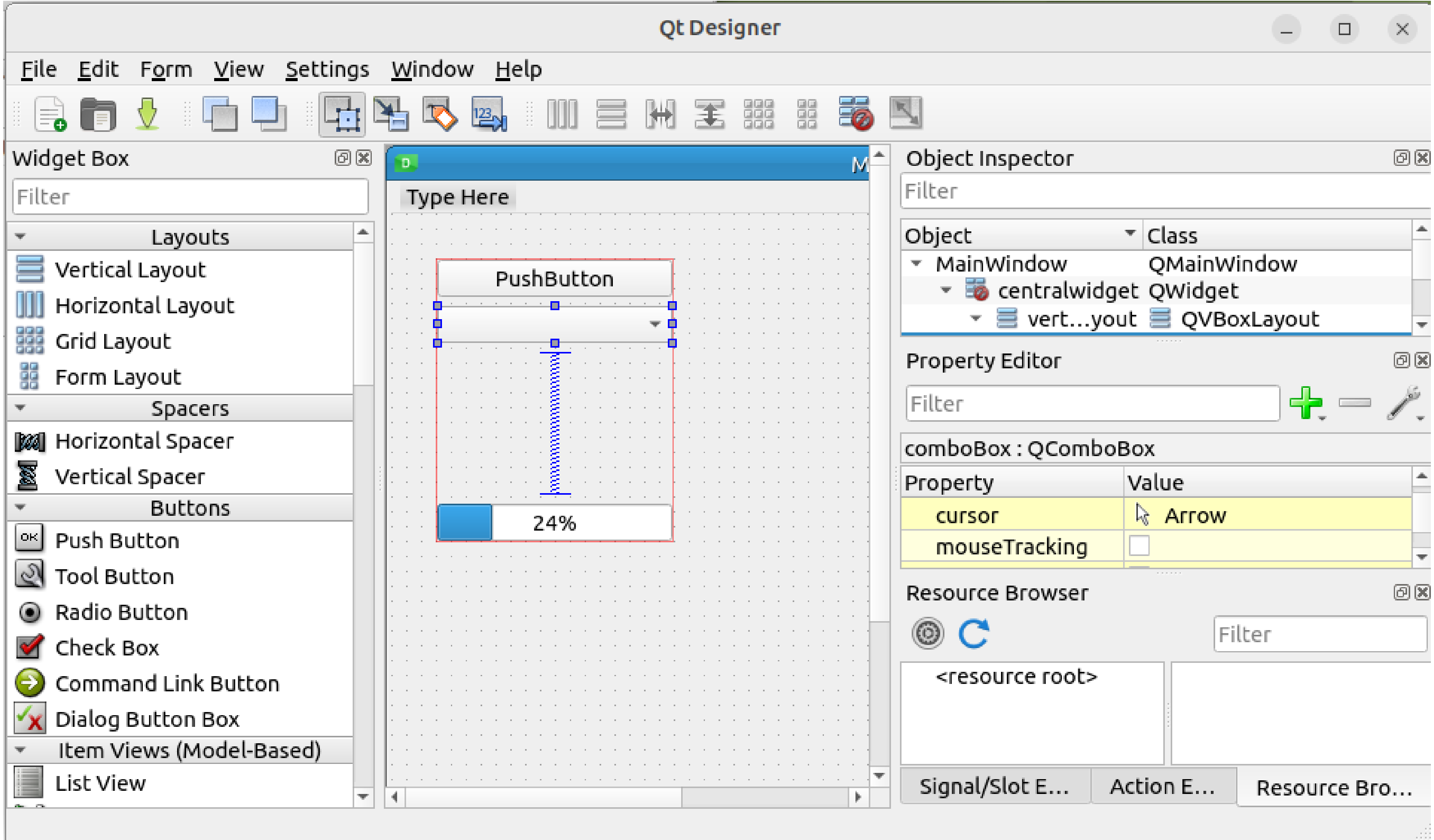
```
$ pip install matplotlib
$ pip install opencv-python  # import cv2
$ pip install numpy
$ pip install PyOpenGL
$ pip install pandas
```

# PySide6 – Utilities. Executables Installed...



```
(venv-pyside6) ian@dell:~/pyside6$ pyside6-
pyside6-assistant         pyside6-lrelease          pyside6-qmllint
pyside6-designer          pyside6-lupdate           pyside6-qmltyperegistrar
pyside6-genpyi            pyside6-metaobjectdump    pyside6-rcc
pyside6-linguist          pyside6-project           pyside6-uic
```

