2547218_CIA1

CODE:

```
import streamlit as st
import pandas as pd
import geopandas as gpd
import matplotlib.pyplot as plt

# ---------------- PAGE CONFIG ----------------
st.set_page_config(
    page_title="Silver Analytics Dashboard",

    layout="wide"
)

# ---------------- LOAD DATA ----------------
silver_sales = pd.read_csv("state_wise_silver_purchased_kg.csv")
price_history = pd.read_csv("historical_silver_price.csv")

# ---------------- SIDEBAR ----------------
st.sidebar.title("Silver Analytics")
page = st.sidebar.radio(
    "Navigate",
    ["Price Calculator", "Sales Insights", "Geo Analysis"]
)

st.sidebar.markdown("---")
st.sidebar.caption("CIA-1 | Streamlit + GeoPandas")

# ====================== PAGE 1 ======================
if page == "Price Calculator":
```

```
st.title("Silver Price Calculator")

st.write("Interactively calculate silver value and analyze historical trends.")

col1, col2, col3 = st.columns(3)

with col1:
    weight = st.number_input("Silver Weight", min_value=0.0, value=100.0)
    unit = st.selectbox("Unit", ["grams", "kilograms"])

with col2:
    price_per_gram = st.slider(
        "Price per gram (INR)",
        min_value=50,
        max_value=120,
        value=75
    )

with col3:
    currency = st.selectbox("Currency", ["INR", "USD"])

if unit == "kilograms":
    weight *= 1000

total_cost = weight * price_per_gram
usd_rate = 0.012

st.markdown("### Calculated Value")

c1, c2 = st.columns(2)
```

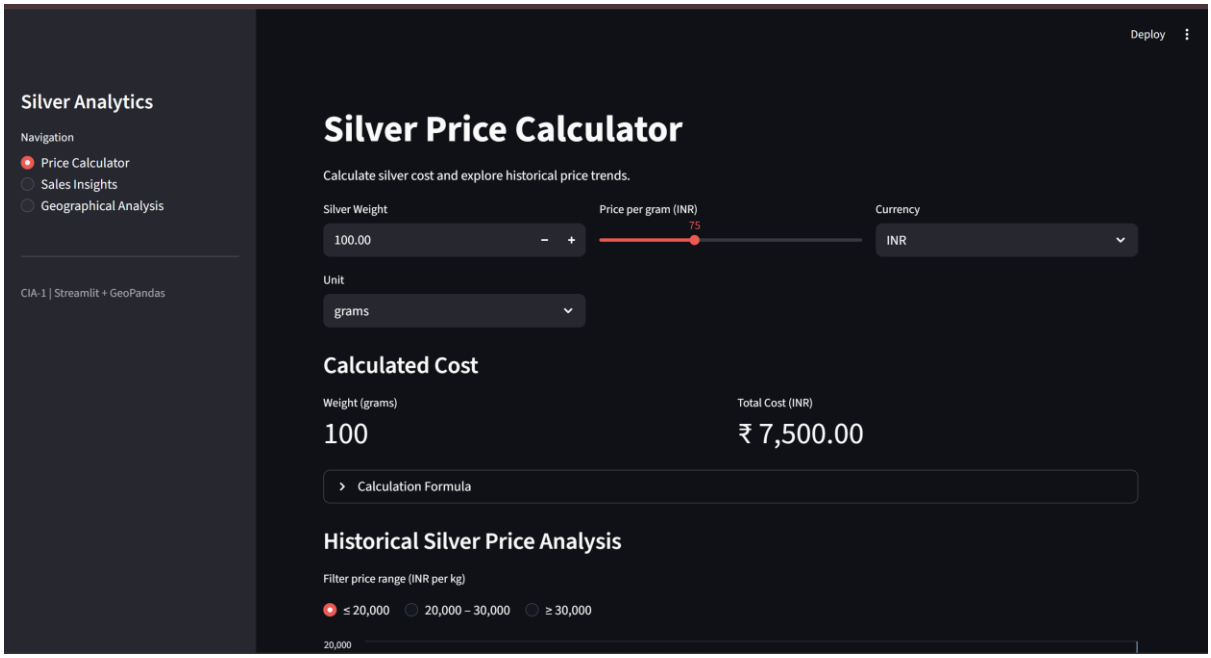
```
    c1.metric("Weight (grams)", f"{weight:,.0f}")
    c2.metric(
        f"Total Cost ({currency})",
        f"{total_cost * usd_rate:,.2f}" if currency == "USD" else f"₹{total_cost:,.2f}"
    )

    with st.expander("Calculation Logic"):
        st.code("Total Cost = Weight (grams) × Price per gram")

    # -------- Historical Price Filter --------
    st.subheader("Historical Silver Price Trend")

    price_filter = st.radio(
        "Filter by Silver Price (INR/kg)",
        ["≤ 20,000", "20,000 – 30,000", "≥ 30,000"],
        horizontal=True
    )

    if price_filter == "≤ 20,000":
        filtered = price_history[price_history["Silver_Price_INR_per_kg"] <= 20000]
    elif price_filter == "20,000 – 30,000":
        filtered = price_history[
            (price_history["Silver_Price_INR_per_kg"] > 20000) &
            (price_history["Silver_Price_INR_per_kg"] < 30000)
        ]
    else:
        filtered = price_history[price_history["Silver_Price_INR_per_kg"] >= 30000]
```

