Objectives

• You will learn about:
  – Graphical user interface (GUI) programming
  – Specifically...
  – For Java: The AWT/Swing GUI library
  – For Python: The PyQt5 GUI library
Agenda

• Event handling
AWT Event Handling Examples

• Show `EventTest1.java`
  – `JButton` event handling
  – Event handling for other components is similar
  – Listener objects must access panel
AWT Event Handling Problem

• **Problem:**
  – Listener object often must access Components other than the one generating the event
  – Awkward to pass Components to Listener constructor

• **Solution:** …
AWT Event Handling Examples

• See **EventTest2.java**
  - *ColorActionListener2* is an *inner class*
    • Defined within *EventTestRunnable2*
    • *ColorActionListener2* object can access fields of *EventTestRunnable2* object that instantiated it
Inner Class Commentary

• Inner class commentary:
  – Q: Use inner classes?
  – A: I vote yes
    • Inner classes yield more succinct code
    • Additional language complexity is minimal
AWT Event Handling Examples

• See `EventTest3.java`
  - `event.getSource()`
    • Listener object can determine source of event
    • Can instantiate one listener object shared by all three `JButton` objects
# AWT Event Handling

Some **semantic events**:  

<table>
<thead>
<tr>
<th>Component</th>
<th>Listener Interface</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Button</td>
<td>ActionListener</td>
<td>ActionEvent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getSource()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getActionCommand()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getModifiers()</td>
</tr>
<tr>
<td>TextField</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MenuItem</td>
<td>ListSelectionListener</td>
<td>ListSelectionEvent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getSource()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getFirstIndex()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getLastIndex()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getValueIsAdjusting()</td>
</tr>
<tr>
<td>Menu</td>
<td>ChangeListener</td>
<td>ChangeEvent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getSource()</td>
</tr>
<tr>
<td>JList</td>
<td>ListSelectionListener</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>valueChanged()</td>
</tr>
<tr>
<td>JSlider</td>
<td>ChangeListener</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>stateChanged()</td>
</tr>
<tr>
<td>JCheckBox</td>
<td>ItemListener</td>
<td>ItemEvent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getSource()</td>
</tr>
<tr>
<td>JRadioButton</td>
<td></td>
<td>itemStateChanged()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getStateChange()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getItemSelectable()</td>
</tr>
</tbody>
</table>
## AWT Event Handling

Some *low-level events*:

<table>
<thead>
<tr>
<th>Component</th>
<th>Listener Interface</th>
<th>Parameter</th>
<th>Some of its Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component</td>
<td>FocusListener</td>
<td>FocusEvent</td>
<td>getSource(), isTemporary()</td>
</tr>
<tr>
<td></td>
<td>focusGained()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>focusLost()</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component</td>
<td>KeyListener</td>
<td>KeyEvent</td>
<td>getSource(), getKeyChar(), getKeyCode(), isAltDown(), isControlDown(), isMetaDown(), isShiftDown(), getKeyModifiersText(), getKeyText(), isActionKey()</td>
</tr>
<tr>
<td></td>
<td>keyPressed()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>keyReleased()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>keyTyped()</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component</td>
<td>MouseListener</td>
<td>MouseEvent</td>
<td>getSource(), getModifiers(), isAltDown(), isControlDown(), isMetaDown(), isShiftDown(), getClickCount(), getX(), getY(), getPoint(), translatePoint()</td>
</tr>
<tr>
<td></td>
<td>mousePressed()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>mouseReleased()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>mouseEntered()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>mouseExited()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>mouseClicked()</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component</td>
<td>MouseMotionListener</td>
<td>MouseEvent</td>
<td>getSource()</td>
</tr>
<tr>
<td></td>
<td>mouseDragged()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>mouseMoved()</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# AWT Event Handling

Some *low-level events*:

<table>
<thead>
<tr>
<th>Component</th>
<th>Listener Interface</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Its Methods</td>
<td>Some of its Methods</td>
</tr>
<tr>
<td>Component</td>
<td>MouseWheelListener</td>
<td>MouseWheelEvent</td>
</tr>
<tr>
<td></td>
<td>mouseWheelMoved()</td>
<td>getSource(),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getWheelRotation(),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getScrollAmount()</td>
</tr>
<tr>
<td>Window</td>
<td>WindowListener</td>
<td>WindowEvent</td>
</tr>
<tr>
<td></td>
<td>windowClosing()</td>
<td>getSource(),</td>
</tr>
<tr>
<td></td>
<td>windowOpening()</td>
<td>getComponent(),</td>
</tr>
<tr>
<td></td>
<td>windowOpened()</td>
<td>getWindow()</td>
</tr>
<tr>
<td></td>
<td>windowIconified()</td>
<td></td>
</tr>
<tr>
<td></td>
<td>windowDeiconified()</td>
<td></td>
</tr>
<tr>
<td></td>
<td>windowClosed()</td>
<td></td>
</tr>
<tr>
<td></td>
<td>windowActivated()</td>
<td></td>
</tr>
<tr>
<td></td>
<td>windowDeactivated()</td>
<td></td>
</tr>
<tr>
<td>Window</td>
<td>WindowFocusListener</td>
<td>WindowEvent</td>
</tr>
<tr>
<td></td>
<td>windowGainedFocus()</td>
<td>getSource(),</td>
</tr>
<tr>
<td></td>
<td>windowLostFocus()</td>
<td>getComponent(),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getWindow()</td>
</tr>
<tr>
<td>Window</td>
<td>WindowStateListener</td>
<td>WindowEvent</td>
</tr>
<tr>
<td></td>
<td>windowStateChanged()</td>
<td>getSource(),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getOldState(),</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getNewState()</td>
</tr>
</tbody>
</table>
PyQt5 Event Handling

• PyQt5 event handling
  – **Mechanism 1: signals & slots**
    • Similar to Java Swing *semantic* event handling
  – **Mechanism 2: overriding event functions**
    • Similar to Java Swing low-level event handling
PyQt5 Event Handling Examples

• See `eventtest1.py`
  – QPushButton event handling
  – Event handling for other widgets is similar
  – global window
    • Defines `window` variable to be global
    • Functions other than `main()` can access `window` variable
PyQt5 Event Handling Examples

- **See eventtest1.py** (cont.)
  - `redButton.clicked.connect(redButtonSlot)`
- **Function callback mechanism**
  - `clicked` is a **signal**
  - `redButton` is the **receiver** of the signal
  - `redButtonSlot` is a **slot**
- **redButton clicked => call redButtonSlot**
PyQt5 Event Handling Problem

• **Problem:**
  – Slots often must access widgets
  – Poor style to use global variables

• **Solution:** …
PyQt5 Event Handling Examples

• See `eventtest2.py`
  – *Inner function definitions*
  – In spirit, similar to Java’s inner class definition
  – Inner function can access variables that are local to containing function/method
PyQt5 Event Handling Examples

• See **eventtest3.py**
  – `window.sender()`
    • Slot can determine source of event
    • Can define one slot shared by all three `QPushButton` objects

• See **eventtest4.py**
  – A nicer approach
  – Use **lambda expressions**
  – Covered later in course
PyQt5 Event Handling

Class

\[ \text{signal} \Rightarrow \text{slot(type param, ...)} \]

(QObject)

destroyed => f(QObject obj)
objectNameChanged => f(str objectName)

(QWidget)

customContextMenuRequested => f(QPoint pos)
windowIconChanged => f(QIcon icon)
windowIconTextChanged => f(QIcon iconText)
windowTitleChanged => f(str title)

(QMainWindow)

iconSizeChanged => f(QSize iconSize)
tabifiedDockWidgetActivated => f(QDockWidget dockWidget)
toolButtonStyleChanged => f(QToolButtonStyle toolButtonStyle)
PyQt5 Event Handling

Class

\[ \text{signal} \Rightarrow \text{slot(type param, ...)} \]

QWidget (continued)

QFrame

QAbstractScrollArea

QAbstractItemView

QListView
PyQt5 Event Handling

Class

\[\text{signal} \Rightarrow \text{slot(type param, ...)}\]

QListView (continued)

QListWidget

\[
\begin{align*}
\text{currentItemChanged} & \Rightarrow f(\text{QListWidgetItem cur, QListWidgetItem previous}) \\
\text{currentRowChanged} & \Rightarrow f(\text{int currentRow}) \\
\text{currentTextChanged} & \Rightarrow f(\text{str currentText}) \\
\text{itemActivated} & \Rightarrow f(\text{QListWidgetItem item}) \\
\text{itemChanged} & \Rightarrow f(\text{QListWidgetItem item}) \\
\text{itemClicked} & \Rightarrow f(\text{QListWidgetItem item}) \\
\text{itemDoubleClicked} & \Rightarrow f(\text{QListWidgetItem item}) \\
\text{itemEntered} & \Rightarrow f(\text{QListWidgetItem item}) \\
\text{itemPressed} & \Rightarrow f(\text{QListWidgetItem item}) \\
\text{itemSelectionChanged} & \Rightarrow f() \\
\end{align*}
\]
PyQt5 Event Handling