# PySide6 – pyside6-designer...

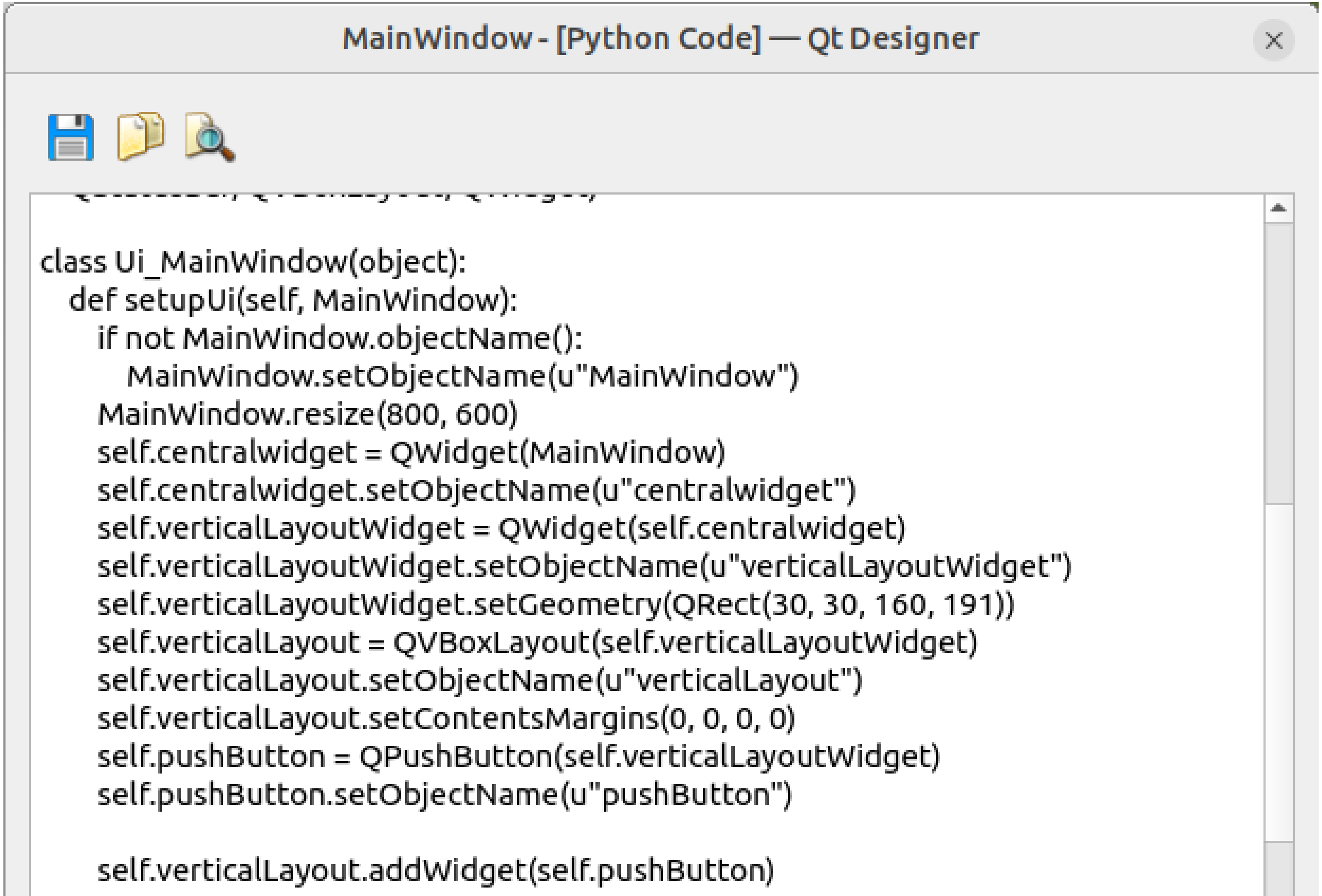


| Object | Class |
|---|---|
| MainWindow | QMainWindow |
| centralwidget | QWidget |
| vert...yout | QVBoxLayout |

comboBox : QComboBox

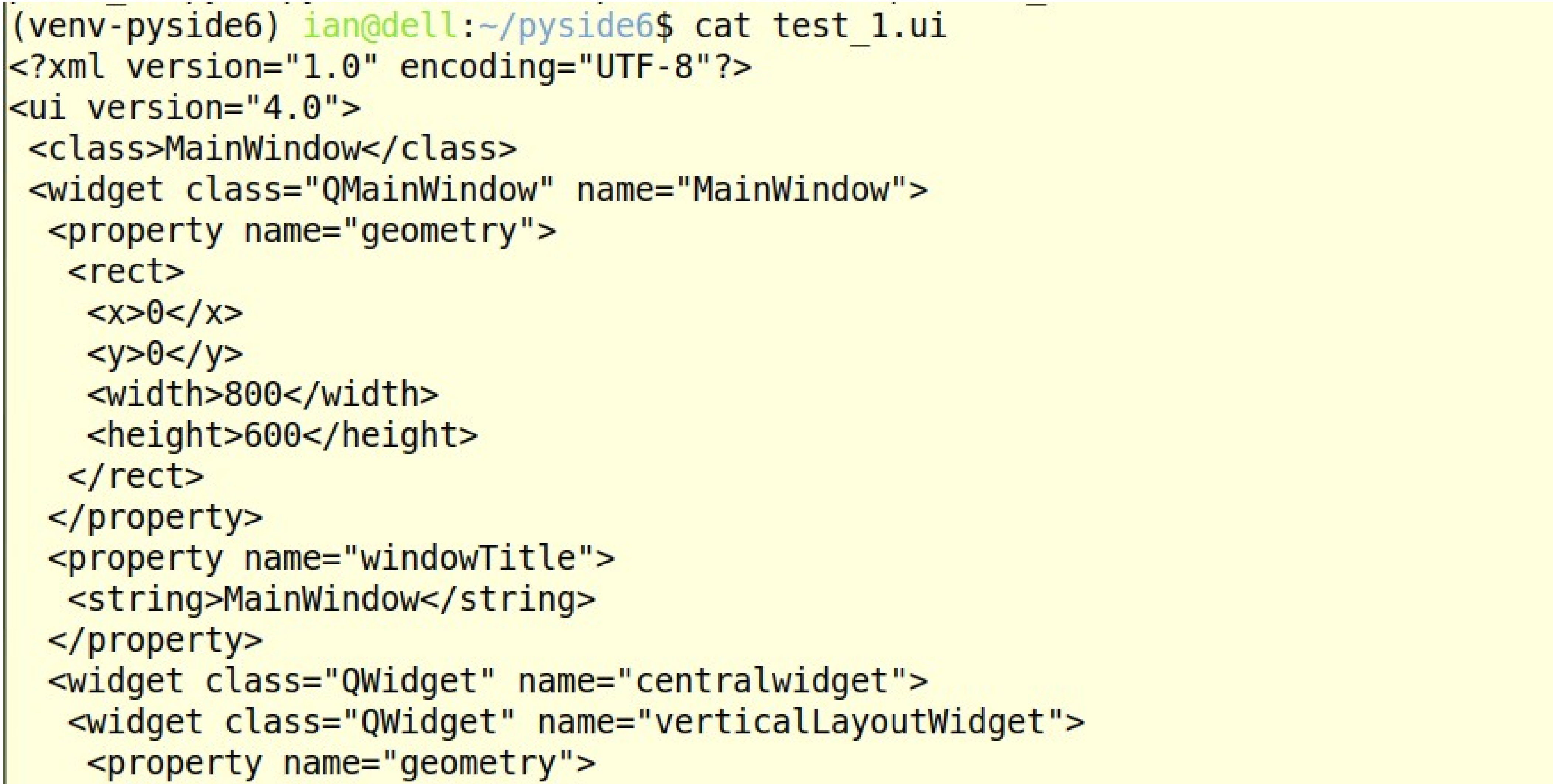
| Property | Value |
|---|---|
| cursor | Arrow |
| mouseTracking | □ |

# PySide6 – Designer Preview. Python code...



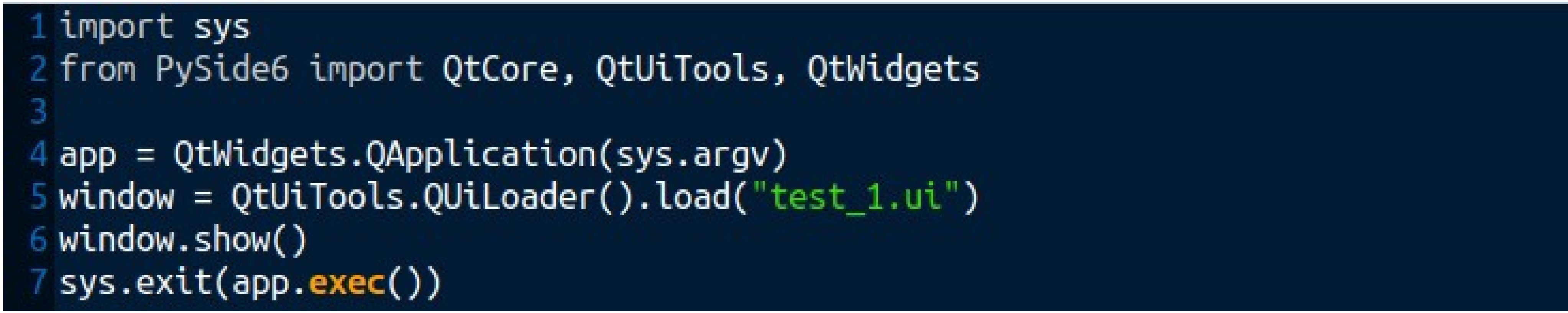
```
    QStatusBar, QVBoxLayout, QWidget)

class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        if not MainWindow.objectName():
            MainWindow.setObjectName(u"MainWindow")
        MainWindow.resize(800, 600)
        self.centralwidget = QWidget(MainWindow)
        self.centralwidget.setObjectName(u"centralwidget")
        self.verticalLayoutWidget = QWidget(self.centralwidget)
        self.verticalLayoutWidget.setObjectName(u"verticalLayoutWidget")
        self.verticalLayoutWidget.setGeometry(QRect(30, 30, 160, 191))
        self.verticalLayout = QVBoxLayout(self.verticalLayoutWidget)
        self.verticalLayout.setObjectName(u"verticalLayout")
        self.verticalLayout.setContentsMargins(0, 0, 0, 0)
        self.pushButton = QPushButton(self.verticalLayoutWidget)
        self.pushButton.setObjectName(u"pushButton")

        self.verticalLayout.addWidget(self.pushButton)
```

# PySide6 – Designer GUI saved as ui file...

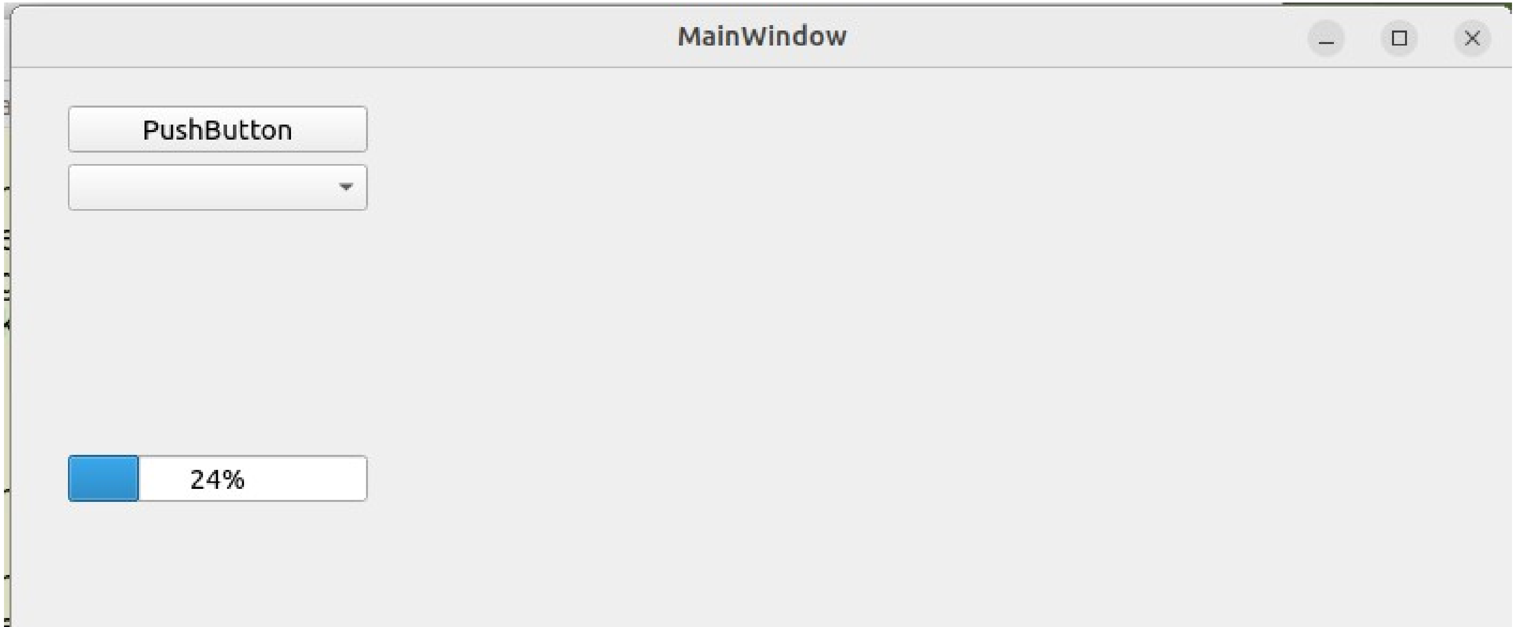


```
(venv-pyside6) ian@dell:~/pyside6$ cat test_1.ui
<?xml version="1.0" encoding="UTF-8"?>
<ui version="4.0">
 <class>MainWindow</class>
 <widget class="QMainWindow" name="MainWindow">
  <property name="geometry">
   <rect>
    <x>0</x>
    <y>0</y>
    <width>800</width>
    <height>600</height>
   </rect>
  </property>
  <property name="windowTitle">
   <string>MainWindow</string>
  </property>
  <widget class="QWidget" name="centralwidget">
   <widget class="QWidget" name="verticalLayoutWidget">
    <property name="geometry">
```

