



Docker Desktop in Python



Allen Thomas Varghese
12th Nov 2023



Introduction

- Introduce a desktop GUI library
- Build a python based GUI for Docker Desktop
- Perform basic operations related to docker containers



Required Experience

- Familiarity with Python
- Object Oriented programming
- Use of packages
- Familiarity with Docker



Agenda

- Install dependencies
- UI Walkthrough
- PySimpleGUI introduction
- Problem definition and breakdown
- Creating the UI
- Exploring docker python package
- Connecting docker client to UI elements
- Trigger actions in a table



Install Dependencies

- Required software
 - [Docker Desktop](#)
 - [VS Code IDE](#)
 - [Python 3.9+](#)
- Create a new folder called `pycon2023-docker-desktop`
- Inside the folder, create python virtual environment: `python -m venv .venv`
- Activate virtualenv: `./.venv/bin/activate` or `./.venv\Scripts\activate`
- Install required python packages: `pip install pysimplegui docker`
- Save PySimpleGUI cookbook as PDF: pysimplegui.org/en/latest/cookbook as reference



Use Powershell for running all commands

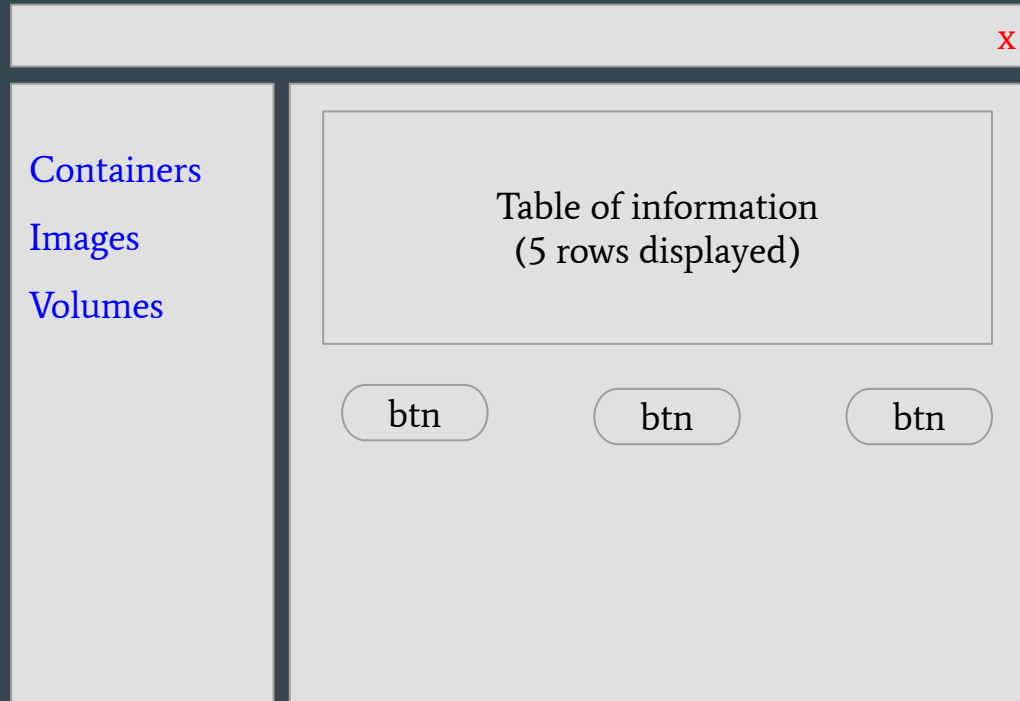


Folder Structure

Within the workspace folder `pycon2023-docker-desktop` create the following:

- `src` folder: for all source files
- `requirements.txt` file: for keeping track of package dependencies
 - `docker==6.1.3`
 - `pysimplegui==4.60.5`
- `src/screen.py` file: entrypoint for the application

UI Prototype



Phase 1

Display Window

Output





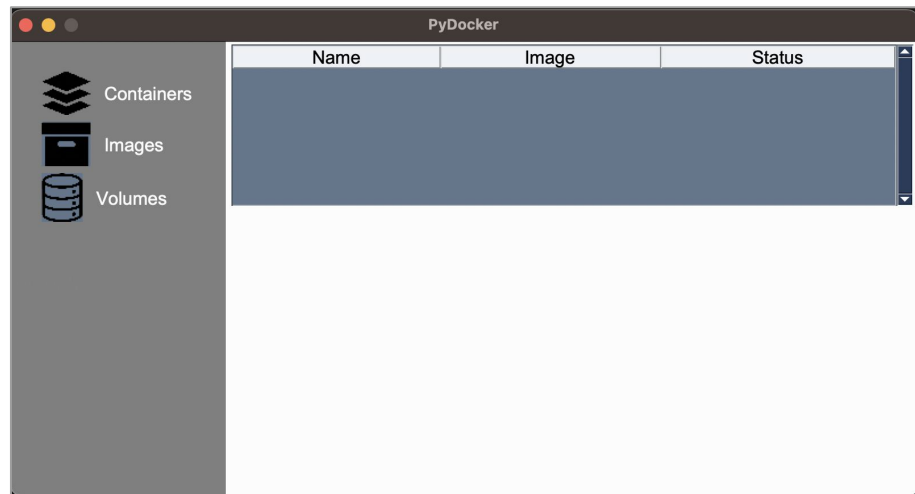
Steps

- Open the PySimpleGUI cookbook PDF
- Search for creating a window
- Copy the code to `screen.py` file
- Run the above file

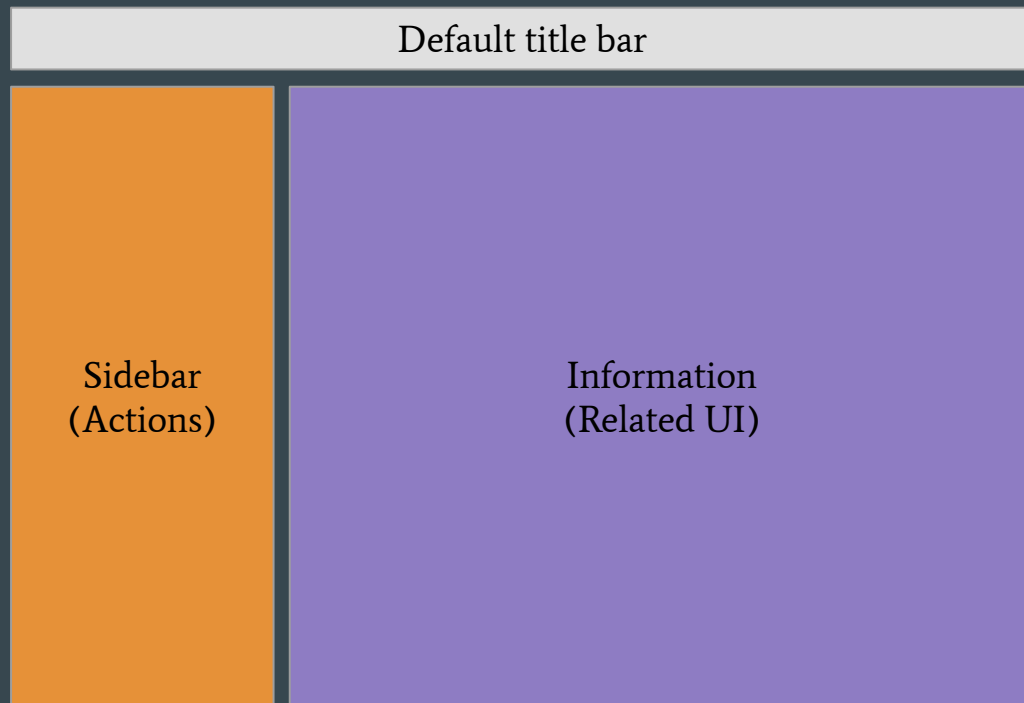
Phase 2

Sidebar and Info UI

Output



UI Layout





Steps

- Open the PySimpleGUI cookbook PDF
- Explore how layouts work in PySimpleGUI (hint: list of lists)
- Create the sidebar as a component and render it
- Create the info section as a component and render it next to the sidebar
- Add a table to the info section
- Convert the info component to a class to handle click events

Use icons from any website that offers free download



Container Information

Docker Desktop

Upgrade plan

Search for images, containers, volumes, extensions and more...

%K

Sign in

Containers

Images

Volumes

Dev Environments

BETA

Docker Scout

Learning center

Builds

BETA

Extensions

Containers

[Give feedback](#)

Container CPU usage ⓘ
No containers are running.

Container memory usage ⓘ
No containers are running.

Show charts ▾

Search

Only show running containers

	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<div></div>	<div></div>	<div><div>nostalgi</div><div>453a978b</div></div> <div><div>alpine:latest</div></div>	Exited	N/A		9 seconds ago	<div><div></div><div></div><div></div></div>

Image Information



Containers

Images

Volumes

Dev Environments BETA

Docker Scout

Learning center

Builds BETA

Extensions

Extensions are disabled

Upgrade plan

Search for images, containers, volumes, extensions and more...

%%K

Sign in

Images

[Give feedback](#)

Local

Hub

Artifactory

EARLY ACCESS

8.01 MB / 9.87 GB in use

5 images

Last refresh: 3 minutes ago

Search

	Name	Tag	Status	Created	Size	Actions
<input type="checkbox"/>	alpine	latest	In use	11 days ago	11.42 MB	<div></div> <div></div> <div></div>
<input type="checkbox"/>	eece025e4321					<div></div> <div></div> <div></div>
<input type="checkbox"/>	neo4j	community	Unused	27 days ago	801.33 MB	<div></div> <div></div> <div></div>
<input type="checkbox"/>	98858e8c66dc					<div></div> <div></div> <div></div>

Volume Information



Containers

Images

Volumes

Dev Environments BETA

Docker Scout

Learning center

Builds BETA

Extensions

Upgrade plan

Search for images, containers, volumes, extensions and more...

