



# Streamlit

## Introduction ML in PL 23

# Prerequisites

- Install Anaconda
- Open Anaconda Prompt
- Optional: Activate new environment
  - `conda create -n streamlit python=3.9`
  - `conda activate streamlit`
- `pip install streamlit scikit-learn`
- `streamlit version`



# Fanilo Andrianasolo



**WORLDLINE** 

Data & AI Strategy

 Streamlit

3 Year old Fan 



@andfanilo

# Safe & Fun Atmosphere for Practicing



Help Each Other 🤝



Questions Welcome 🙋



## **15mn Coffee Break**

**11:00 am**

## **5mn small breaks**

**10:00 am**

**12:00 am**

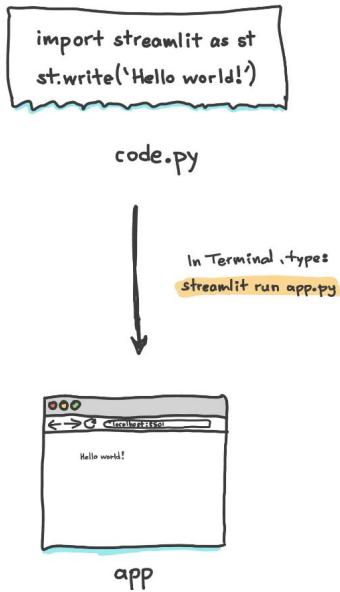
# What About You?



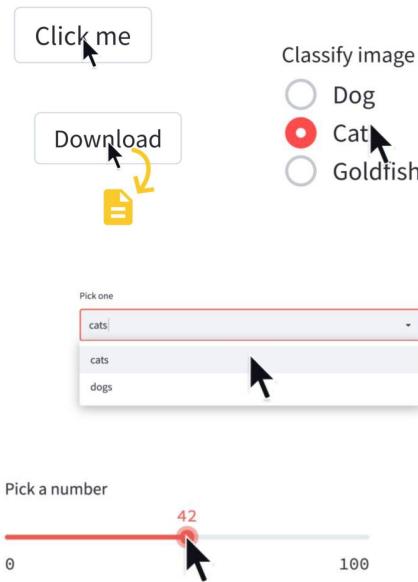
<https://pollev.com/fanand256>

# What we will learn

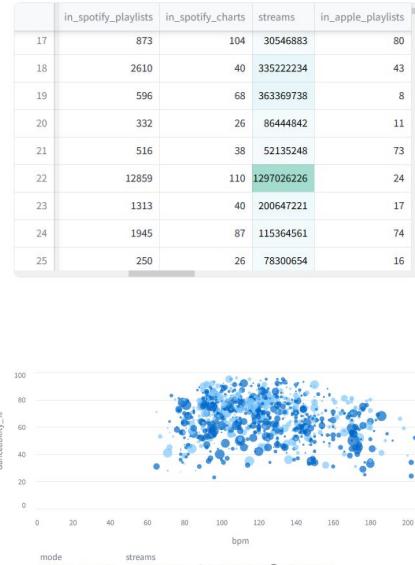
## Create your First Streamlit App



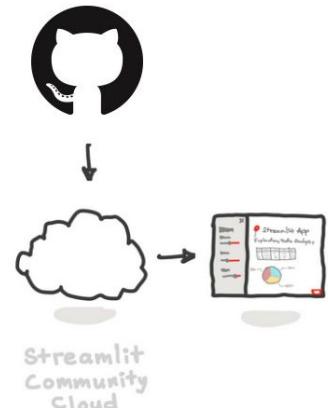
## Use a variety of Interactive Widgets



## Display Dataframes and Plots



## Deploy on Streamlit Cloud



# We Won't Have Time to See...

Testing Streamlit Components

Interacting with ML Libraries / APIs

Advanced visualizations (cross-filtering, maps)