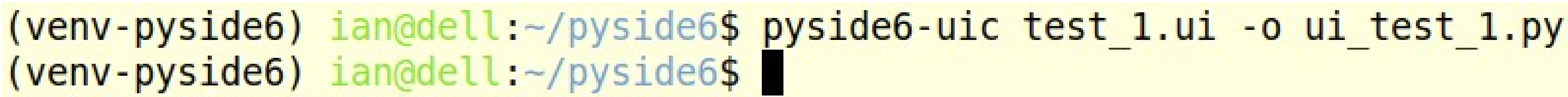
# PySide6 – Program to use ui file. Using “ui loader”...



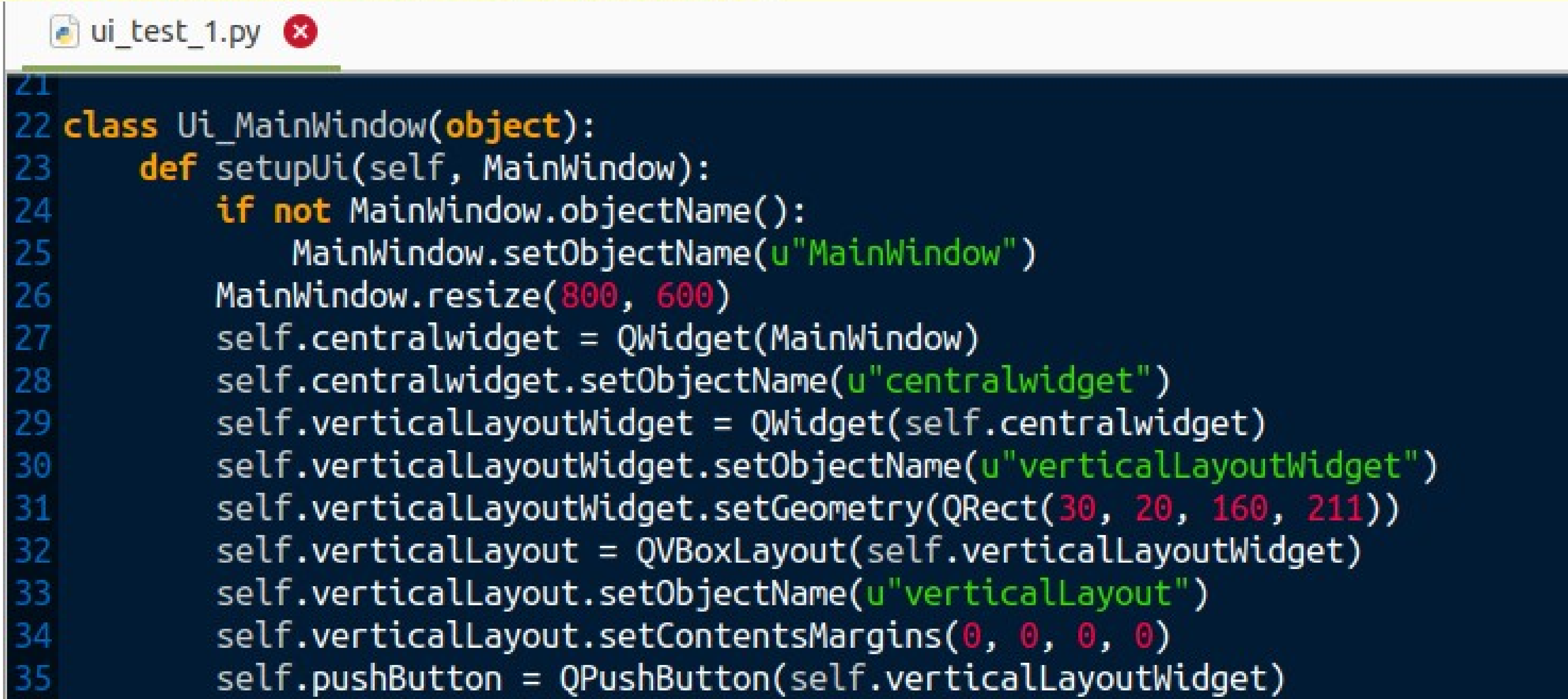
```python
import sys
from PySide6 import QtCore, QtUiTools, QtWidgets

app = QtWidgets.QApplication(sys.argv)
window = QtUiTools.QUiLoader().load("test_1.ui")
window.show()
sys.exit(app.exec())
```



