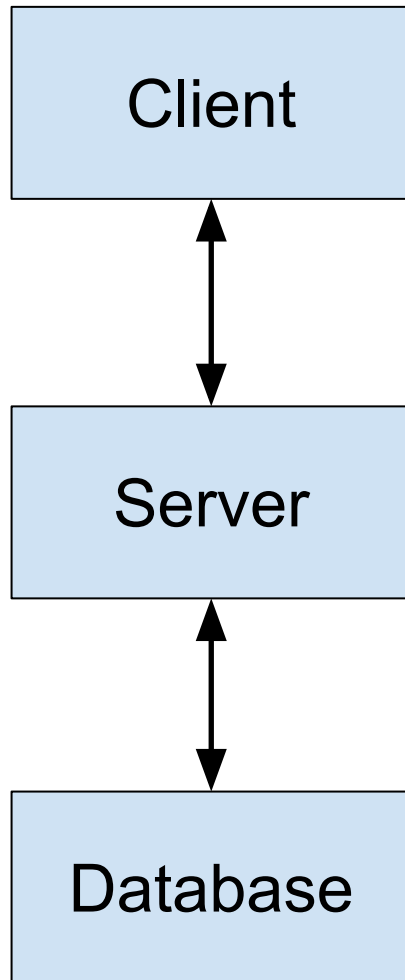


# Client-Side Options (Part 1)

Copyright © 2024 by  
Robert M. Dondero, Ph.D.  
Princeton University

# Objectives



Python  
Browser/HTML/JavaScript (jQuery, React)  
**Desktop apps (PyQt5)**

blue => course default  
red => other option

Python  
Python/Flask/Jinja2  
**Java/Servlets**  
**Java/Spring/Mustache**  
**JavaScript/Express/Mustache**

SQLite  
PostgreSQL

# Objectives

- We will cover:
  - Desktop app programming
  - A Penny **desktop** client

# Objectives

- More specifically...
- We will cover:
  - “High-level” desktop app programming
- We will not cover:
  - “Low-level” desktop app programming

# Motivation

- **Question:** Why study desktop app programming?
- **Answers:**
  - Mainstream “advanced programming”
  - Illustrates event-driven programming
  - Illustrates concurrent programming
  - Important!

# Agenda

- **Qt and PyQt5**
- PennyJson server
- Penny desktop client
  - Version 1: Baseline
  - Version 2: Sequential
  - Version 3: Bad
  - Version 4: Multi-threaded
  - Version 5: Stop
  - Version 6: Debouncing

# Qt and PyQt5

Cross-platform desktop app libraries:

Desktop App Library	Language	Platform
<b>GTK</b>	C, C++	Linux, Windows, macOS
<b>Qt</b>	<b>C, C++</b>	<b>Linux, Windows, macOS</b>
<b>wxWidgets</b>	C, C++	Linux, Windows, macOS
<b>Flutter</b>	C, C++, Dart	Linux, Windows, macOS, Android, iOS
<b>Kivy</b>	Python	Linux, Windows, macOS, Android, iOS
<b>Swing</b>	Java	Linux, Windows, macOS
<b>Tck/Tk</b>	Tcl	Linux, Windows, macOS, Android, iOS
...	...	...
<b>PyQt</b>	<b>Python</b>	<b>Linux, Windows, macOS</b>

# Qt and PyQt5

- **Qt**
  - Haavard Nord and Eirik Chambe-Eng
  - Trollteck, Nokia, The Qt Company





# Qt and PyQt5

- **Question:** Why study Qt?
  - Instead of some other GUI library?
- **Answers:**
  - Well designed, stable, robust
  - Excellent reputation
  - Very popular

# Qt and PyQt5

- ***PyQt5***
  - Phil Thompson
  - Riverbank Computing
  - A “Python binding” of Qt

# Qt and PyQt5

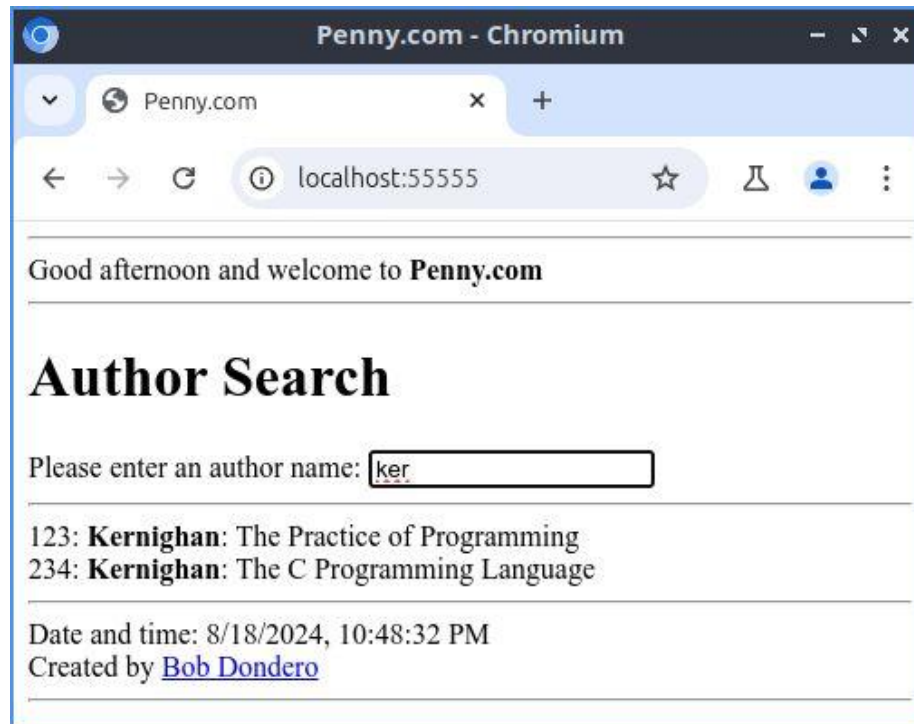
- **Question:** Why study PyQt5?
  - Instead of some other GUI library?
- **Answers:**
  - Convenient way to learn the fundamentals of Qt!

