1**. What is Streamlit, and how does it differ from other web frameworks?**

Streamlit is a popular Python open source library used for building web applications for data science and machine learning projects.It helps developers to build web applications easily and comfortably. Deployment of the Streamlit app is easy compared to other web frameworks like Django,flask. We are able to create interactive apps with just a few lines of Python code. With Streamlit, you write your app like a Python script focusing more on the data and less on the web development part.

Unlike other traditional web frameworks that involve HTML, CSS, and JavaScript, Streamlit use Python code to create widgets, charts, and interactive components. You can easily display data, charts, images.

The app reruns from top to bottom on every user interaction.We can use session state to store variables outside of rerun.Session state helps to share varibales between reruns. In short, Streamlit makes it really easy for data scientists and developers to quickly build web apps .

**2. Explain the basic structure of a Streamlit app**

Streamlit structure mainly involves a python script.First, we install streamlit app using ‘pip install streamlit’.Then Import necessary libraries , including ‘streamlit’ and other libraries for data processing, visualization and all the libraries that are needed in the app.

**Titles and headers**

st.header()-returns a header

st.subheader()-shows a subheader

st.title()-sets a title for the streamlit app

display data

st.write()-can write text,data or markdown in the app

st.text\_input()-accepts a text input

st.number\_input()-accepts a number input

st.sidebar()-creates a side bar

**interactive widgets**

st.button()-creates a button

st.checkbox()-creates a checkbox

st.radio(): Displays radio buttons.

st.slider(): Allows sliding selection.

st.selectbox(): displays a dropdown selection

st.multiselect():can make multiple selections

**file uploader**

st.file\_uploader()-to upload files

st.progress-shows a progress bar

st.spinner()-shows a spinner for indicating the progress

st.map()-creates a map visualization

we can use conditional statements to control display of the elements

ex:

import streamlit as st  
if st.button('Click me!'):  
 st.write('Button clicked!')

you can deploy the using streamlit sharing ,Heroku or aws.

**3. How can you add a title to your Streamlit app?**

import streamlit as st  
  
st.title('Welcome to Streamlit!')

you can add title using st.title()

output:

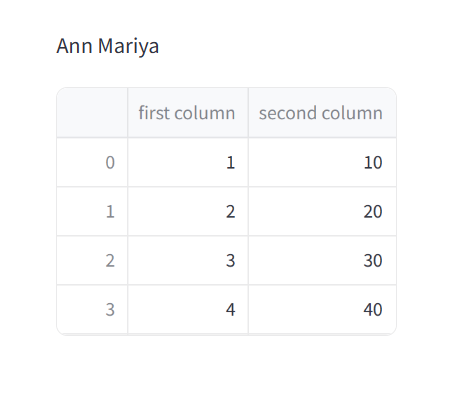


**4. What is the purpose of the st.write() function in Streamlit?**

It helps to display data, markdown, and text in the app

import streamlit as st  
st.write("Ann Mariya")  
st.write(pd.DataFrame({'first column':[1,2,3,4],'second column':[10,20,30,40]}))

output:



**5. How do you create interactive widgets in Streamlit? Provide examples.**

**interactive widgets**

st.button()-creates a button

st.checkbox()-creates a checkbox

st.radio(): Displays radio buttons.

st.slider(): Allows sliding selection.

st.selectbox(): displays a dropdown selection

st.multiselect():can make multiple selections

import streamlit as st  
if st.button('select class '):  
 st.write('class is selected')  
  
option = st.checkbox('Show/hide details')  
if option:  
 st.write('Details are visible now!')  
  
age = st.slider('Select your age', 1, 100, 25)  
st.write(f'Your age is {age}')  
  
option\_list = ['English', 'Computer Science', 'Maths']  
selected\_option = st.selectbox('Choose an option', option\_list)  
  
radio\_selected = st.radio('Choose one option', option\_list)  
  
multiselect\_selected = st.multiselect('Choose multiple options', option\_list)  
  
st.write(f'Radio selected: {radio\_selected}')  
st.write(f'Multiselect selected: {multiselect\_selected}')