# PySide6 – Alternative Method – Use “pyside6-uic” 1/2

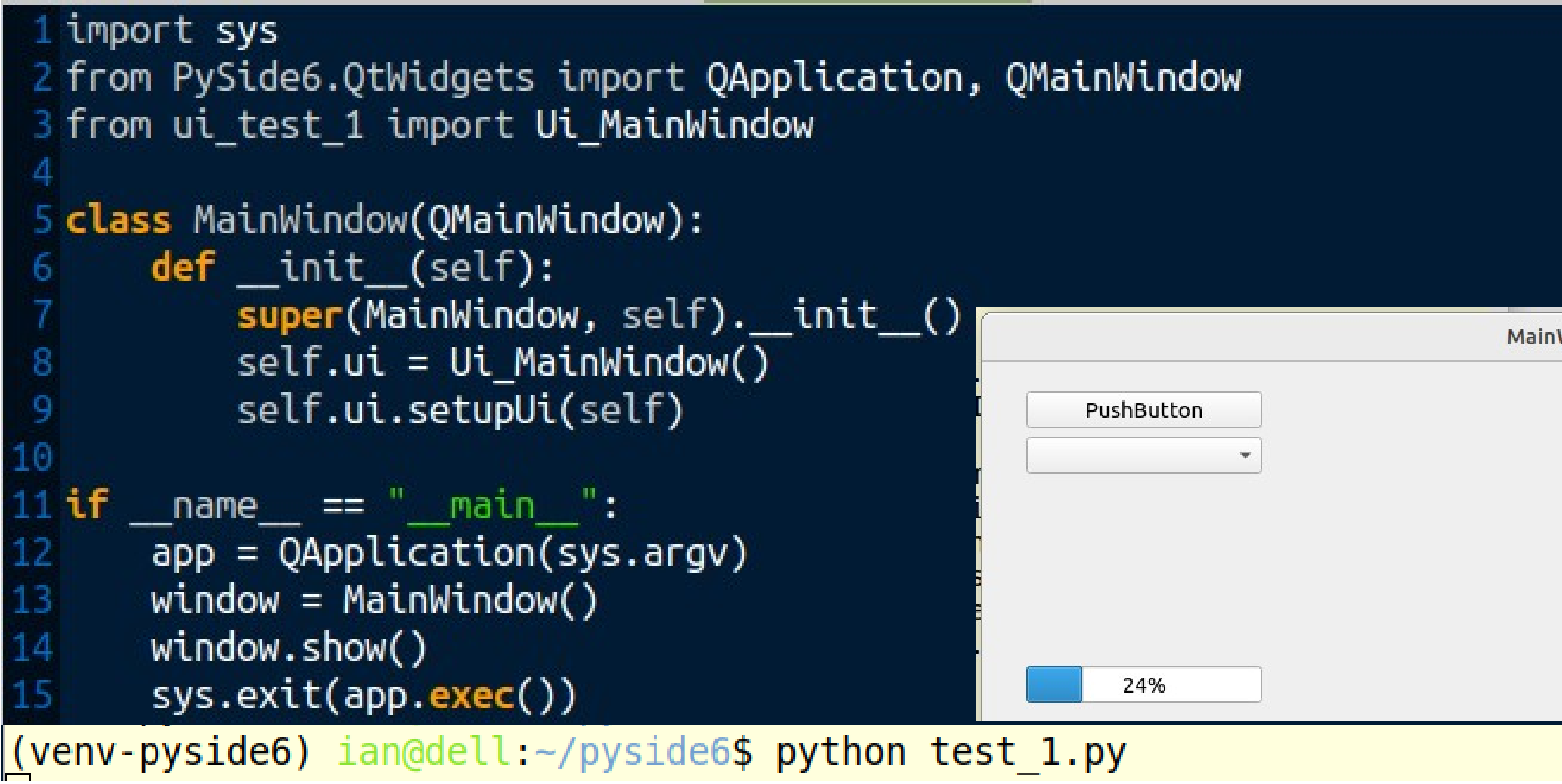


```
(venv-pyside6) ian@dell:~/pyside6$ pyside6-uic test_1.ui -o ui_test_1.py
(venv-pyside6) ian@dell:~/pyside6$ 
```



```
class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        if not MainWindow.objectName():
            MainWindow.setObjectName(u"MainWindow")
        MainWindow.resize(800, 600)
        self.centralwidget = QWidget(MainWindow)
        self.centralwidget.setObjectName(u"centralwidget")
        self.verticalLayoutWidget = QWidget(self.centralwidget)
        self.verticalLayoutWidget.setObjectName(u"verticalLayoutWidget")
        self.verticalLayoutWidget.setGeometry(QRect(30, 20, 160, 211))
        self.verticalLayout = QVBoxLayout(self.verticalLayoutWidget)
        self.verticalLayout.setObjectName(u"verticalLayout")
        self.verticalLayout.setContentsMargins(0, 0, 0, 0)
        self.pushButton = QPushButton(self.verticalLayoutWidget)
```

# PySide6 – test_1.py, importing ui-test_1 2/2



```python
import sys
from PySide6.QtWidgets import QApplication, QMainWindow
from ui_test_1 import Ui_MainWindow

class MainWindow(QMainWindow):
    def __init__(self):
        super(MainWindow, self).__init__()
        self.ui = Ui_MainWindow()
        self.ui.setupUi(self)

if __name__ == "__main__":
    app = QApplication(sys.argv)
    window = MainWindow()
    window.show()
    sys.exit(app.exec())
```

```
(venv-pyside6) ian@dell:~/pyside6$ python test_1.py
```