# Agenda

- Qt and PyQt5
- **PennyJson server**
- Penny desktop client
  - Version 1: Baseline
  - Version 2: Sequential
  - Version 3: Bad
  - Version 4: Multi-threaded
  - Version 5: Stop
  - Version 6: Debouncing

# PennyJson Server

- See **PennyJson** app
  - (Almost) same as **PennyAjax4** app from previous lecture



# PennyJson Server

- See **PennyJson** app (cont.)
  - runserver.py
  - penny.sql, penny.sqllite
  - database.py
  - index.html
    - Irrelevant for this lecture
  - **penny.py**

# PennyJson Server

- See **PennyJson** app (cont.)
  - Relevant observations
    - Added `IO_DELAY` environment variable
      - Simulates slow DB
      - Shows how client performs with I/O bound server

# Agenda

- Qt and PyQt5
- PennyJson server
- **Penny desktop client**
  - **Version 1: Baseline**
  - Version 2: Sequential
  - Version 3: Bad
  - Version 4: Multi-threaded
  - Version 5: Stop
  - Version 6: Debouncing



# Penny Desktop v1

- Penny desktop v1

```
$ export IO_DELAY=1  
$ python runserver.py 55555
```

```
$ python pennydesktop1base.py https://localhost:55555
```

# Penny Desktop v1

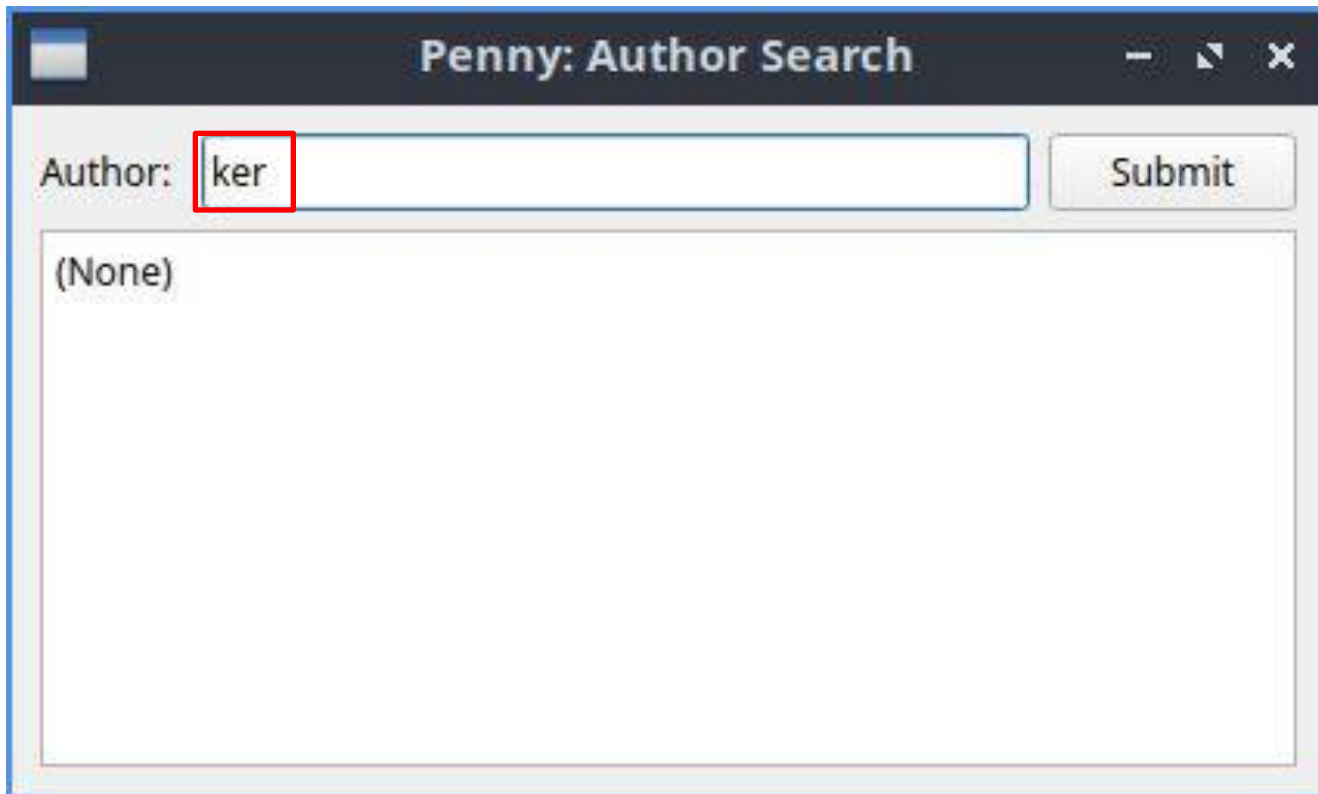
- Penny desktop v1



The image shows a desktop application window titled "Penny: Author Search". The window has a dark title bar with standard window controls (minimize, maximize, close) on the right. Below the title bar, there is a text input field labeled "Author:" and a "Submit" button. The main content area of the window is currently empty, displaying the text "(None)".

# Penny Desktop v1

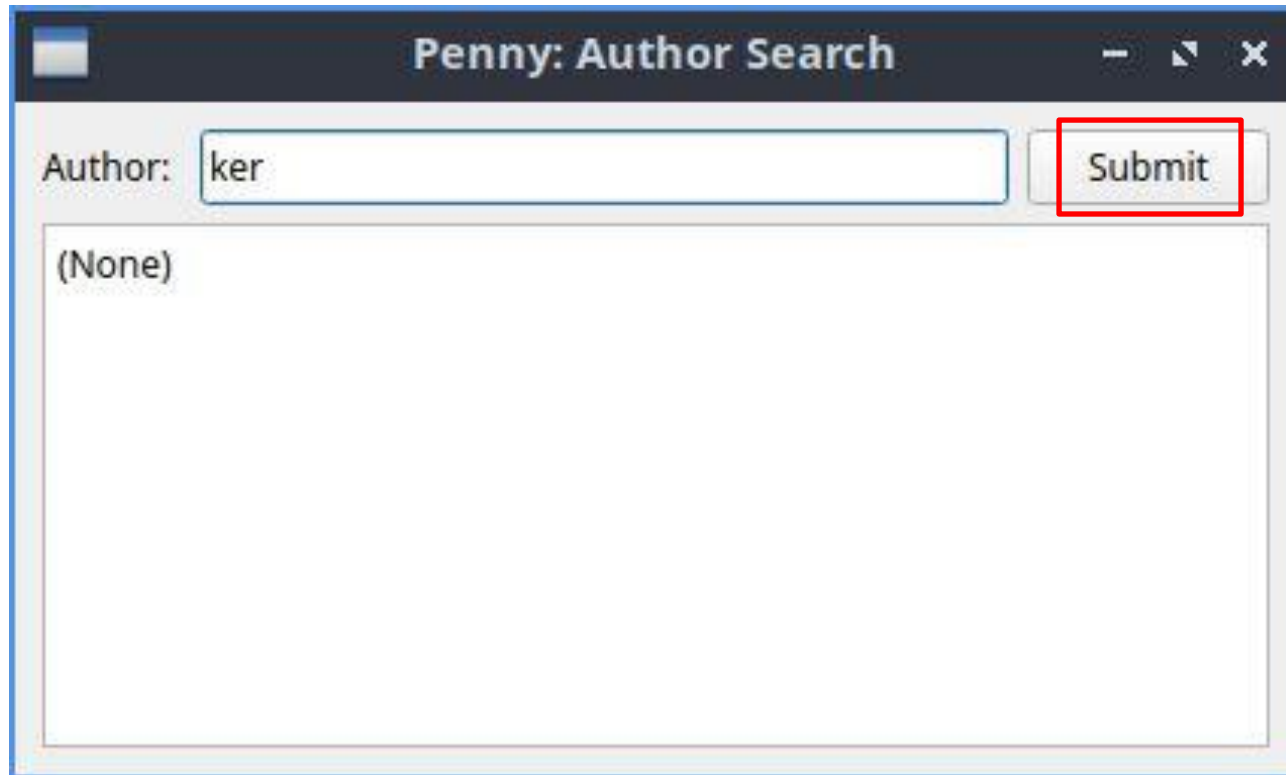
- Penny desktop v1



The image shows a desktop application window titled "Penny: Author Search". The window has a dark blue title bar with standard window controls (minimize, maximize, close) on the right. Below the title bar, there is a search form. On the left, the label "Author:" is followed by a text input field containing the text "ker". The input field is highlighted with a red rectangular border. To the right of the input field is a "Submit" button. Below the input field and button is a large, empty rectangular area, likely intended for search results, which currently displays the text "(None)".