Class

\[ signal \Rightarrow slot(type\ param, \ldots) \]

QAbstractScrollArea (continued)

QTextEdit

\[ \begin{align*}
\text{copyAvailable} & \Rightarrow f(\text{bool yes}) \\
\text{currentCharFormatChanged} & \Rightarrow f(\text{QTextCharFormat fmt}) \\
\text{cursorPositionChanged} & \Rightarrow f() \\
\text{redoAvailable} & \Rightarrow f(\text{bool available}) \\
\text{selectionChanged} & \Rightarrow f() \\
\text{textChanged} & \Rightarrow f() \\
\text{undoAvailable} & \Rightarrow f(\text{bool available})
\end{align*} \]
PyQt5 Event Handling

Class

\[\text{signal} \Rightarrow \text{slot(type param, ...)}\]

QWidget (continued)
QAbstractButton

\[\text{clicked} \Rightarrow f(\text{bool checked})\]
pressed \(\Rightarrow f()\)
released \(\Rightarrow f()\)
toggled \(\Rightarrow f(\text{bool checked})\)

QPushButton

QCheckBox

\[\text{stateChanged} \Rightarrow f(\text{int state})\]

QRadioButton

QLabel

\[\text{linkActivated} \Rightarrow f(\text{str link})\]
\[\text{linkHovered} \Rightarrow f(\text{str link})\]
PyQt5 Event Handling

Class

\[ \text{signal} \rightarrow \text{slot(type param, ...)} \]

QWidget (continued)

QLineEdit

\[
\begin{align*}
\text{cursorPositionChanged} & \rightarrow f(\text{int} \ old\text{Pos}, \text{int} \ new\text{Pos}) \\
\text{editingFinished} & \rightarrow () \\
\text{inputRejected} & \rightarrow f() \\
\text{returnPressed} & \rightarrow f() \\
\text{selectionChanged} & \rightarrow f() \\
\text{textChanged} & \rightarrow f(\text{str} \ new\text{Text}) \\
\text{textEdited} & \rightarrow f(\text{str} \ new\text{Text})
\end{align*}
\]
PyQt5 Event Handling

Class

\[ signal \Rightarrow slot(type \ param, \ldots) \]

QWidget (continued)

QAbstractSlider

- actionTriggered \Rightarrow f(int \ action)
- rangeChanged \Rightarrow f(int \ min, \ int \ max)
- sliderMoved \Rightarrow f(int \ value)
- sliderPressed \Rightarrow f()
- sliderReleased \Rightarrow f()
- valueChanged \Rightarrow f(int \ value)

QSlider
PyQt5 Event Handling

Class

signal => slot(type param, …)

QWidget (continued)

QMenuBar

hovered => f(QAction action)
triggered => f(QAction action)

QMenu

aboutToHide => f()
aboutToShow => f()
hovered => f(QAction action)
triggered => f(QAction action)
PyQt5 Event Handling

• PyQt5 event handling
  – Mechanism 1: signals & slots
    • Similar to Java Swing semantic event handling
  – Mechanism 2: overriding event functions
    • Similar to Java Swing low-level event handling
PyQt5 Event Handling Examples

• **See** `eventtest5.py`
  – Lower-level event handling mechanism
  – *(con)* Less flexible
    • Compile-time (vs. run-time) binding of widgets to event handlers
  – *(pro)* More powerful
    • Allows handling of any event
    • Allows access to the event
PyQt5 Event Handling

Some PyQt5 classes and the event methods that they define:

**QObject**
- childEvent(event)
- customEvent(event)
- event(event)
- timerEvent(event)

**QWidget**
- actionEvent(event)
- changeEvent(event)
- closeEvent(event)
- contextMenuEvent(event)
- dragEnterEvent(event)
- dragLeaveEvent(event)
- dragMoveEvent(event)
- dropEvent(event)
- enterEvent(event)
- focusInEvent(event)
- focusOutEvent(event)
- hideEvent(event)
- inputMethodEvent(event)
- keyPressEvent(event)
- keyReleaseEvent(event)
- mouseMoveEvent(event)
- mousePressEvent(event)
- mouseReleaseEvent(event)
- moveEvent(event)
- nativeEvent(event)
- paintEvent(event)
- resizeEvent(event)
- showEvent(event)
- tabletEvent(event)
- wheelEvent(event)
Summary

• We have covered:
  – Graphical user interface (GUI) programming
  – Specifically...
  – For Java: The AWT/Swing GUI library
  – For Python: The PyQt5 GUI library
Summary

• For each:
  – Event handling