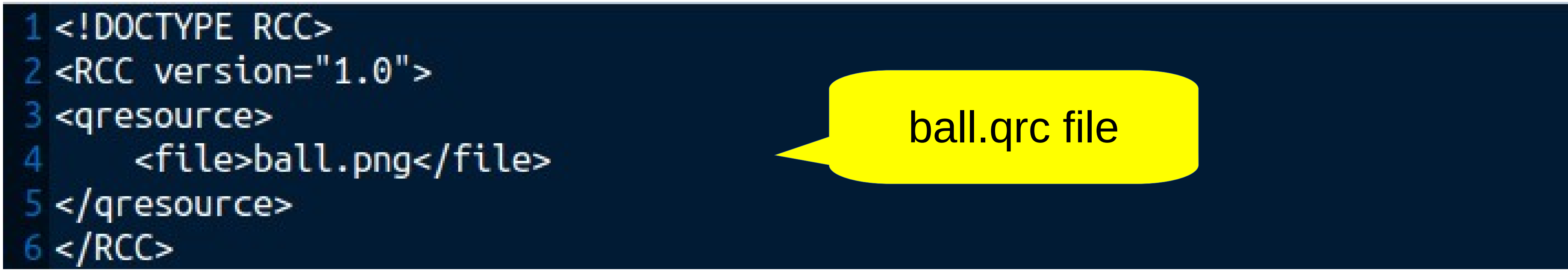
# PySide6 – Qt for Python documentation

Using .ui files from Designer or QtCreator with QUiLoader and pyside6-uic

**https://doc.qt.io/qtforpython/tutorials/basictutorial/uifiles.html**

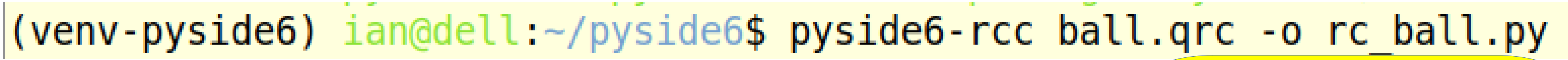
**https://www.pythonguis.com/tutorials/packaging-data-files-pyside6-with-qresource-system/**

# PySide6 – Resource Compiler

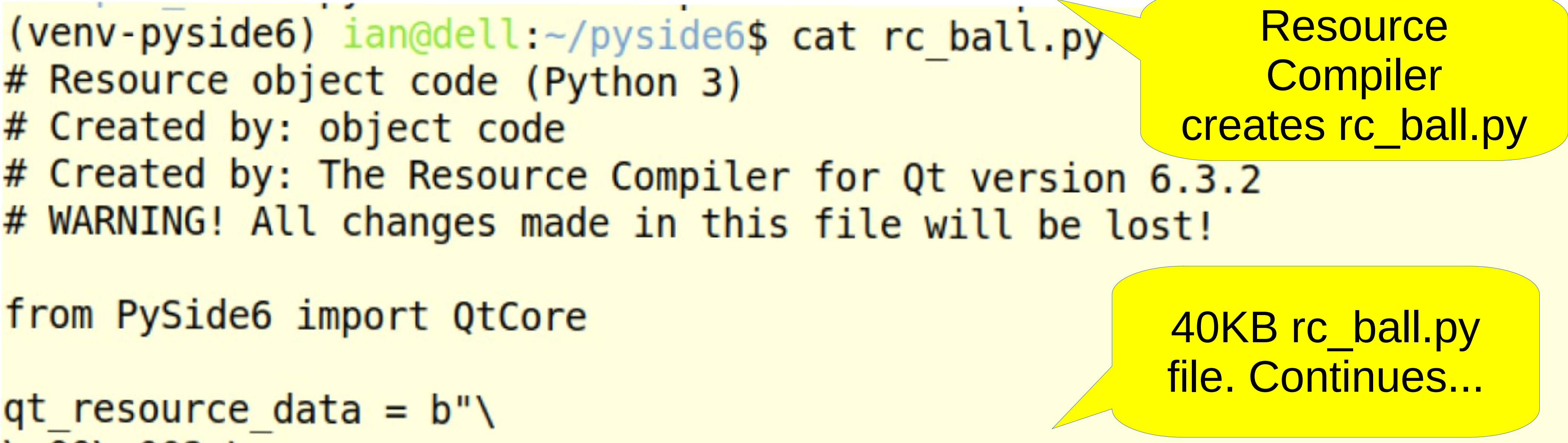


```
<!DOCTYPE RCC>
<RCC version="1.0">
<qresource>
    <file>ball.png</file>
</qresource>
</RCC>
```

ball.qrc file



```
(venv-pyside6) ian@dell:~/pyside6$ pyside6-rcc ball.qrc -o rc_ball.py
```