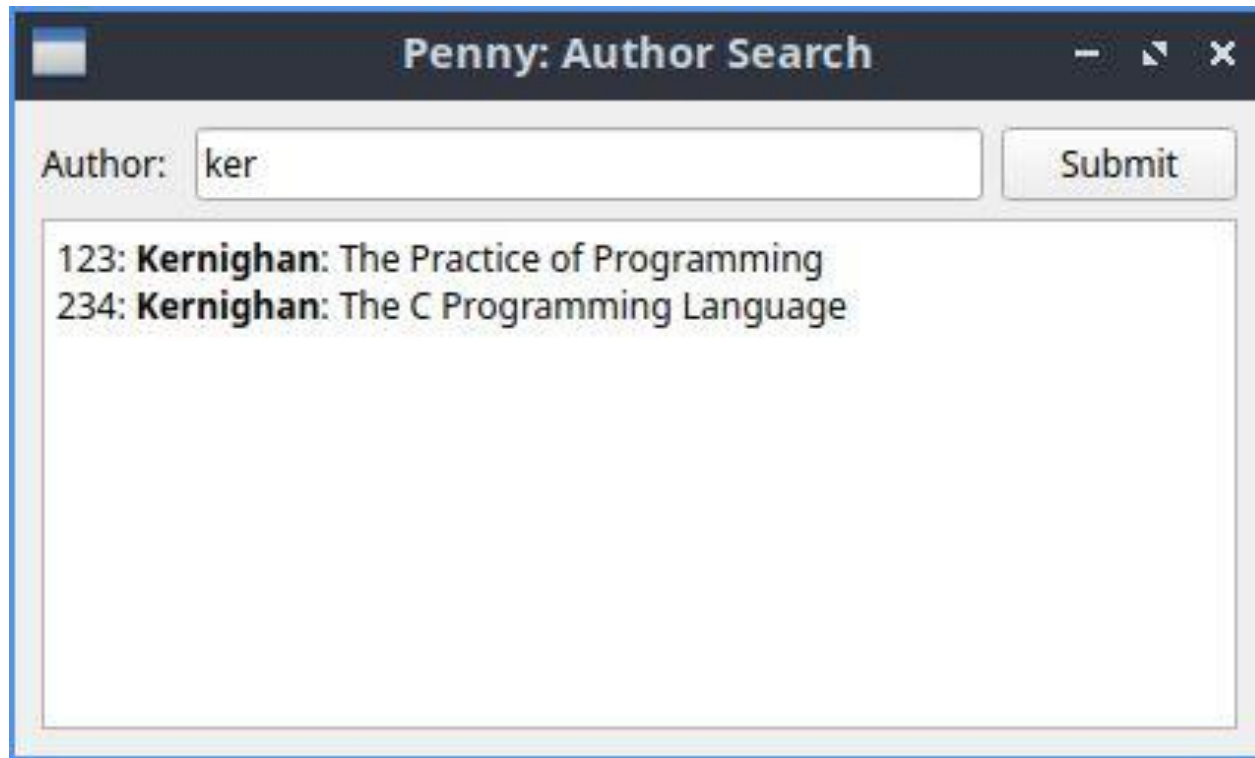
# Penny Desktop v1

- Penny desktop v1



# Penny Desktop v1

- Penny desktop v1



After a  
1 second  
delay

# Penny Desktop v1

- Penny desktop v1 (cont.)
  - See [pennydesktop1base.py](#)

# Penny Desktop v1

- Penny desktop v1
  - **(maybe) Problem:**
    - Inconsistent window state after typing author and before clicking Submit button
  - **Solution: redesign...**
    - Eliminate Submit button
    - GUI refreshes with each keystroke

# Agenda

- Qt and PyQt5
- PennyJSON server
- **Penny desktop client**
  - Version 1: Baseline
  - **Version 2: Sequential**
  - Version 3: Bad
  - Version 4: Multi-threaded
  - Version 5: Stop
  - Version 6: Debouncing



# Penny Desktop v2

- Penny desktop v2

```
$ export IO_DELAY=1  
$ python runserver.py 55555
```

```
$ python pennydesktop2seq.py https://localhost:55555
```

