Streamlit Tutoriala

Setup: Install and run Streamlit

Install

In your terminal / command prompt:

```
pip install streamlit
```
on mac:

```
pip3 install streamlit
```

Create and run your first app

Create a file, e.g. `app.py`:

```
import streamlit as st

st.title("Hello, Streamlit! 👋")
st.write("This is my very first Streamlit app.")
```
Run it:

```
streamlit run app.py
```
Your browser should open at `http://localhost:8501`.

Official docs:

- Get started guide (docs.streamlit.io)
- API reference overview (docs.streamlit.io)

---

1. Basic page elements (text, data, images)

Replace the contents of `app.py` with this and play around:

```
import streamlit as st
import pandas as pd

st.title("1. Basic Page Elements")
st.header("Headers and text")
st.subheader("This is a subheader")
st.caption("Small, grey caption text")

st.write("`st.write` is very flexible — you can pass strings,
numbers, dataframes, etc.")
st.text("Plain fixed-width text for code-like things.")
st.markdown("You can use **Markdown** here, including *italic*
and `code`.")
```

```
st.divider()

st.header("Display data")
df = pd.DataFrame({
    "name": ["Alice", "Bob", "Charlie"],
    "age": [25, 32, 29]
})
st.write("Here is a small dataframe:")
st.dataframe(df)   # scrollable, sortable table

st.divider()

st.header("Images")
st.write("You can show images from a URL or local file.")
st.image(
    "https://streamlit.io/images/brand/streamlit-logo-secondary-
colormark-darktext.png",
    caption="Streamlit logo",
)
```

Key functions used:

- `st.title`, `st.header`, `st.subheader`, `st.caption`, `st.text`, `st.markdown`, `st.write`
- `st.dataframe`
- `st.image`

Official docs:

- Text elements (text, markdown, title, etc.) (docs.streamlit.io)
- Data display elements (`st.dataframe`, etc.) (docs.streamlit.io)

---

2. Basic widgets (interactivity)

Now let's add user input. Replace the file (or create a new one like `widgets_demo.py`):

```
import streamlit as st

st.title("2. Basic Widgets Demo")

st.header("Buttons & text inputs")
name = st.text_input("What is your name?")
if st.button("Say hello"):
    if name:
        st.success(f"Hello, {name} 👋")
    else:
        st.warning("Please enter your name first.")

st.divider()

st.header("Numeric inputs")
```

```
age = st.number_input("Your age", min_value=0, max_value=120,
value=25)
st.write("You entered age:", age)

st.slider("Pick a value", min_value=0, max_value=100, value=50,
key="slider_example")

st.divider()

st.header("Selection widgets")
color = st.selectbox("Favourite colour", ["Red", "Green",
"Blue"])
options = st.multiselect("Choose some fruits", ["Apple",
"Banana", "Orange", "Grapes"])
st.write("Colour:", color)
st.write("Fruits:", options)

st.divider()

st.header("Booleans and dates")
agree = st.checkbox("I agree to the terms")
if agree:
    st.info("Thanks for agreeing. ")

date = st.date_input("Pick a date")
st.write("Chosen date:", date)
```

Common widgets you've just used:

- `st.button`, `st.text_input`, `st.number_input`, `st.slider`
- `st.selectbox`, `st.multiselect`
- `st.checkbox`, `st.date_input`

Official docs:

- Input widgets reference (docs.streamlit.io)

---

3. Basic charts and graphs

Let's visualise some data using built-in chart functions.

Create `charts_demo.py`:

```
import streamlit as st
import pandas as pd
import numpy as np

st.title("3. Charts Demo")

st.header("Line, area, and bar charts")

# Fake time series data
data = pd.DataFrame(
    np.random.randn(20, 3),
```

```
    columns=["A", "B", "C"]
)

st.subheader("Line chart")
st.line_chart(data)

st.subheader("Area chart")
st.area_chart(data)

st.subheader("Bar chart")
st.bar_chart(data)

st.divider()

st.header("Scatter chart & map")

# Scatter chart
scatter_data = pd.DataFrame(
    np.random.randn(100, 3),
    columns=["x", "y", "size"]
)
st.subheader("Scatter chart")
st.scatter_chart(scatter_data, x="x", y="y", size="size")

# Map (requires lat/lon columns)
st.subheader("Map")
map_data = pd.DataFrame({
    "lat": 51.5 + np.random.randn(100) * 0.01,
    "lon": -0.12 + np.random.randn(100) * 0.01,
})
st.map(map_data)
```