K

Sign in

Volumes

[Give feedback](#)

Create

Search

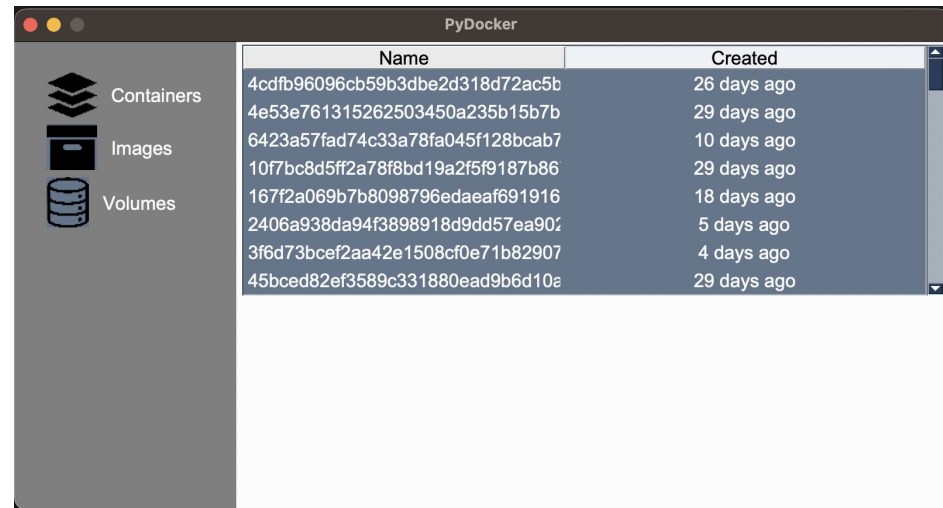
<input type="checkbox"/>	Name	Status	Created	Size	Actions
<input type="checkbox"/>	dd6997624160411675d7d19860aace92467cc61802a1f86fc2831 -		1 hour ago	45.6 MB	
<input type="checkbox"/>	7ac9c310301ef426348c269f7646fa54677428fbf1fb0a0a0679a8 -		2 hours ago	46.2 MB	
<input type="checkbox"/>	9c312b52abe205e41132b46313a947cc873713bdc278fbce34e2 -		2 hours ago	46.2 MB	



Output

Phase 3

Load Info based on actions



The image shows a screenshot of the PyDocker application window. On the left sidebar, there are three icons: a stack of three rectangles for 'Containers', a single rectangle for 'Images', and a cylinder for 'Volumes'. The 'Containers' section is selected. The main area displays a table with two columns: 'Name' and 'Created'. The table contains eight rows of container data, each with a long alphanumeric ID and a creation time relative to 'now'.

Name	Created
4cdfb96096cb59b3dbe2d318d72ac5b	26 days ago
4e53e761315262503450a235b15b7b	29 days ago
6423a57fad74c33a78fa045f128bcab7	10 days ago
10f7bc8d5ff2a78f8bd19a2f5f9187b86	29 days ago
167f2a069b7b8098796edaeaf691916	18 days ago
2406a938da94f3898918d9dd57ea90f	5 days ago
3f6d73bcef2aa42e1508cf0e71b82907	4 days ago
45bcd82ef3589c331880ead9b6d10e	29 days ago



Docker Python Package

- Requires docker desktop running in the background
- Can retrieve information about containers, images and volumes
- Can trigger operations
- Details available at github.com/docker/docker-py
- In a python shell, explore how to fetch information about:
 - All containers
 - All images
 - All volumes



Steps

- Add a new python module that will handle docker interactions
- Create a class with methods for fetching:
 - List of active containers
 - List of all images
 - List of all volumes
- Parse and render the above information when the specific link is clicked. For ex: load images when the images button is clicked
- Show only the fields in the table that has information from the docker package
- Use the **humanize** package for showing relative time like 2 hours ago

Phase 4

Trigger docker operations



Output



PyDocker




 Containers
 Images
 Volumes

Image	Commit	Tag	Size	Created
alpine	eece025e4321	latest	3.25 MB	a month ago
node	8e015de364a2	20-alpine	52.99 MB	16 days ago
nginx	86e53c4c16a6	latest	67.28 MB	9 days ago
python	9e00960bde4d	3.11	360.62 MB	26 days ago
neo4j	98858e8c66dc	community	279.11 MB	a month ago

Actions

[Start Container](#)

Enter docker image name [Download](#)



Steps

- Add 2 buttons to the Images section
 - Start Container
 - Download Image

Actions

- The start container button should trigger a container for the selected image
- There should be a text box to fetch the image name
- The download image button should be next to the text box
- Clicking the download button should pull the specified image
- Show a popup if there is an error with the pull operation