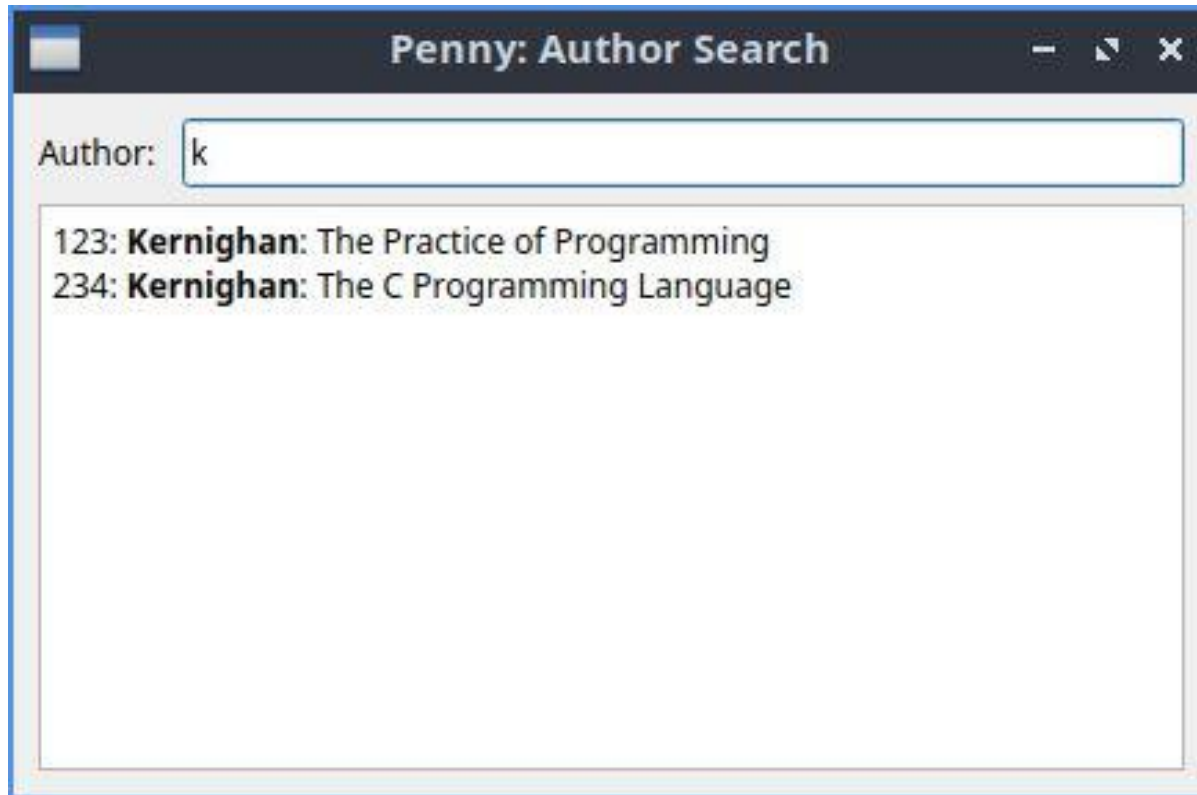
# Penny Desktop v2

- Penny desktop v2 (cont.)



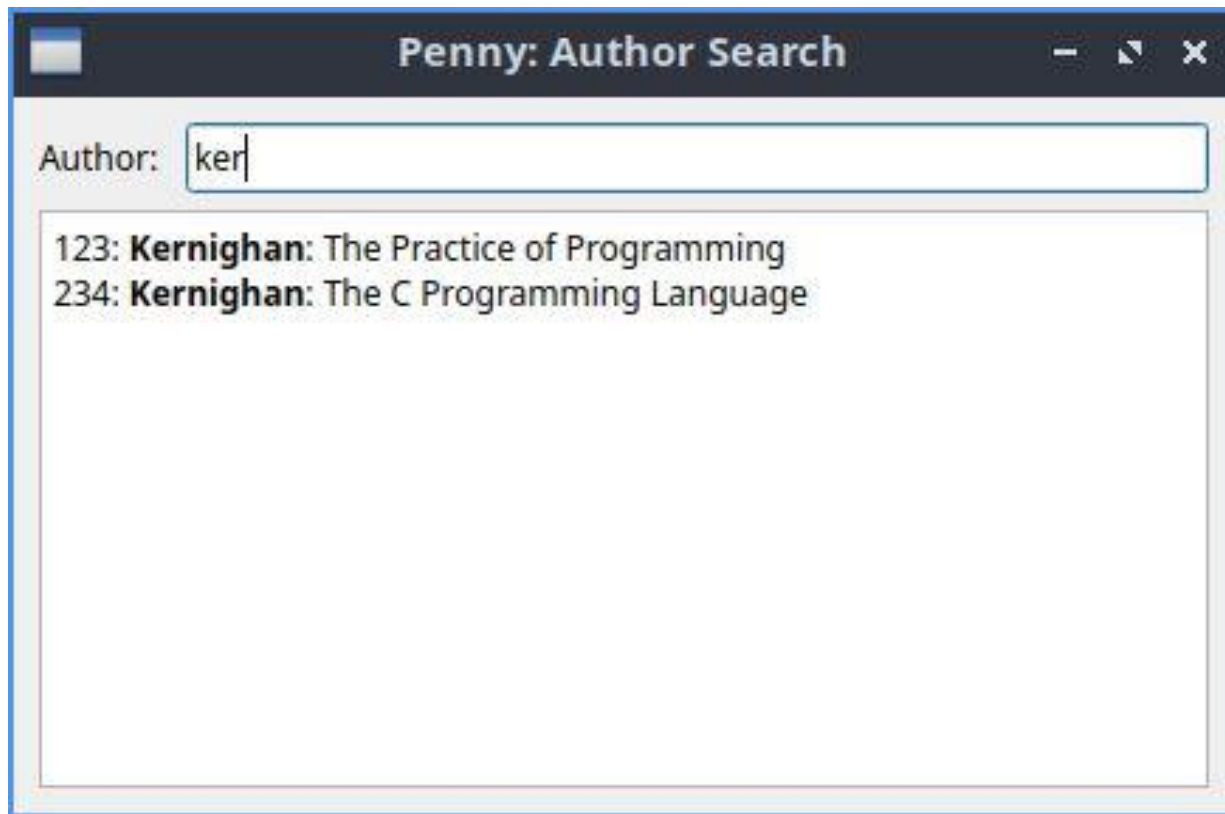
# Penny Desktop v2

- Penny desktop v2 (cont.)