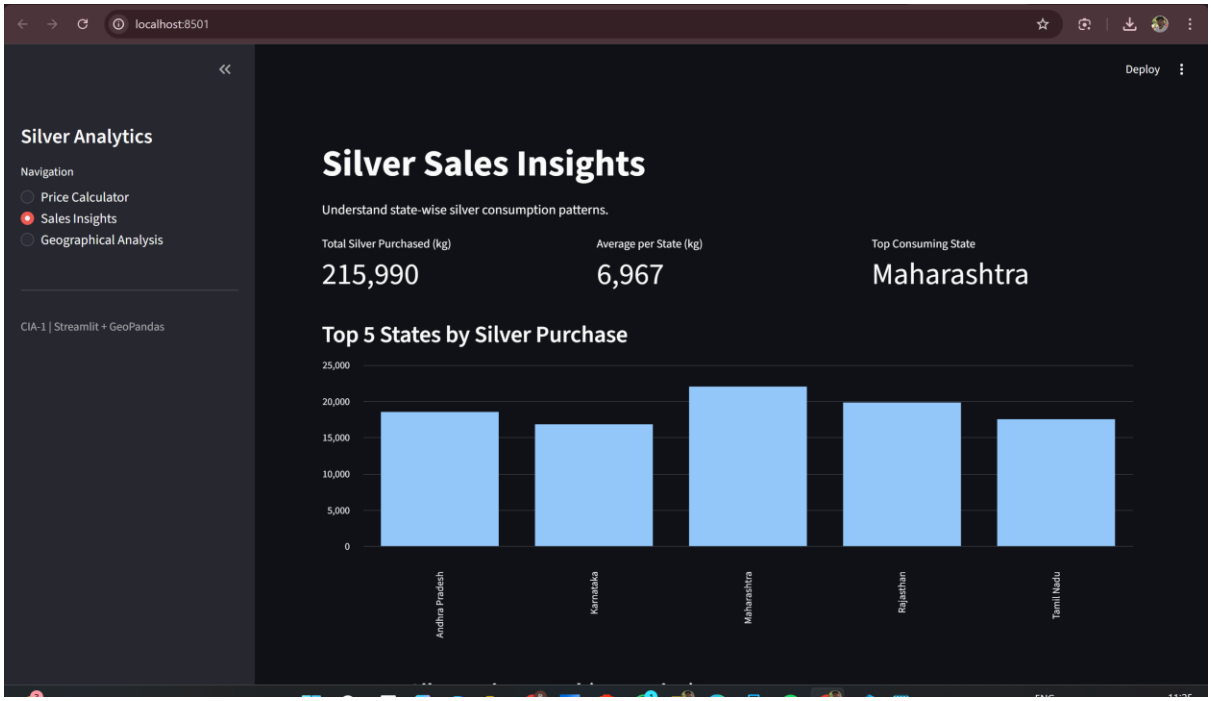
```
    st.line_chart(
        filtered.set_index("Year")["Silver_Price_INR_per_kg"]
    )

# ====================== PAGE 2 ======================
elif page == "Sales Insights":
    st.title("Silver Sales Insights")

    total = int(silver_sales["Silver_Purchased_kg"].sum())
    avg = int(silver_sales["Silver_Purchased_kg"].mean())
    max_state = silver_sales.loc[
        silver_sales["Silver_Purchased_kg"].idxmax(), "State"
    ]

    c1, c2, c3 = st.columns(3)
    c1.metric("Total Silver Purchased (kg)", f"{total:,}")
    c2.metric("Average per State (kg)", f"{avg:,}")
    c3.metric("Top Consuming State", max_state)

    st.subheader("Top 5 States by Silver Purchase")

    top_states = silver_sales.sort_values(
        by="Silver_Purchased_kg",
        ascending=False
    ).head(5)

    st.bar_chart(
        top_states.set_index("State")["Silver_Purchased_kg"]
    )
```

```
    # -------- January Trend (Corrected) --------
    st.subheader("January Silver Price Trend (Year-wise)")

    january_prices = price_history[price_history["Month"] == "Jan"]

    st.line_chart(
        january_prices.set_index("Year")["Silver_Price_INR_per_kg"]
    )

# ====================== PAGE 3 ======================
else:
    st.title("Geographical Silver Analysis")

    st.write("State-wise silver purchases visualized using GeoPandas.")

    india_states = gpd.read_file("India_State_Boundary.shp")

    with st.expander("Shapefile Columns"):
        st.write(india_states.columns)

    # Adjust column name if required
    merged = india_states.merge(
        silver_sales,
        left_on="STATE_NAME",   # change if your shapefile uses a different name
        right_on="State",
        how="left"
    )

    fig, ax = plt.subplots(figsize=(10, 10))
    merged.plot(
```

```
    column="Silver_Purchased_kg",
    cmap="Greys",
    linewidth=0.7,
    edgecolor="black",
    legend=True,
    ax=ax
)

ax.set_title("State-wise Silver Purchases in India (kg)")
ax.axis("off")

st.pyplot(fig)

st.caption("Darker regions indicate higher silver consumption.")
```

OUTPUT:




localhost:8501
Deploy
Silver Analytics
Navigation
Price Calculator
Sales Insights
Geographical Analysis
CIA-1 | Streamlit + GeoPandas
Silver Sales Insights
Understand state-wise silver consumption patterns.
Total Silver Purchased (kg)
215,990
Average per State (kg)
6,967
Top Consuming State
Maharashtra
Top 5 States by Silver Purchase
25,000
20,000
15,000
10,000
5,000
0
Andhra Pradesh
Karnataka
Maharashtra
Rajasthan
Tamil Nadu