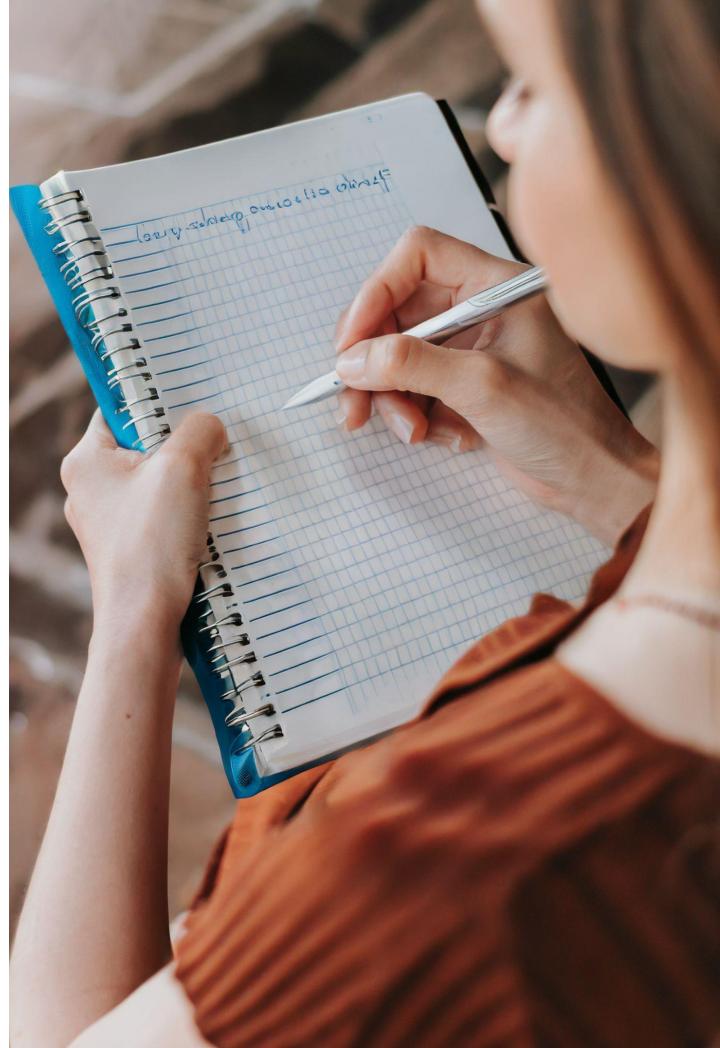
Hacking Streamlit style



but you'll know where to look for it

# Agenda

- Intro to Streamlit
- Play with (a lot of!) Streamlit Widgets
- Case 1: Explore Spotify Data
- Case 2: Deploy Spotify ML Tuning Board
- Wrapping up

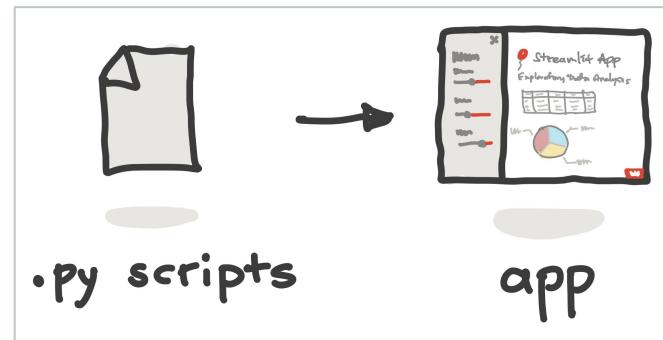




# Enter Streamlit

# From Script to Web App

- Embrace Scripting
- No HTML / CSS / JS experience needed
- Integrates well with the Data Science packages



Thank you Streamlit for all the cartoons!

# Examples from the Community



RAScore

**Navigation**  
Go To ->  
● Webcam  
○ Image

**Options**  
Choose the style model:  
Starry\_night

QUALITY (May reduce the speed)  
150 150 500

Neural Style Transfer

Webcam Live Feed

STOP SELECT DEVICE

WebRTC

**Meow**

I iz CatGDP, meow-speak anypawdy to me and I'll purr-ly there with a paw-some meow reply.

Tell me a good joke!

Meeow, what do you call a cat who can sing? A meow-sician! Meow!

Purrhaps that joke didn't land quite right. Let me try again! Meow!

O\_o what?