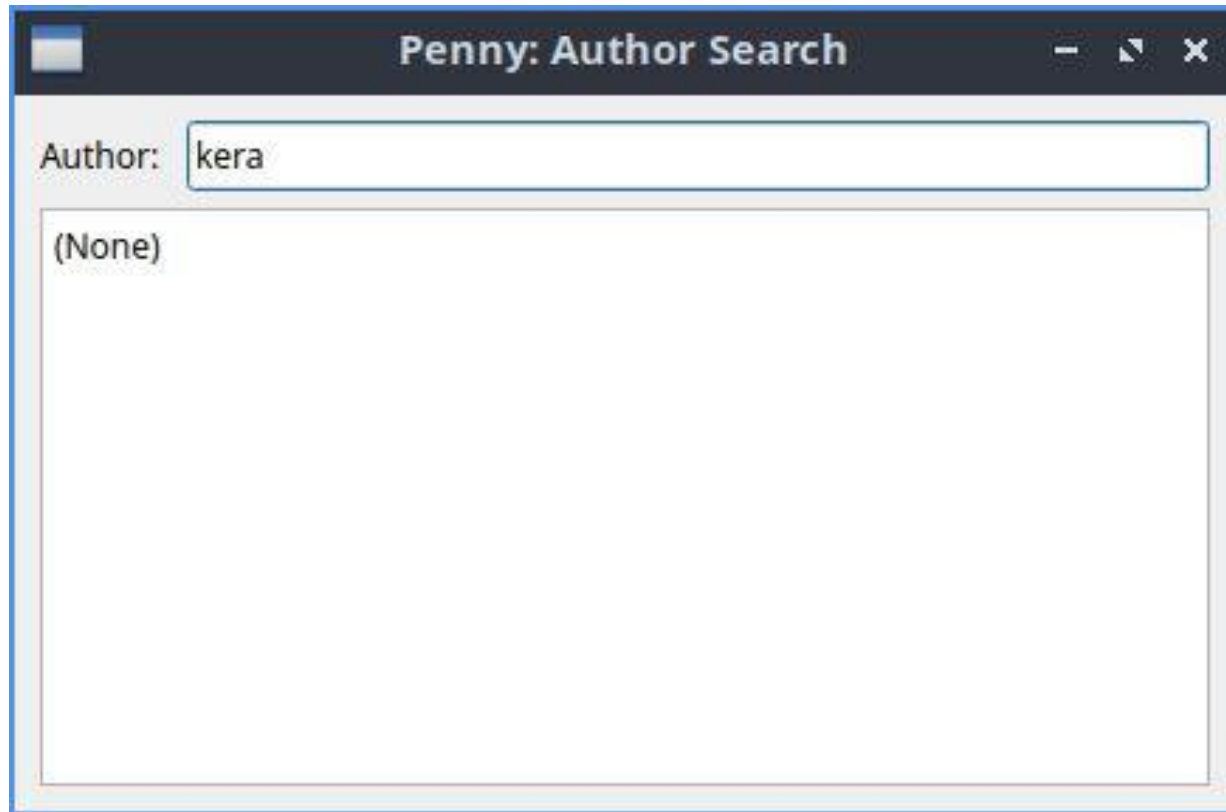
# Penny Desktop v2

- Penny desktop v2 (cont.)



# Penny Desktop v2

- Penny desktop v2 (cont.)



# Penny Desktop v2

- Penny desktop v2 (cont.)
  - See [pennyclient2seq.py](#)

# Penny Desktop v2

- Penny desktop v2 (cont.)
  - **Problem:**
    - Serious GUI lag
  - **Solution:** redesign...
    - Use threads!!!

# Agenda

- Qt and PyQt5
- PennyJson server
- **Penny desktop client**
  - Version 1: Baseline
  - Version 2: Sequential
  - **Version 3: Bad**
  - Version 4: Multi-threaded
  - Version 5: Stop
  - Version 6: Debouncing



# Penny Desktop v3

- Penny desktop v3
  - See **pennyclient3bad.py**

# Penny Desktop v3

- Penny desktop v3 (cont.)

```
$ export IO_DELAY=1  
$ python runserver.py 55555
```

```
$ python pennydesktop3bad.py http://localhost:55555  
QObject: Cannot create children for a parent that is  
in a different thread.  
(Parent is QTextDocument(0x25db0c0), parent's thread  
is QThread(0x2213e40), current thread is  
QThread(0x79c7cc000cd0)  
Segmentation fault (core dumped)  
$
```

# Penny Desktop v3

- **Problem:**
  - PyQt5 widgets are not thread safe
  - So PyQt5 prohibits worker thread from updating widgets
- **Solution:** redesign...
  - Worker thread communicates book list to main thread
  - Main thread updates GUI

