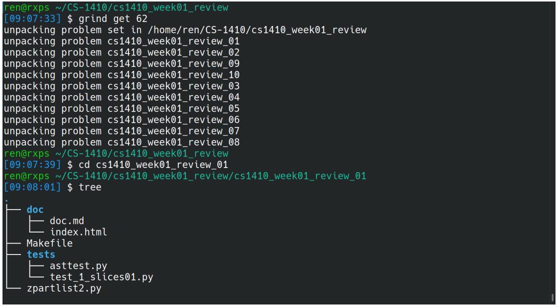
A Web-Based Development Environment for Beginning Python Programmers

Duy Huynh, Ren Quinn

Department of Computing

Code Grinder [1]

- Daily active-learning exercises for students
- Analogy: practicing a musical instrument



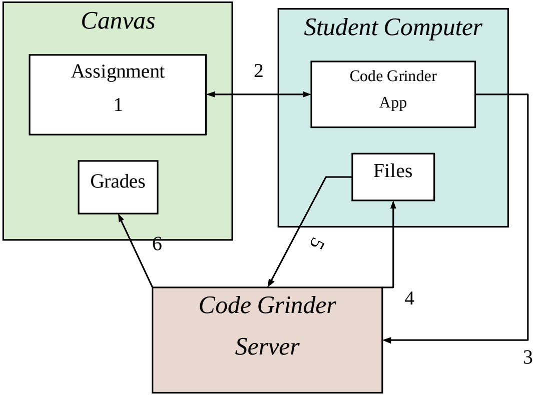
```
ren@rxps ~/CS-1410/cs1410_week01_review
[09:07:33] $ grind get 62
unpacking problem set in /home/ren/CS-1410/cs1410_week01_review
unpacking problem cs1410_week01_review_01
unpacking problem cs1410_week01_review_02
unpacking problem cs1410_week01_review_09
unpacking problem cs1410_week01_review_10
unpacking problem cs1410_week01_review_03
unpacking problem cs1410_week01_review_04
unpacking problem cs1410_week01_review_05
unpacking problem cs1410_week01_review_06
unpacking problem cs1410_week01_review_07
unpacking problem cs1410_week01_review_08
ren@rxps ~/CS-1410/cs1410_week01_review
[09:07:39] $ cd cs1410_week01_review_01
ren@rxps ~/CS-1410/cs1410_week01_review/cs1410_week01_review_01
[09:08:01] $ tree
.
├── doc
│   ├── doc.md
│   └── index.html
├── Makefile
├── tests
│   ├── asttest.py
│   └── test_1_slices01.py
└── zpartlist2.py
```

[1] Created by Russ Ross

Code Grinder Workflow

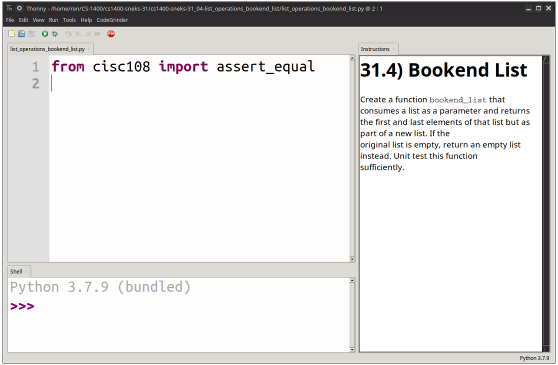
"Close the loop"

1. Starts as assignment in Canvas
2. Student app syncs with Canvas
3. Student app syncs with Code Grinder
4. Downloads files for assigned work
5. Student submits solution for grading
6. Sends grade to Canvas



Thonny

- Integrated Development Environment (IDE)
    - Integrates developer tools into single environment
- Thonny: Python IDE for beginners
- Extension for Code Grinder


Thonny - /home/ren/CS-1400/cs1400-sneks-31/cs1400-sneks-31_04-list_operations_bookend_list/list_operations_bookend_list.py @ 2 : 1
File
Edit
View
Run
Tools
Help
CodeGrinder
list_operations_bookend_list.py
Instructions
Shell
Python 3.7.9

```
from cisc108 import assert_equal

```

```
Python 3.7.9 (bundled)
>>>
```

31.4) Bookend List

Create a function `bookend_list` that consumes a list as a parameter and returns the first and last elements of that list but as part of a new list. If the original list is empty, return an empty list instead. Unit test this function sufficiently.