Purr:

Page views: 01192  
Unique visitors: 04114  
GitHub: Stars: 34

CatGDP

# Roadmap into Streamlit

`st.write`

`st.write(text)`

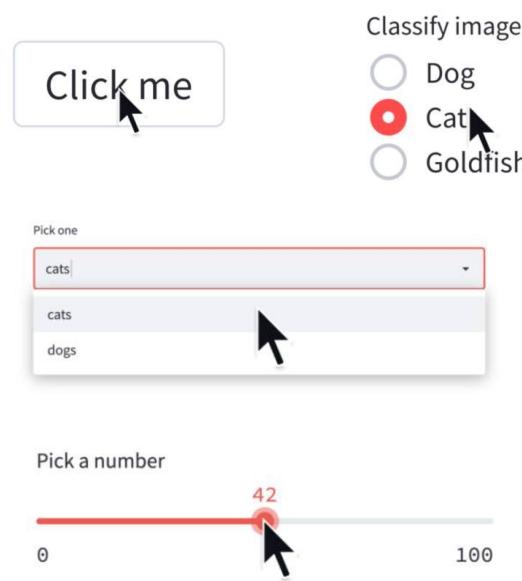
`st.write(df)`

`st.write(chart)`

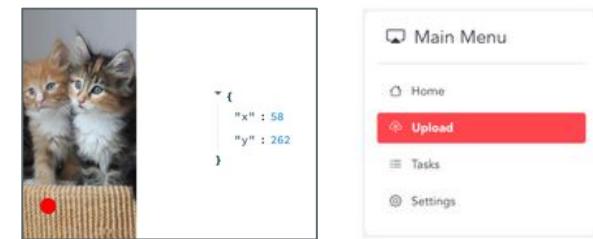
`st.write(func)`



`st.<widget>`



`Components`



**Drawable Canvas**

Draw on the canvas, get the drawings back to Streamlit!

- Doubleclick to remove the selected object when not in drawing mode

streamlit\_app.py

```
import streamlit as st  
  
st.write("# Hello world 🎈")  
st.balloons()
```

Console (ensure you're in the correct Conda environment)

```
> streamlit version  
  
> streamlit run streamlit_app.py
```

# Get Started!

---

# Playground time

# Hands On Discovery

- Download the code
- Run it with the streamlit command
- Start chipping through the questions



<https://rb.gy/e9odp>

[https://github.com/andfanilo/  
ml-in-pl-23](https://github.com/andfanilo/ml-in-pl-23)

# Summary - Streamlit Lifecycle

```
import streamlit as st  
  
st.title("Hello world 🎈")  
  
button_clicked = st.button("Send Balloons")  
  
if button_clicked:  
    st.balloons()
```

streamlit run streamlit\_app.py

Hello world 🎈

Send Balloons

Reruns script from top to bottom  
with button\_clicked state updated

# Explore Data with Widgets

# Streamlit EDA over Spotify Data

Upload CSV file

Drag and drop file here  
Limit 200MB per file

Browse files

spotify-2023.csv 103.7KB

X

Preview Dataframe

Hide columns

artist(s)\_name X artist\_count X released\_year X released\_month X

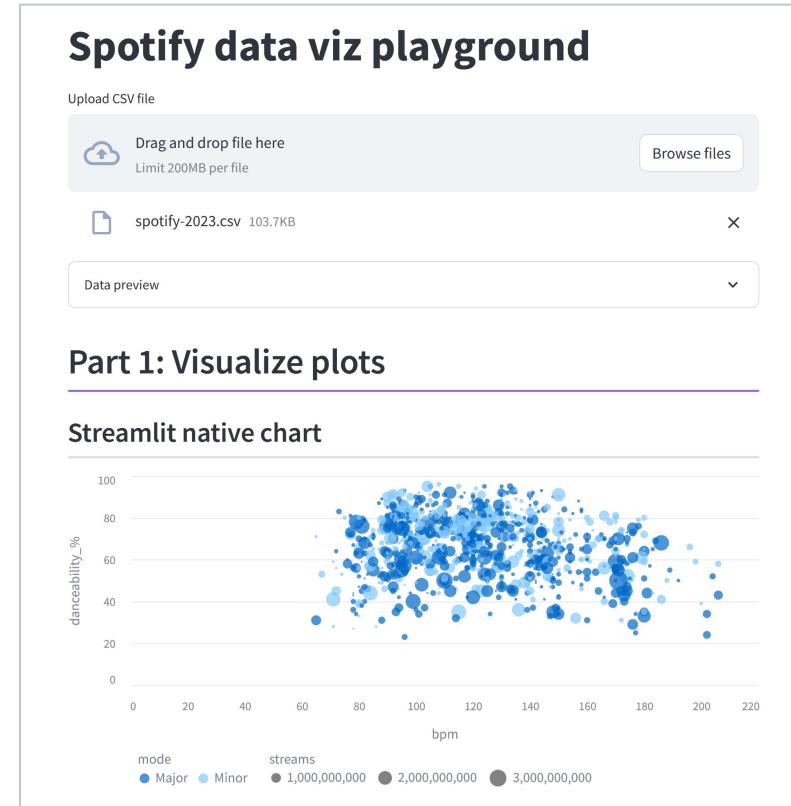
Keep songs with number of streams over:

1000000000

0 4000000000

	track_name	released_day	in_spotify_playlists	in_spotify_charts	streams	in_apple_playlist
12	Flowers	12	12,211	115	1,316,855,716	
14	As It Was	31	23,575	130	2,513,188,493	
15	Kill Bill	8	8,109	77	1,163,093,654	
22	I Wanna Be Yours	1	12,859	110	1,297,026,226	

Download data as CSV



# Summary - Advanced Features

- |                |   |  |
|----------------|---|--|
| Forms          | • | • Arrange widgets in containers          |
| Layout         | • | • Prevent Streamlit rerun on interaction |
| Secrets        | • | • Save interactions between reruns       |
| Cache          | • | • Display image from folder              |
| Session State  | • | • Fetch credentials from file            |
| Static serving | • | • Save function results between reruns   |

# Deploy A ML App

# What will you build?

Model Configuration

of Jobs: 1 of Estimators: 25

Criterion: gini Max Features: sqrt

Max Depth: 25

% of Samples: 0.80

Min Samples Split: 50 Min Samples Leaf: 50

Bootstrap

Click here to run model

Deploy

## Spotify: Predict in Spotify chart

Data preview

Model ran in: 0.2021 seconds

Training Score	Test Score	Precision	Recall	F1
60.6%	59.05% <span style="color:red">↓ -1.55%</span>	61.23%	88.48%	72.38%

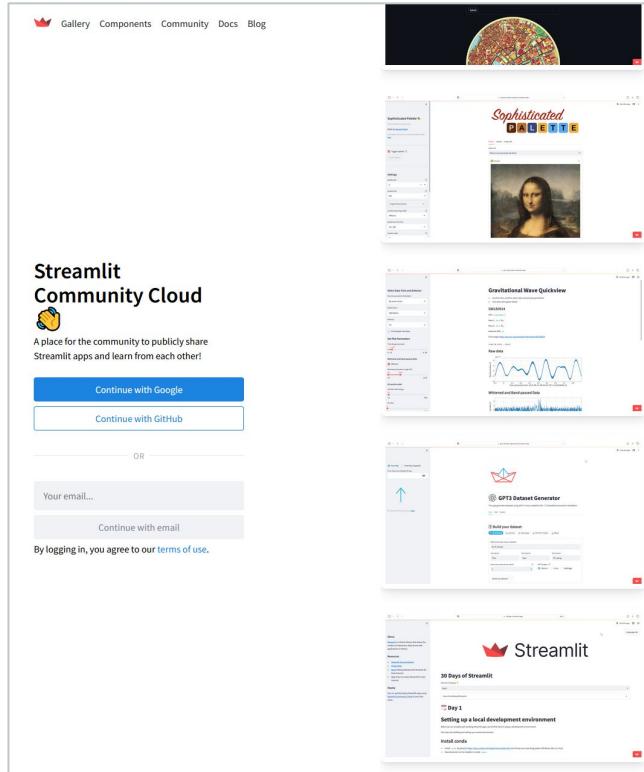
**Confusion Matrix**

		Predicted	
		Negative	Positive
Actual	Negative	169	22
	Positive	107	17

**ROC Curve (AUC = 0.54)**

The ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR). The area under the curve (AUC) is 0.54. The x-axis is labeled "False Positive Rate (FPR)" and ranges from 0.0 to 1.0. The y-axis is labeled "True Positive Rate (TPR)" and ranges from 0.0 to 1.0. The curve starts at (0,0) and ends at (1,1), passing through several points including approximately (0.1, 0.15), (0.2, 0.25), (0.3, 0.35), (0.4, 0.45), (0.5, 0.55), (0.6, 0.65), (0.7, 0.75), (0.8, 0.85), (0.9, 0.95), and (1.0, 1.0).