# Agenda

- Qt and PyQt5
- PennyJson server
- **Penny desktop client**
  - Version 1: Baseline
  - Version 2: Sequential
  - Version 3: Bad
  - **Version 4: Multi-threaded**
  - Version 5: Stop
  - Version 6: Debouncing

# Penny Desktop v4

- Penny desktop v4

```
$ export IO_DELAY=1  
$ python runserver.py 55555
```

```
$ python pennydesktop4threads.py https://localhost:55555
```

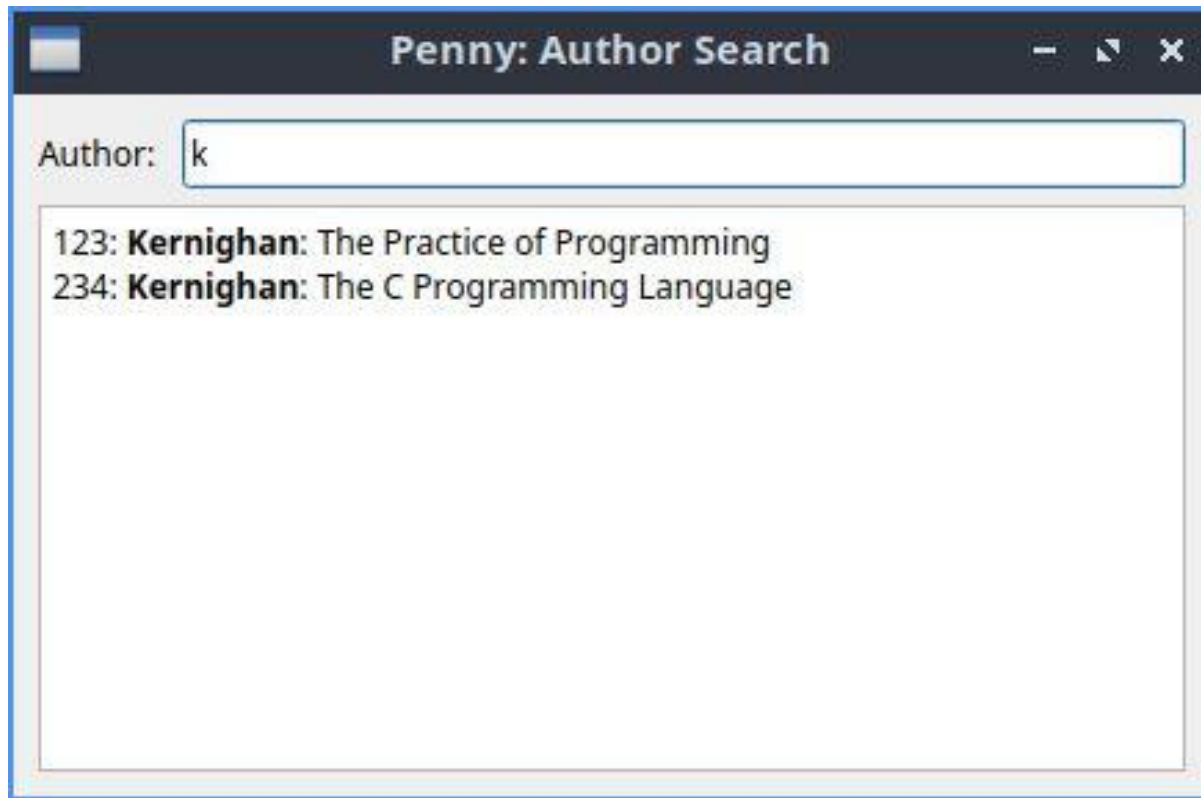
# Penny Desktop v4

- Penny desktop v4 (cont.)



