ShachiPy Streamlit Workshop

Onur Boyar

March 2nd, 2024

Session 1: Introduction to Streamlit

1.1 Overview of Streamlit

What is Streamlit?

- Streamlit is an open-source Python library for creating web apps for data science and machine learning projects.
- It enables rapid prototyping and deployment of interactive data applications.

Key Features and Benefits:

- Write apps with simple Python scripts.
- No need for front-end experience.
- Interactive widgets for live data and plots.
- Easy deployment of apps.

Example Applications:

- Image background remover.
- Chat application
- Molecule optimizer.

1.2 Installation and Setup

Installing Streamlit:

```
pip install streamlit
```

Setting up a basic Streamlit app:
Create a file named `app.py` and add the following code:

```
import streamlit as st

st.write('Hello, Streamlit!')
```

Run your app with:

```
streamlit run app.py
```

1.3 Creating Your First Streamlit App

Structure of a Streamlit Script:

A Streamlit script is a Python script with Streamlit commands.

Basic Components:

Title:

```
st.title('My First Streamlit App')
```

Button:

```
if st.button('Say hello'):
    st.write('Hello, World!')
```

Text and Data Display:

```
st.write('This is a simple text display.')
```

Data Display:

```
import pandas as pd

data = pd.DataFrame({'Column A': [1, 2, 3], 'Column B': [4, 5, 6]})
st.write(data)
```

1.4 Interactive Widgets

Introduction to Widgets:

Widgets allow users to interact with the application dynamically.

Sliders:

```
age = st.slider('How old are you?', 0, 130, 25)
st.write("You're ", age, 'years old')
```

Buttons:

```
if st.button('Click me'):
    st.write('Button clicked!')
```

Session 2: Advanced Streamlit Concepts and Applications

Customizing the Layout

Using Columns:

```python
col1, col2 = st.columns(2)

with col1:
    st.header('Column 1')
    st.write('Hello, Column 1!')

with col2:
    st.header('Column 2')
    st.write('Hello, Column 2!')
```

Real-world Application Examples

Creating a Calculator Application with Streamlit:

```python
st.title('Simple Calculator')

# Inputs
num1 = st.number_input('Enter first number', format='%f')
num2 = st.number_input('Enter second number', format='%f')

# Operations
operation = st.selectbox('Choose an operation:', ['Add', 'Subtract'
    , 'Multiply', 'Divide'])

# Calculate
if st.button('Calculate'):
    if operation == 'Add':
        result = num1 + num2
    elif operation == 'Subtract':
        result = num1 - num2
    elif operation == 'Multiply':
        result = num1 * num2
    elif operation == 'Divide':
        if num2 != 0:
            result = num1 / num2
        else:
            result = "Error: Division by zero"
    st.success(f'Result: {result}')
```

Tips for Advanced Applications & Exploring the Streamlit App Gallery

- Explore the Streamlit documentation for advanced features like caching, file uploads, and creating custom components.
- Visit the Streamlit App Gallery to see real-world applications and get inspired.

Additional Example Applications

Data Visualization App

```

import streamlit as st
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Set the title of the app
st.title('Data Science App with Streamlit')

# Generate sample data
@st.cache  # Use caching to generate the data only once
def generate_data(n_rows, n_cols):
    """Generates a DataFrame with random data"""
    dates = pd.date_range(start="2021-01-01", periods=n_rows, freq="D")
    data = np.random.randn(n_rows, n_cols)
    columns = [f"Column_{i}" for i in range(1, n_cols + 1)]
    return pd.DataFrame(data, index=dates, columns=columns)

# User input for the size of the dataset
n_rows = st.sidebar.slider("Number of rows", min_value=10, max_value=1000, value=100, step=10)
n_cols = st.sidebar.slider("Number of columns", min_value=1, max_value=20, value=5, step=1)

# Generate and display the dataframe
df = generate_data(n_rows, n_cols)
st.write("### Generated Data", df)

# Show column statistics
if st.sidebar.checkbox('Show Column Statistics'):
    st.write("### Column Statistics", df.describe())

# Visualization
if st.sidebar.checkbox('Show Histogram'):
    column_to_plot = st.sidebar.selectbox('Select Column to Visualize', df.columns)
    fig, ax = plt.subplots()
    df[column_to_plot].hist(bins=20, ax=ax)
    ax.set_title(f'Histogram of {column_to_plot}')
    st.pyplot(fig)

st.write("This app demonstrates Streamlit's capability for data science tasks using generated data.")
```

Text Analysis App

```
import streamlit as st

st.title('Text Analysis App')

```

```
# Text input
user_input = st.text_area("Enter your text here:")

# Analysis
if st.button('Analyze'):
    char_count = len(user_input)
    word_count = len(user_input.split())

    st.write(f"Character Count: {char_count}")
    st.write(f"Word Count: {word_count}")

st.write("This app counts the number of words and characters in
    your text.")
```

User Feedback Form

```
import streamlit as st

st.title('User Feedback Form')

# User input fields
name = st.text_input('Name')
rating = st.select_slider('Rating', options=['Poor', 'Fair', 'Good'
    , 'Excellent'])
comments = st.text_area('Comments')

# Submit button
if st.button('Submit'):
    st.success(f'Thank you {name}, for your feedback!')

st.write("This app collects user feedback.")
```

Image Flipper

```
import streamlit as st
from PIL import Image, ImageOps

st.title('Image Flipper')

# File uploader
uploaded_file = st.file_uploader("Choose an image...", type=['jpg',
     'jpeg', 'png'])
if uploaded_file is not None:
    image = Image.open(uploaded_file)

    # Display original image
    st.image(image, caption='Uploaded Image', use_column_width=True
    )

    # Options for flipping
    flip_type = st.radio("Flip Type", ('Normal', 'Horizontal', '
    Vertical'))

    if flip_type == 'Horizontal':
```

```
        flipped_image = ImageOps.mirror(image)
    elif flip_type == 'Vertical':
        flipped_image = ImageOps.flip(image)
    else:
        flipped_image = image

    # Display flipped image
    st.image(flipped_image, caption='Flipped Image',
    use_column_width=True)

st.write("This app flips the uploaded image based on the selected
    option.")
```