# Where will you deploy?



Prerequisite: Github/Google Account

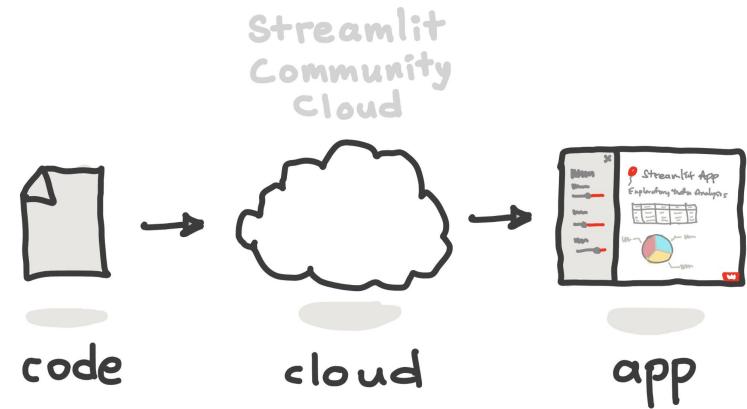


<https://share.streamlit.io/>

# Summary - Deployment

Streamlit apps can be deployed everywhere:

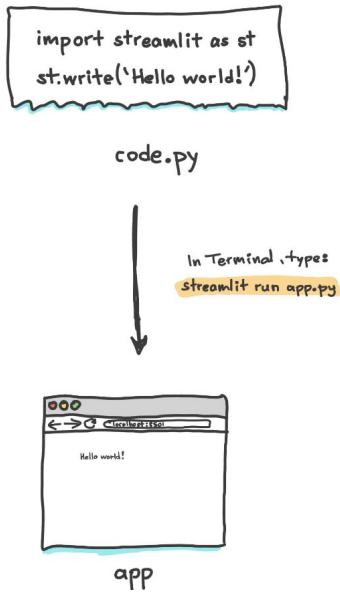
- Locally
- Cloud-host
  - [Streamlit Community Cloud](#)
  - [HF Spaces](#)
- Self-host
  - [AWS](#), [Azure](#), [GCP](#), etc.
- In-browser
  - [Stlite](#)



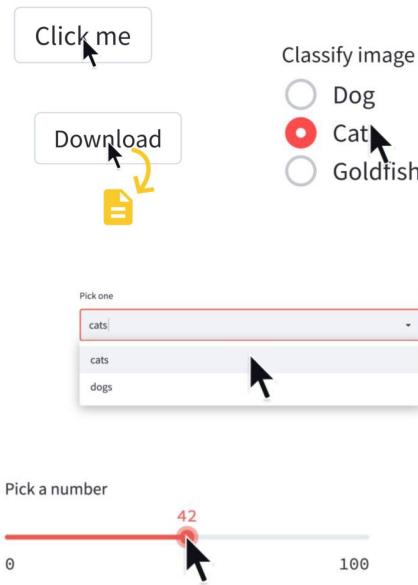
# Wrapping Up

# What we learned

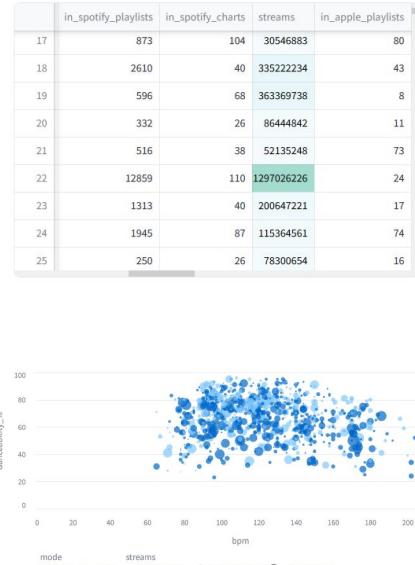
## Create your First Streamlit App



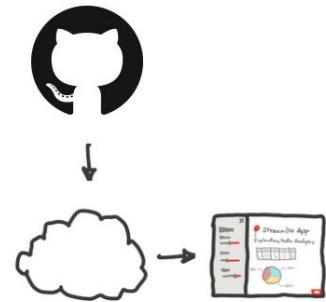
## Use a variety of Interactive Widgets



## Display Dataframes and Plots



## Deploy on Streamlit Cloud



Streamlit  
Community  
Cloud

# The One Key Takeaway

```
import streamlit as st  
  
st.title("Hello world 🎈")  
  
button_clicked = st.button("Send Balloons")  
  
if button_clicked:  
    st.balloons()
```

streamlit run streamlit\_app.py

Hello world 🎈

Send Balloons

Reruns script from top to bottom  
With button\_clicked state updated

# Showcase Your Works to the



Cartoon Style  
● Ribbon  
○ Trace

Cartoon Transparency

Surface Transparency (Mutations Only)

Powered by [Smpj](#) (PDB: [1HE8](#)).

SW1 Conformation: [Y32in.3P.ON](#)

SW2 Conformation: [Y71in.3P.R](#)

Y32 Position: [Y32in](#)

Y71 Position: [Y71in](#)

Build Apps

andfanilo's apps

New app ▾

- lyon2 · master · streamlit\_app.py
- s4a-dl-yt-thumbnail · master · streamlit\_app.py
- s4a-twitter-engagement · main · streamlit\_app.py
- streamlit-drawable-canvas-demo · master · app.py
- streamlit-echarts-demo · master · app.py
- streamlit-lottie-demo · master · app.py
- s4a\_cats\_grid · main · app.py

Deploy for Free

CANCER RESEARCH | CONVERGENCE AND TECHNOLOGIES

Delineating the RAS Conformational Landscape

Mitchell I. Parker<sup>1,2</sup>, Joshua E. Meyer<sup>1,3</sup>, Erica A. Golemis<sup>1,4</sup>, and Roland L. Dunbrack, Jr.<sup>1</sup>

ABSTRACT

Mutations in RAS isoforms (KRAS, NRAS, and HRAS) are among the most frequent oncogenic alterations in many cancers, making these proteins high priority therapeutic targets. Efficiently targeting these proteins requires a detailed understanding of their active, inactive, and druggable conformations. However, there is no monolithic catalog of RAS conformations available for therapeutic targeting, nor is the structural impact of RAS mutations known. Here we present an expanded classification of RAS conformations based on the spatial positions of Y32 in SW1 and Y71 in SW2 loops. From 721 human KRAS, NRAS, and HRAS structures available in the Protein Data Bank (PDB), RAS–protein complexes, 1999 RAS–GDP complexes, and 100 RAS–GTP complexes (including mutated structures), we created a broad conformational classification based on the spatial positions of Y32 in SW1 and Y71 in SW2. Classifications were generated using a hierarchical decision tree-based machine learning algorithm defined additional conformational subtypes that could not be predicted from SW1 conformations alone. These SW2 conformations were identified and associated with different nucleotide states (GTP, GDP, GTPyS, and GTPyF) and GTP-bound SW1 conformation codes, the basis of the hydrogen bond type (HBT) between the side chains of GL12 and GL2V mutations and the nucleotide, and the ability to bind to each druggable RAS conformation. This classification provides a detailed analysis of RAS conformations that could facilitate future RAS drug discovery.

Significance Analysis of >7000 RAS conformations revealed an expanded landscape of active, inactive, and druggable conformations.

J. M. Nápoles  
@napoles3D

Amazing use case of [@streamlit](#) as a complement to a scientific publication by [@Mitch\\_P](#) 😊

[share.streamlit.io/mitch-parker/r...](#)

Main Menu

Home Page

Database-Related Links

bioRxiv Paper

GitHub Page

PDB Protein Data Bank

Community-Related Links

Rascore

A tool for analyzing RAS protein structures

Created by Mitchell Parker and Roland Dunbrack

Fox Chase Cancer Center

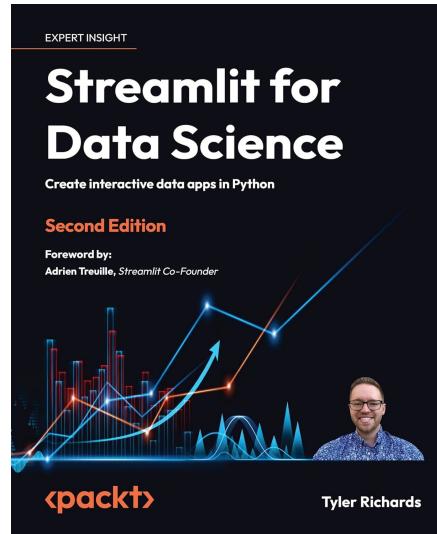
Summary

3:39 PM - Mar 16, 2022

Embed into publications / websites / Medium

# Resources to Dive Deeper

# Build POWERFUL apps



# Are there things to clear up?



# One Positive Takeaway

More critical feedback  
accepted after the session



<https://pollev.com/fanand256>



Happy  
Streamlitin'

@andfanilo