Building Dashboards with Streamlit

Learn how to create interactive data dashboards using Python and Streamlit with code examples

By Ayoolumi Melehon

ayofemimelehon.com

1. What is Streamlit?

Streamlit is an open-source Python library that makes it incredibly easy to create and share beautiful, interactive web applications for data science and machine learning.

Why Streamlit?

* Pure Python - No web development experience needed

* Fast development - Build apps in hours, not days

* Interactive widgets - Sliders, buttons, dropdowns built-in

* Live reloading - See changes instantly as you code

* Free hosting - Deploy to Streamlit Cloud for free

Installation

```
pip install streamlit
```

Your First App

```
import streamlit as st

st.title("Hello, Streamlit!")
st.write("This is my first Streamlit app.")
```

Run with:

```
streamlit run app.py
```

2. Basic Elements

Text Elements

```
st.title("Main Title")
st.header("Section Header")
st.subheader("Subsection")
st.text("Fixed-width text")
st.write("Markdown and data")
st.markdown("**Bold** and *italic*")
st.caption("Small caption text")
```

Data Display

```
import pandas as pd

df = pd.DataFrame({'A': [1, 2, 3], 'B': [4, 5, 6]})

st.dataframe(df) # Interactive table
st.table(df) # Static table
st.json({'key': 'value'}) # JSON display
st.metric("Revenue", "$1.2M", "+12%") # KPI metric
```

Metrics and KPIs

```
col1, col2, col3 = st.columns(3)
col1.metric("Temperature", "70F", "1.2F")
col2.metric("Wind", "9 mph", "-8%")
col3.metric("Humidity", "86%", "4%")
```

3. Interactive Widgets

Input Widgets

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

# Text input
name = st.text_input("Enter your name")

# Number input
age = st.number_input("Age", min_value=0, max_value=120)

# Slider
value = st.slider("Select a value", 0, 100, 50)

# Select box (dropdown)
option = st.selectbox("Choose", ["A", "B", "C"])

# Multi-select
options = st.multiselect("Select multiple", ["X", "Y", "Z"])

# Checkbox
if st.checkbox("Show details"):
st.write("Here are the details...")

# Date picker
date = st.date_input("Select date")
```

File Upload

```
uploaded_file = st.file_uploader("Upload CSV", type="csv")
if uploaded_file is not None:
df = pd.read_csv(uploaded_file)
st.dataframe(df)
```

4. Charts and Visualizations

Built-in Charts

```
import numpy as np

data = pd.DataFrame(np.random.randn(20, 3), columns=['A', 'B', 'C'])

st.line_chart(data) # Line chart
st.area_chart(data) # Area chart
st.bar_chart(data) # Bar chart
```

Matplotlib Integration

```
import matplotlib.pyplot as plt

fig, ax = plt.subplots()
ax.plot([1, 2, 3, 4], [1, 4, 2, 3])
ax.set_title("My Plot")
st.pyplot(fig)
```

Plotly Integration

```
import plotly.express as px

fig = px.scatter(df, x="col1", y="col2", color="category")
st.plotly_chart(fig, use_container_width=True)
```

5. Layout and Organization

Sidebar

```
st.sidebar.title("Settings")
option = st.sidebar.selectbox("Select", ["A", "B"])
value = st.sidebar.slider("Value", 0, 100)
```

Columns

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

with col1:
st.header("Column 1")
st.write("Content here")

with col2:
st.header("Column 2")
st.write("More content")
```

Tabs

```
tab1, tab2, tab3 = st.tabs(["Data", "Charts", "Settings"])

with tab1:
st.dataframe(df)

with tab2:
st.line_chart(df)

with tab3:
st.write("Settings go here")
```

Expanders

```
with st.expander("Click to expand"):
st.write("Hidden content here")
st.image("image.png")
```

Containers

```
container = st.container()
container.write("This appears first")
st.write("This appears second")
container.write("This also appears first")
```

6. Caching for Performance

Streamlit reruns your entire script on every interaction. Caching prevents expensive operations from running repeatedly.

```
@st.cache_data # For data
def load_data():
df = pd.read_csv("large_file.csv")
return df

@st.cache_resource # For models/connections
def load_model():
model = load_my_ml_model()
return model

# Data only loads once, then cached
df = load_data()
```

Tip: Use @st.cache_data for DataFrames and data. Use @st.cache_resource for ML models and database connections.

7. Complete Dashboard Example

```
import streamlit as st
import pandas as pd
import plotly.express as px

# Page config
st.set_page_config(page_title="Sales Dashboard", layout="wide")

# Title
st.title("Sales Dashboard")

# Sidebar filters
st.sidebar.header("Filters")
year = st.sidebar.selectbox("Year", [2024, 2025, 2026])
region = st.sidebar.multiselect("Region", ["North", "South", "East", "West"])

# Load and filter data
@st.cache_data
def load_data():
return pd.read_csv("sales_data.csv")

df = load_data()
filtered_df = df[(df['Year'] == year)]
if region:
filtered_df = filtered_df[filtered_df['Region'].isin(region)]

# KPI metrics
col1, col2, col3, col4 = st.columns(4)
col1.metric("Total Sales", f"${filtered_df['Sales'].sum():,.0f}")
col2.metric("Orders", f"{len(filtered_df):,}")
col3.metric("Avg Order", f"${filtered_df['Sales'].mean():,.0f}")
col4.metric("Customers", f"{filtered_df['Customer'].nunique():,}")

# Charts
col1, col2 = st.columns(2)

with col1:
st.subheader("Sales by Region")
fig = px.bar(filtered_df.groupby('Region')['Sales'].sum().reset_index(),
x='Region', y='Sales')
st.plotly_chart(fig, use_container_width=True)

with col2:
st.subheader("Monthly Trend")
fig = px.line(filtered_df.groupby('Month')['Sales'].sum().reset_index(),
x='Month', y='Sales')
st.plotly_chart(fig, use_container_width=True)

# Data table
st.subheader("Raw Data")
st.dataframe(filtered_df)
```

8. Deployment to Streamlit Cloud

Step 1: Prepare Your Repository

Your GitHub repo needs:

* app.py - Your main Streamlit app

* requirements.txt - Python dependencies

requirements.txt example:

```
streamlit==1.32.0
pandas==2.0.0
plotly==5.18.0
```

```
numpy==1.24.0
```

Step 2: Deploy

1. Go to share.streamlit.io
2. Sign in with GitHub
3. Click 'New app'
4. Select your repository and branch
5. Click 'Deploy'

Your app will be live at: your-app-name.streamlit.app

Resources

* Official Docs: docs.streamlit.io

* Gallery: streamlit.io/gallery

* Community: discuss.streamlit.io

Created by Ayoolumi Melehon | ayofemimelehon.com
2026 | Free to use and share