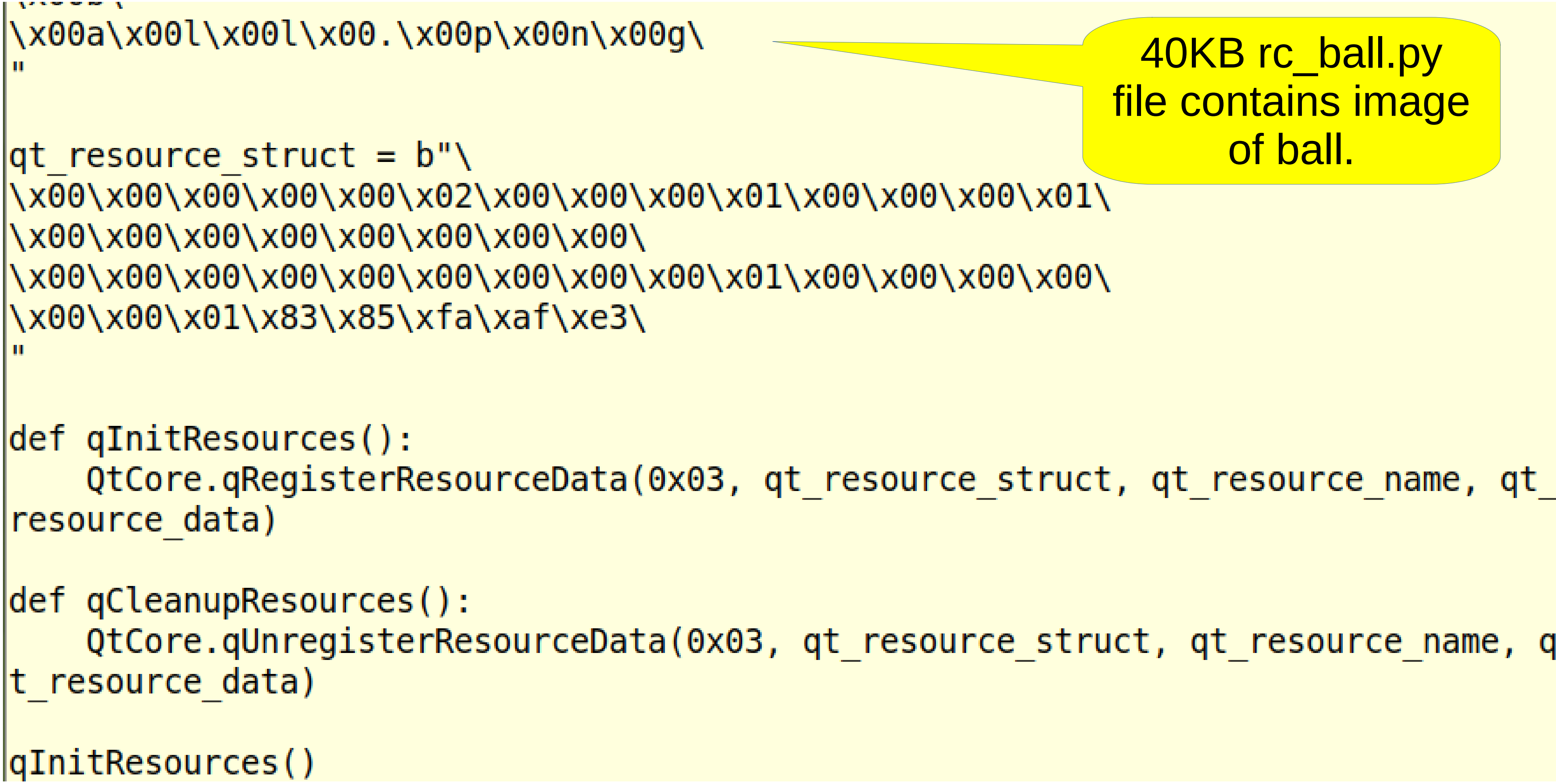


```
(venv-pyside6) ian@dell:~/pyside6$ cat rc_ball.py
# Resource object code (Python 3)
# Created by: object code
# Created by: The Resource Compiler for Qt version 6.3.2
# WARNING! All changes made in this file will be lost!

from PySide6 import QtCore

qt_resource_data = b"\
```

Resource Compiler creates rc_ball.py

40KB rc_ball.py file. Continues...

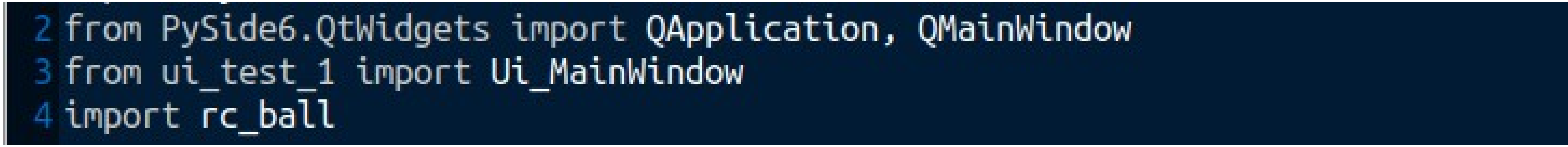
# PySide6 – RRC – rc_ball.py file continued...



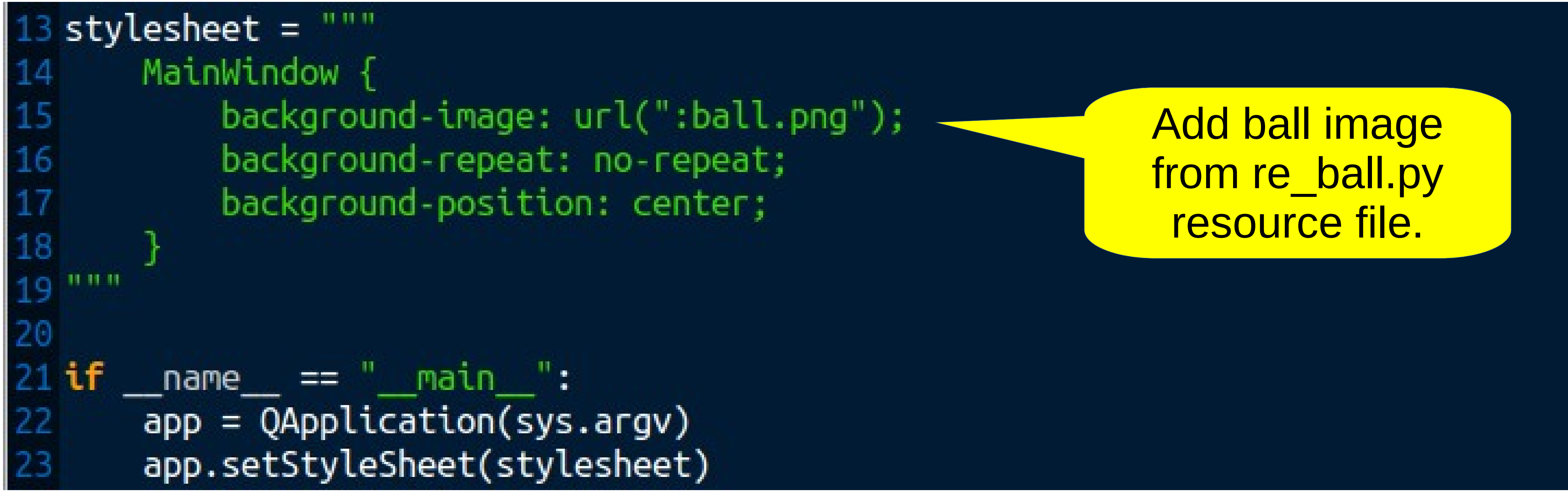
```
\x00a\x00l\x00l\x00.\x00p\x00n\x00g\
"

qt_resource_struct = b"\
\x00\x00\x00\x00\x00\x02\x00\x00\x00\x01\x00\x00\x00\x01\
\x00\x00\x00\x00\x00\x00\x00\x00\
\x00\x00\x00\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\
\x00\x00\x01\x83\x85\xfa\xaf\xe3\
"


def qInitResources():
    QtCore.qRegisterResourceData(0x03, qt_resource_struct, qt_resource_name, qt_
resource_data)

def qCleanupResources():
    QtCore.qUnregisterResourceData(0x03, qt_resource_struct, qt_resource_name, q
t_resource_data)

qInitResources()
```