If you want more control, you can use libraries like Altair, Matplotlib, or Plotly:

```
import altair as alt

st.header("Altair example")

chart = (
    alt.Chart(data.reset_index())
    .mark_line()
    .encode(
        x="index",
        y="A",
    )
)
st.altair_chart(chart, use_container_width=True)
```

Official docs:

- Chart elements (line/area/bar/scatter/map) (docs.streamlit.io)
- Plotly in Streamlit (`st.plotly_chart`) (docs.streamlit.io)

---

4. Layout: sidebars, columns, and sections

Now we'll control the layout so it looks like a dashboard.

Create `layout_demo.py`:

```
import streamlit as st
import pandas as pd
import numpy as np

st.set_page_config(page_title="Layout Demo", layout="wide")

st.title("4️⃣ Layout Demo — Building a Simple Dashboard")

# ---- SIDEBAR ----
with st.sidebar:
    st.header("Controls")
    n_points = st.slider("Number of points", 10, 500, 100)
    show_table = st.checkbox("Show raw data")

# ---- MAIN CONTENT ----
st.header("Main Content")

# Create some random data
data = pd.DataFrame(
    np.random.randn(n_points, 3),
    columns=["Feature 1", "Feature 2", "Feature 3"]
)

# Use columns to split space
col1, col2 = st.columns(2)

with col1:
    st.subheader("Line chart")
    st.line_chart(data)

with col2:
    st.subheader("Bar chart")
    st.bar_chart(data)

# Optional table in an expander
with st.expander("See raw data"):
    if show_table:
        st.dataframe(data)
    else:
        st.info("Tick 'Show raw data' in the sidebar to see the
table.")
```

Key layout tools:

- `st.set_page_config(layout="wide")`
- `st.sidebar` for a sidebar
- `st.columns()` for side-by-side sections
- `st.expander()` for collapsible sections
- `st.divider()` to visually separate blocks

Official docs:

- Layout and containers (columns, tabs, sidebar, expanders, containers) (docs.streamlit.io)

---

5. Putting it together: a tiny single-page "dashboard"

Now combine widgets + charts + layout into a simple but real-ish dashboard.

Create `mini_dashboard.py`:

```
import streamlit as st
import pandas as pd
import numpy as np

st.set_page_config(page_title="Mini Dashboard", layout="wide")

st.title("📊 Mini Sales Dashboard")

# ---------- Sidebar filters ----------
with st.sidebar:
    st.header("Filters")
    year = st.selectbox("Year", [2023, 2024, 2025])
    min_revenue = st.slider("Min revenue", 0, 100000, 20000,
step=5000)

# ---------- Fake data ----------
np.random.seed(42)
df = pd.DataFrame({
    "year": np.random.choice([2023, 2024, 2025], size=200),
    "region": np.random.choice(["Europe", "Asia", "Americas"],
size=200),
    "revenue": np.random.randint(5_000, 100_000, size=200),
})

# Apply filters
filtered = df[(df["year"] == year) & (df["revenue"] >=
min_revenue)]

st.caption(f"Showing {len(filtered)} rows for year {year} with
revenue ≥ {min_revenue}")

# ---------- Layout ----------
col1, col2 = st.columns(2)

with col1:
    st.subheader("Revenue by Region")
    rev_by_region = filtered.groupby("region")["revenue"].sum()
    st.bar_chart(rev_by_region)

with col2:
    st.subheader("Revenue distribution")
    st.area_chart(filtered["revenue"])
```

```
with st.expander("See filtered data"):
    st.dataframe(filtered)
```

This is essentially a **simple dashboard**:

- Sidebar controls input (year + threshold)
- Main page shows charts
- Expander reveals details

In [ ]: