

Getting Started with Streamlit for Data Science: A Comprehensive Guide for Beginners

Streamlit is an open-source Python library that makes it easy to create interactive data science applications. With Streamlit, you can quickly and easily build apps that display data, generate visualizations, and allow users to interact with your data.



Getting Started with Streamlit for Data Science: Create and deploy Streamlit web applications from scratch in Python by Tyler Richards

★★★★☆ 4.5 out of 5

Language : English
File size : 19129 KB
Text-to-Speech : Enabled
Screen Reader : Supported
Enhanced typesetting : Enabled
Print length : 282 pages



In this guide, we will provide a comprehensive overview of Streamlit, including how to install it, create your first app, and add interactive widgets. We will also provide some tips and best practices for using Streamlit in your data science projects.

Installation

To install Streamlit, simply run the following command in your terminal:

```
pip install streamlit
```

Once Streamlit is installed, you can create your first app by creating a new Python file and saving it with the `.py` extension. For example, you could create a file called `my_first_app.py`.

Creating Your First App

The following code shows a simple Streamlit app that displays the text "Hello, world!":

```
python import streamlit as st

st.write("Hello, world!")
```

To run your app, simply run the following command in your terminal:

```
streamlit run my_first_app.py
```

This will open a new browser window and display your app.

Adding Interactive Widgets

Streamlit provides a variety of interactive widgets that you can use to add interactivity to your apps. For example, you can add buttons, sliders, and text input fields.

To add a button to your app, you can use the `st.button()` function. The following code shows how to add a button that displays the text "Click me!":

```
python import streamlit as st
```

```
st.button("Click me!")
```

To add a slider to your app, you can use the `st.slider()` function. The following code shows how to add a slider that allows users to select a value between 0 and 10:

```
python import streamlit as st
```

```
st.slider("Select a value:", 0, 10)
```

To add a text input field to your app, you can use the `st.text_input()` function. The following code shows how to add a text input field that allows users to enter their name:

```
python import streamlit as st
```

```
st.text_input("Enter your name:")
```

Tips and Best Practices

Here are a few tips and best practices for using Streamlit in your data science projects:

- * Use descriptive widget labels. This will help users understand what each widget does and how it affects your app.
- * Use the `st.write()` function to display data and visualizations. This function will automatically format your data and visualizations in a way that is easy to read and understand.
- * Use the `st.sidebar` to add widgets that are not essential to the main functionality of your app. This will help keep your app organized and easy to use.
- * Test your app regularly to make sure that it is working as

expected. This will help you catch any bugs early on and prevent them from causing problems for your users.

Streamlit is a powerful and easy-to-use library for creating interactive data science applications. With Streamlit, you can quickly and easily build apps that display data, generate visualizations, and allow users to interact with your data.

In this guide, we have provided a comprehensive overview of Streamlit, including how to install it, create your first app, and add interactive widgets. We have also provided some tips and best practices for using Streamlit in your data science projects.

We encourage you to explore the Streamlit documentation and tutorials to learn more about this powerful library. With a little practice, you will be able to create beautiful and interactive data science applications that will help you communicate your insights to others.



Getting Started with Streamlit for Data Science: Create and deploy Streamlit web applications from scratch in Python

by Tyler Richards

★★★★☆ 4.5 out of 5

Language : English
File size : 19129 KB
Text-to-Speech : Enabled
Screen Reader : Supported
Enhanced typesetting : Enabled
Print length : 282 pages

FREE

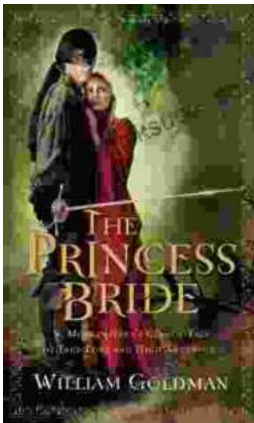
DOWNLOAD E-BOOK





79 ESL Activities, Games, and Teaching Tips for Big Classes (20+ Students)

Teaching large ESL classes can be a challenge, but it's definitely possible with the right strategies. Here are 79 ESL activities, games, and...



Morgenstern: A Classic Tale of True Love and High Adventure

Morgenstern is a classic tale of true love and high adventure. Set in a medieval world, the story follows the journey of Morgenstern, a young...