

Руководство по установке PySide на 32-битную Windows в условиях локальной сети ЛКШ.

Можно ставить как на местные компьютеры, так и на личные.

1. Скачать со страницы <https://ejudge.lksh.ru/soft/> Wing IDE 101 5.0.7 (если у вас версия ниже, обязательно нужно скачать именно эту!)
2. Оттуда же скачать Python 3.4.1 (ставить в папку по умолчанию, которая называется Python34)
3. Установить скачанное в пунктах 1, 2 (в любом порядке)
4. Скачать с той же страницы PySide 1.2.2, не меняя названия файла (\*.whl)
5. Набрать в терминале команду  
***%путь к папке Python34%/Scripts/pip.exe install %путь к файлу из 4 пункта%***

В запущенном Wing IDE 5.0.7 изменить версию интерпретатора (найти в меню пункт Configure Python и изменить, например, в пути к интерпретатору имя папки Python23 на Python34).

Ниже приведены примеры кода (строки со звездочками копировать не нужно, это названия примеров). Какие-то из них разбирались на спецкурсе.

Тutorial, откуда были взяты примеры: <http://zetcode.com/gui/pysidetutorial/> (в ЛКШ ссылка не будет открываться, это нормально!)

```
*****Hello World*****  
import sys  
from PySide import QtGui  
  
app = QtGui.QApplication(sys.argv)  
wid = QtGui.QWidget()  
  
wid.resize(250, 150)  
wid.setWindowTitle("Hello, world!")  
wid.show()  
  
sys.exit(app.exec_())
```

```
*****Box Layout.*****
```

```
import sys
from PySide import QtGui

class Example(QtGui.QWidget):
    def __init__(self):
        super(Example, self).__init__()
        self.initUI()

    def initUI(self):
        okButton = QtGui.QPushButton("OK")
        cancelButton = QtGui.QPushButton("Cancel")
        hbox = QtGui.QHBoxLayout()
        hbox.addStretch(1)
        hbox.addWidget(okButton)
        hbox.addWidget(cancelButton)
        vbox = QtGui.QVBoxLayout()
        vbox.addStretch(1)
        vbox.addLayout(hbox)
        self.setLayout(vbox)
        self.setGeometry(300, 300, 300, 150)
        self.setWindowTitle('Buttons')
        self.show()

def main():
    app = QtGui.QApplication(sys.argv)
    ex = Example()
    sys.exit(app.exec_())
```

```
*****Emit Signal*****
import sys
from PySide import QtGui, QtCore

class Communicate(QtCore.QObject):
    closeApp = QtCore.Signal()

class Example(QtGui.QMainWindow):
    def __init__(self):
        super(Example, self).__init__()
        self.initUI()

    def initUI(self):
        self.c = Communicate()
        self.c.closeApp.connect(self.close)
        self.setGeometry(300, 300, 290, 150)
        self.setWindowTitle('Emit signal')
        self.show()

    def mousePressEvent(self, event):
        self.c.closeApp.emit()

def main():
    app = QtGui.QApplication(sys.argv)
    ex = Example()
    sys.exit(app.exec_())
```

\*\*\*\*\*Events\*\*\*\*\*

```
import sys
from PySide import QtGui, QtCore

class Example(QtGui.QWidget):
    def __init__(self):
        super(Example, self).__init__()
        self.initUI()

    def initUI(self):
        self.setGeometry(300, 300, 250, 150)
        self.setWindowTitle('Event handler')
        self.show()

    def keyPressEvent(self, e):
        print(e.type())
        if e.key() == QtCore.Qt.Key_Escape:
            self.close()

def main():
    app = QtGui.QApplication(sys.argv)
    ex = Example()
    sys.exit(app.exec_())
```

```
*****Grid Layout*****
```

```
import sys
from PySide import QtGui

class Example(QtGui.QWidget):
    def __init__(self):
        super(Example, self).__init__()
        self.initUI()

    def initUI(self):
        names = ['Cls', 'Bck', " ", 'Close', '7', '8', '9', '/',
                '4', '5', '6', '*', '1', '2', '3', " ",
                '0', '.', '=', '+']
        grid = QtGui.QGridLayout()
        j = 0
        pos = [(0, 0), (0, 1), (0, 2), (0, 3),
              (1, 0), (1, 1), (1, 2), (1, 3),
              (2, 0), (2, 1), (2, 2), (2, 3),
              (3, 0), (3, 1), (3, 2), (3, 3),
              (4, 0), (4, 1), (4, 2), (4, 3)]
        for i in names:
            button = QtGui.QPushButton(i)
            if j == 2:
                grid.addWidget(QtGui.QLabel('Label'), 0, 2)
            else:
                grid.addWidget(button, pos[j][0], pos[j][1])
            j = j + 1
        self.setLayout(grid)
        self.move(300, 150)
        self.setWindowTitle('Calculator')
        self.show()

def main():
    app = QtGui.QApplication(sys.argv)
    ex = Example()
    sys.exit(app.exec_())
```