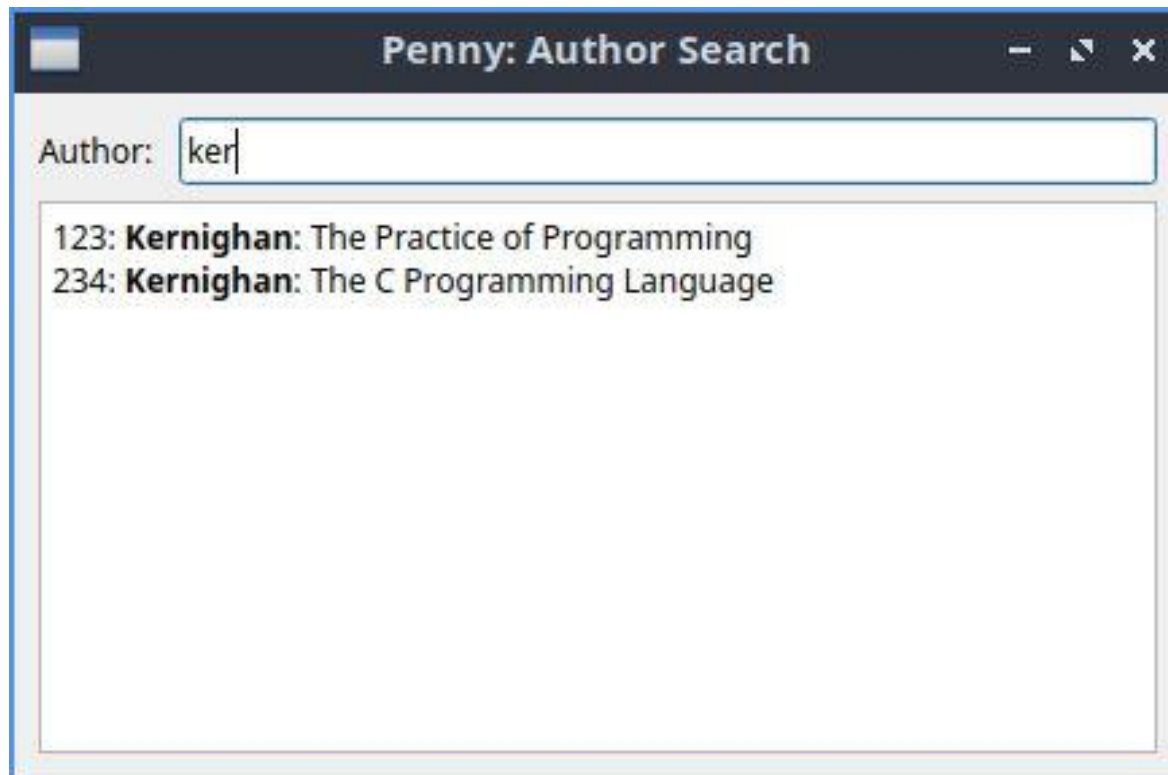
# Penny Desktop v4

- Penny desktop v4 (cont.)



# Penny Desktop v4

- Penny desktop v4 (cont.)





# Penny Desktop v4

- Penny desktop v4 (cont.)



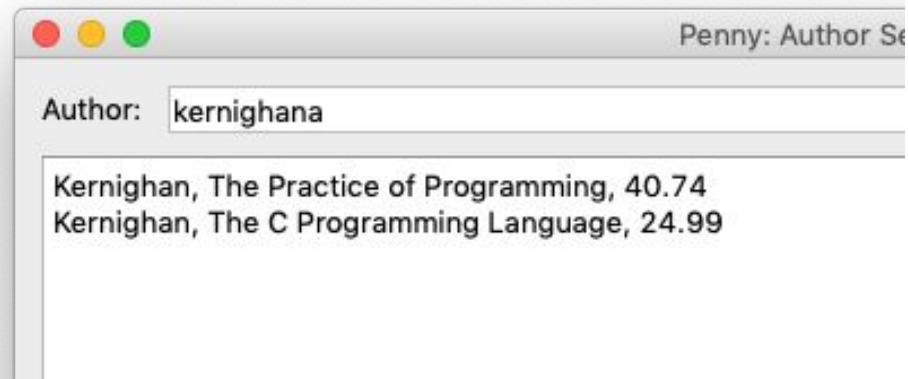
# Penny Desktop v4

- Penny desktop v4 (cont.)
  - See [pennydesktop4threads.py](#)

# Penny Desktop v4

- **Problem:**
  - Server will respond to requests in arbitrary order

old  
screen  
shot



- **Solution:**
  - Abort previous request

# Agenda

- Qt and PyQt5
- PennyJson server
- **Penny desktop client**
  - Version 1: Baseline
  - Version 2: Sequential
  - Version 3: Bad
  - Version 4: Multi-threaded
  - **Version 5: Stop**
  - Version 6: Debouncing

# Penny Desktop v5

- Penny desktop v5

```
$ export IO_DELAY=0  
$ python runserver.py 55555
```

```
$ python pennydesktop5stop.py https://localhost:55555
```

[Same screen images]

# Penny Desktop v5

- Penny desktop v5 (cont.)
  - See [pennydesktop5stop.py](#)

# Penny Desktop v5

- **Problem:**
  - Server could be overwhelmed with requests
- **Solution:**
  - *Debounce* the requests
- **Bonus:**
  - Reduces (but does not eliminate) the need to abort old requests!

# Agenda

- Qt and PyQt5
- PennyJson server
- **Penny desktop client**
  - Version 1: Baseline
  - Version 2: Sequential
  - Version 3: Bad
  - Version 4: Multi-threaded
  - Version 5: Stop
  - **Version 6: Debouncing**



# Penny Desktop v6

- Penny desktop v6

```
$ export IO_DELAY=0  
$ python runserver.py 55555
```

```
$ python pennydesktop6debounce.py https://localhost:55555
```

[Same screen images]

# Penny Desktop v6

- Penny desktop v6 (cont.)
  - See [pennydesktop6debounce.py](#)

# Summary

- Note:
  - If a web application delivers JSON (instead of HTML), then...
  - The client reasonably can be:
    - A browser (given HTML/JavaScript code from the server)
    - A desktop client

# Summary

- We have covered:
  - Desktop programming
  - A Penny **desktop** client
    - A desktop client that works with the Penny server