Problem: Need for a more portable IDE

Need	Use Case
Beginning Programmers	Lowers initial overhead (cost and complexity)
Outreach Programs	Short-lived experiences for kids
Project Demonstrations	short-lived code sharing/collaboration
Assessment	Testing center

Web Assembly (Wasm)

W3C Standard [2]

- Write web applications with non-web tools/languages
- Emscripten

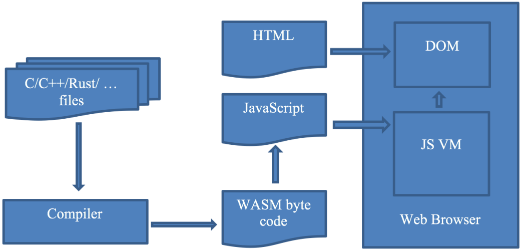


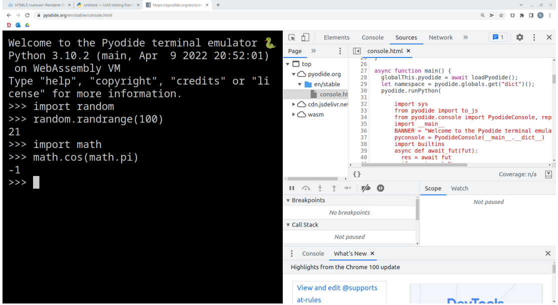
Figure 1: diagram

[2] https://www.w3.org/TR/wasm-core-1

Pyodide [3]

Python compiled to Wasm

```
async function main() {
  let pyodide = await loadPyodide();
  // Pyodide is now ready to use...
  console.log(pyodide.runPython(`
    import sys
    sys.version
  `));
};
main();
```



```
Welcome to the Pyodide terminal emulator 🐍
Python 3.10.2 (main, Apr  9 2022 20:52:01)
 on WebAssembly VM
Type "help", "copyright", "credits" or "li
cense" for more information.
>>> import random
>>> random.randrange(100)
21
>>> import math
>>> math.cos(math.pi)
-1
>>> 
```

```
async function main() {
  globalThis.pyodide = await loadPyodide();
  let namespace = pyodide.globals.get("dict")();
  pyodide.runPython(
    `
      import sys
      from pyodide import to_js
      from pyodide.console import PyodideConsole, rep
      import __main__
      BANNER = "Welcome to the Pyodide terminal emula
      pyconsole = PyodideConsole(__main__.__dict__)
      import builtins
      async def await_fut(fut):
        res = await fut
```

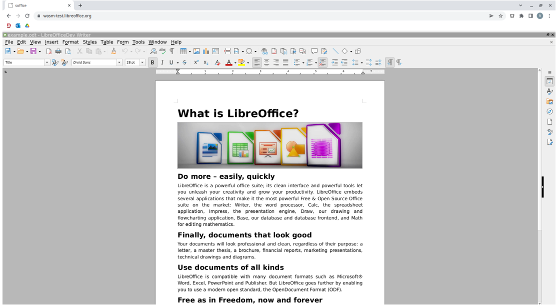
[3] https://pyodide.org

Background & Related Work

- Lack of work on Wasm-based GUIs

Background & Related Work

- Lack of work on Wasm-based GUIs
- Libre Office
    - uses Qt


soffice
wasm-test.libreoffice.org
example.odt - LibreOfficeDev Writer
File Edit View Insert Format Styles Table Form Tools Window Help
Title
Droid Sans
28 pt

What is LibreOffice?