40KB rc_ball.py file contains image of ball.

# PySide6 – test_1.py using rc_ball.py file.



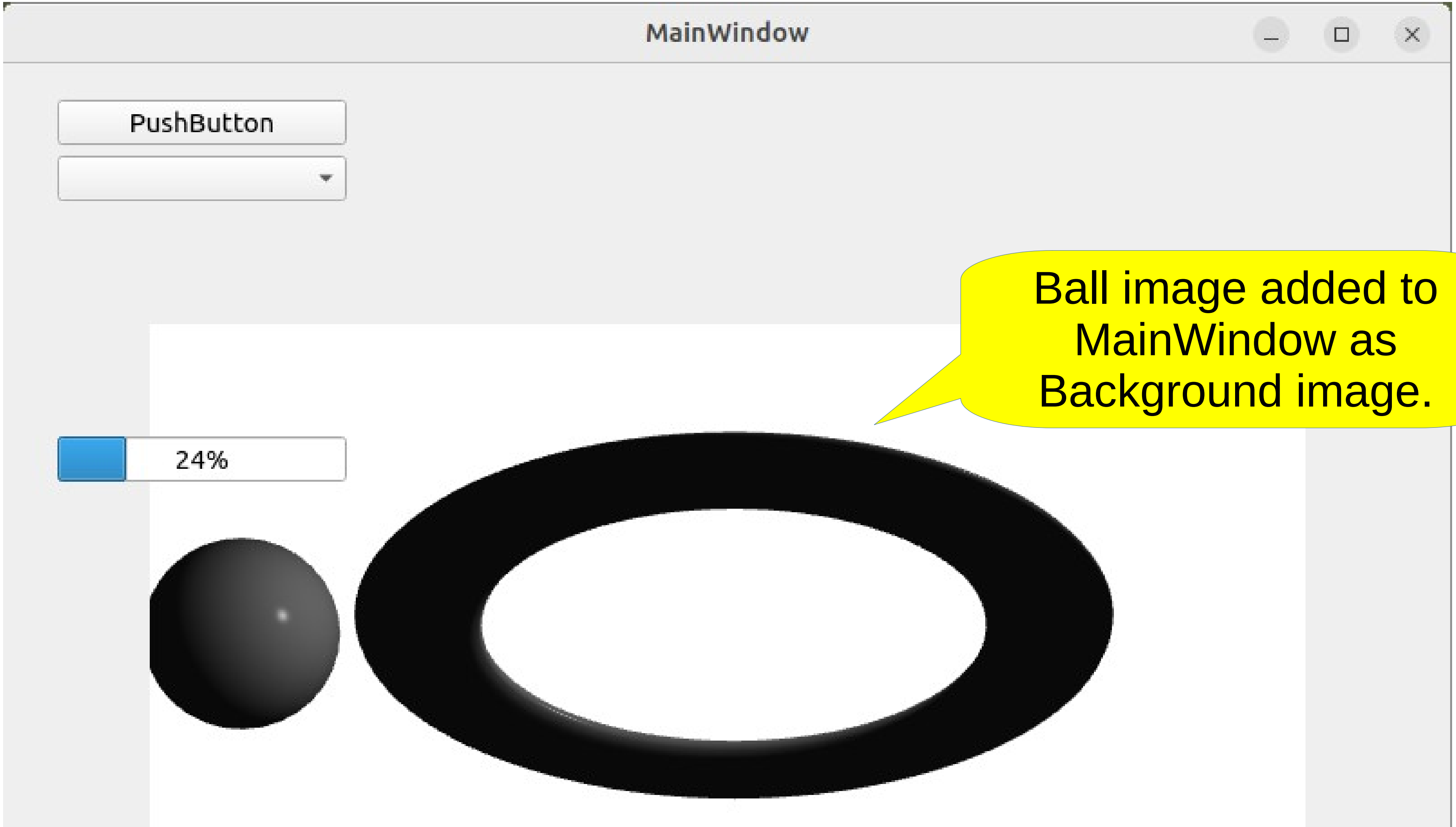
```
from PySide6.QtWidgets import QApplication, QMainWindow
from ui_test_1 import Ui_MainWindow
import rc_ball
```



```
stylesheet = """
    MainWindow {
        background-image: url(":ball.png");
        background-repeat: no-repeat;
        background-position: center;
    }
"""

if __name__ == "__main__":
    app = QApplication(sys.argv)
    app.setStyleSheet(stylesheet)
```

Add ball image from re_ball.py resource file.

# PySide6 – Image from Resource Compiler file.



Ball image added to MainWindow as Background image.