\*\*\*\*\*Absolute Positioning Layout\*\*\*\*\*

```
import sys
from PySide import QtGui

class Example(QtGui.QWidget):
    def __init__(self):
        super(Example, self).__init__()
        self.initUI()

    def initUI(self):
        label1 = QtGui.QLabel('<font size=15>Summer</font>', self)
        label1.move(15, 10)
        label2 = QtGui.QLabel('<font size=15>Informatics</font>', self)
        label2.move(35, 40)
        label3 = QtGui.QLabel('<font size=15>School</font>', self)
        label3.move(55, 70)
        self.setGeometry(300, 300, 250, 150)
        self.setWindowTitle('Absolute')
        self.show()

def main():
    app = QtGui.QApplication(sys.argv)
    ex = Example()
    sys.exit(app.exec_())
```

```
*****Signal and slot*****
```

```
import sys
from PySide import QtGui, QtCore
```

```
def f(x):
    print(x)
```

```
class Example(QtGui.QWidget):
    def __init__(self):
        super(Example, self).__init__()
        self.initUI()
```

```
    def initUI(self):
        lcd = QtGui.QLCDNumber(self)
        sld = QtGui.QSlider(QtCore.Qt.Horizontal, self)
        vbox = QtGui.QVBoxLayout()
        vbox.addWidget(lcd)
        vbox.addWidget(sld)
        self.setLayout(vbox)
        sld.valueChanged.connect(lcd.display)
        sld.valueChanged.connect(f)
        self.setGeometry(300, 300, 250, 150)
        self.setWindowTitle('Signal & slot')
        self.show()
```

```
def main():
    app = QtGui.QApplication(sys.argv)
    ex = Example()
    sys.exit(app.exec_())
```

```
*****Tooltips*****
```

```
import sys
```

```
from PySide import QtGui
```

```
class Example(QtGui.QWidget):
```

```
    def __init__(self):
```

```
        super(Example, self).__init__()
```

```
        self.initUI()
```

```
    def initUI(self):
```

```
        QtGui.QToolTip.setFont(QtGui.QFont('SansSerif', 10))
```

```
        self.setToolTip('This is a <b>QWidget</b> widget')
```

```
        btn = QtGui.QPushButton('Button', self)
```

```
        btn.setToolTip('This is a <b>QPushButton</b> widget')
```

```
        btn.resize(btn.sizeHint())
```

```
        btn.move(50, 50)
```

```
        self.setGeometry(300, 300, 250, 150)
```

```
        self.setWindowTitle('Tooltips')
```

\*\*\*\*\*Menus and Toolbars\*\*\*\*\*

```
import sys
from PySide import QtGui

class Example(QtGui.QMainWindow):

    def __init__(self):
        super(Example, self).__init__()

        self.initUI()

    def initUI(self):

        textEdit = QtGui.QTextEdit()
        self.setCentralWidget(textEdit)

        exitAction = QtGui.QAction('Exit', self)
        exitAction.setShortcut('Ctrl+Q')
        exitAction.setStatusTip('Exit application')
        exitAction.triggered.connect(self.close)

        self.statusBar()

        menubar = self.menuBar()
        fileMenu = menubar.addMenu('&File')
        fileMenu.addAction(exitAction)

        toolbar = self.addToolBar('Exit')
        toolbar.addAction(exitAction)

        self.setGeometry(300, 300, 350, 250)
        self.setWindowTitle('Main window')
        self.show()

def main():

    app = QtGui.QApplication(sys.argv)
    ex = Example()
    sys.exit(app.exec_())
```