Do more – easily, quickly

LibreOffice is a powerful office suite; its clean interface and powerful tools let you unleash your creativity and grow your productivity. LibreOffice embeds several applications that make it the most powerful Free & Open Source Office suite on the market: Writer, the word processor, Calc, the spreadsheet application, Impress, the presentation engine, Draw, our drawing and flowcharting application, Base, our database and database frontend, and Math for editing mathematics.

Finally, documents that look good

Your documents will look professional and clean, regardless of their purpose: a letter, a master thesis, a brochure, financial reports, marketing presentations, technical drawings and diagrams.

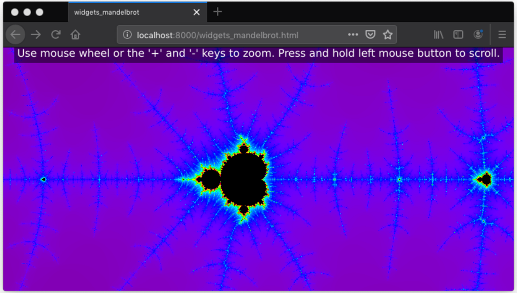
Use documents of all kinds

LibreOffice is compatible with many document formats such as Microsoft® Word, Excel, PowerPoint and Publisher. But LibreOffice goes further by enabling you to use a modern open standard, the OpenDocument Format (ODF).

Free as in Freedom, now and forever

Background & Related Work

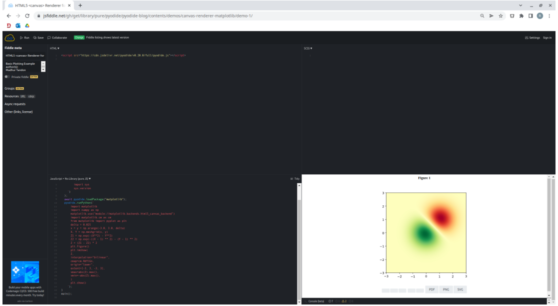
- Lack of work on Wasm-based GUIs
- Libre Office
    - uses Qt
- Mandelbrot Demo
    - also Qt



Use mouse wheel or the '+' and '-' keys to zoom. Press and hold left mouse button to scroll.

Background & Related Work

- Lack of work on Wasm-based GUIs
- Libre Office
  - uses Qt
- Mandelbrot Demo
  - also QT
- Matplotlib Wasm backend
  - uses HTML5 canvas for plotting
  - UI elements are made with traditional html elements



jsfiddle.net/gh/get/library/pure/pyodide/pyodide-blog/contents/demos/canvas-renderer-matplotlib/demo-1/

HTML5 <canvas> Renderer for

Basic Plotting Example
author(s):
Madhur Tandon

```html
<script src="https://cdn.jsdelivr.net/pyodide/v0.20.0/full/pyodide.js"></script>
```

```javascript
          import sys
          sys.version
      `)
    );
    await pyodide.loadPackage("matplotlib");
    pyodide.runPython(`
        import matplotlib
        import numpy as np
        matplotlib.use("module://matplotlib.backends.html5_canvas_backend")
        import matplotlib.cm as cm
        from matplotlib import pyplot as plt
        delta = 0.025
        x = y = np.arange(-3.0, 3.0, delta)
        X, Y = np.meshgrid(x, y)
        Z1 = np.exp(-(X**2) - Y**2)
        Z2 = np.exp(-((X - 1) ** 2) - (Y - 1) ** 2)
        Z = (Z1 - Z2) * 2
        plt.figure()
        plt.imshow(
        Z,
        interpolation="bilinear",
        cmap=cm.RdYlGn,
        origin="lower",
        extent=[-3, 3, -3, 3],
        vmax=abs(Z).max(),
        vmin=-abs(Z).max(),
        )
        plt.show()
      `);
  }
  main();
```

Simplified Approach

- Pyodide
- Code Grinder
- Javascript
- HTML elements

<video>

Acknowledgements

This work is supported by the Dixie State University Research Office and Office of Academic Affairs.