

```

from PyQt5.QtWidgets import \
    QApplication, QWidget, QLabel, \
    QPushButton, QVBoxLayout, QComboBox, QHBoxLayout
from PyQt5.QtNetwork import QTcpSocket

import sys

class Crossing(QWidget):
    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)

        # Výroba ovládacích prvků
        self.connectButton = QPushButton(self, text="Start")
        self.connectButton.clicked.connect(self.connect)

        self.disconnectButton = QPushButton(self, text="Stop")
        self.disconnectButton.clicked.connect(self.disconnect)

        self.backButton = QPushButton(self, text="Return",
enabled=False)
        self.backButton.clicked.connect(self.sendBack)

        self.presentObjects = QComboBox(self)

        self.backAllButton = QPushButton(self, text="Return All")
        self.backAllButton.clicked.connect(self.sendBackAll)

        self.messageLabel = QLabel(self)
        self.statusLabel = QLabel(self)
        self.statusLabel.setText('Disconnected')
        self.statLabel = QLabel(self)

        # Rozložení ovládacích prvků
        self.layout = QVBoxLayout(self)

        self.startLayout = QHBoxLayout()
        self.startLayout.addWidget(self.connectButton)
        self.startLayout.addWidget(self.disconnectButton)
        self.layout.addLayout(self.startLayout)

        self.backLayout = QHBoxLayout()

```

```
self.backLayout.addWidget(self.presentObjects)
self.backLayout.addWidget(self.backButton)
self.backLayout.addWidget(self.backAllButton)
self.layout.addLayout(self.backLayout)

self.layout.addWidget(self.messageLabel)
self.layout.addWidget(self.statusLabel)
self.layout.addWidget(self.statLabel)

# Příprava TCP socketu
self.socket = QTcpSocket(self)
self.socket.readyRead.connect(self.read)
self.socket.connected.connect(self.connected)
self.readBuffer = bytearray()
self.disconnecting = False

# Zobrazení
self.setLayout(self.layout)
self.show()

def disconnect(self):
    self.socket.write("BYE\n".encode())
    self.disconnecting = True
    self.read()

def softDisconnect(self):
    # Odpojí se od serveru
    self.socket.disconnectFromHost()
    self.statusLabel.setText('Disconnected')

def connect(self):
    # Nejdřív se odpoj, pokud už
    # spojení běží
    self.socket.abort()

    # A znovu se připoj
    self.socket.connectToHost("ksp.mff.cuni.cz", 48888)
    self.statusLabel.setText("Connecting")

def connected(self):
    # Pozdravíme server
    self.socket.write("HELLO\n".encode())
```

```
self.statusLabel.setText("Connected")
self.disconnecting = False

def read(self):
    # Přečteme všechno, co jsme dostali
    while self.socket.bytesAvailable() > 0:
        self.readBuffer += self.socket.read(128)

    # Rozdělíme na řádky
    lines = self.readBuffer.split(b"\n")

    # Zbytek uložíme na příště
    self.readBuffer = lines.pop()

    # Zpracujeme řádky, které dorazily
    for l in lines:
        text = l.decode().rstrip()
        if "STAT" not in text:
            self.backButton.setEnabled(True)
            self.presentObjects.addItem(text)
            self.messageLabel.setText(text)
        else:
            self.statLabel.setText(text)
            if self.disconnecting:
                self.softDisconnect()

def sendBack(self):
    text = self.presentObjects.currentText() + "\n"
    if not text == "":
        index = self.presentObjects.currentIndex()
        self.socket.write(text.encode())
        self.presentObjects.removeItem(index)

        if self.presentObjects.currentText() == "":
            self.backButton.setEnabled(False)

def sendBackAll(self):
    allItems = [self.presentObjects.itemText(i) for i
                in range(self.presentObjects.count())]
    for item in allItems:
        text = item + "\n"
        self.socket.write(text.encode())
```

```
self.presentObjects.clear()
```

```
# Spuštění celého programu  
app = QApplication(sys.argv)  
crossing = Crossing()  
app.exec()
```