In [1]:

```
%%writefile app.py
import streamlit as st
from datetime import datetime
import pandas as pd

#must define this here so joblib unpickling finds it in __main__
def extract_date_features(df_in):
    df2 = df_in.copy()
    dates = pd.to_datetime(df2["date"], errors="coerce")
    df2["dt_year"] = dates.dt.year.fillna(0).astype(int)
    df2["dt_month"] = dates.dt.month.fillna(0).astype(int)
    df2["dt_day"] = dates.dt.day.fillna(0).astype(int)
    df2["dt_dow"] = dates.dt.dayofweek.fillna(0).astype(int)
    return df2[["dt_year","dt_month","dt_day","dt_dow"]]


from predict import predict_category

# PAGE HEADER

st.title("AI-Based Financial Transaction Categorisation Demo")
st.markdown("Enter transaction details below to get the predicted category.")


#TRANSACTION NOTES (multiple examples inside placeholder)


notes = st.text_input(
    "Transaction Notes",
    value="",
    placeholder="e.g., uber to office / grocery shopping / electricity bill / dinner at restaurant / monthly rent"
)


# 2) LOCATION (multiple examples inside placeholder)


location = st.text_input(
    "Location",
    value="",
    placeholder="e.g., Mumbai / Chennai / Bengaluru / Hyderabad / Delhi"
)


# 3) PAYMENT MODE – CLEANED FINAL LIST


payment_mode = st.selectbox(
    "Payment Mode",
    [
        "Select Payment Mode",
        "upi",
        "bank transfer",
        "card",
        "cash",
        "other"
    ],
    index=0
)



# 4) TRANSACTION TYPE


transaction_type = st.selectbox(
    "Transaction Type",
    ["Select Transaction Type", "expense", "income"],
    index=0
)


# 5) DATE INPUT


date = st.date_input("Date", value=None)
date_str = date.strftime("%Y-%m-%d") if date else None



# 6) AMOUNT


amount = st.number_input(
    "Amount",
    min_value=0.0,
    value=None,
    placeholder="Enter amount"
)


# 7) PREDICT CATEGORY


if st.button("Predict Category"):

    # VALIDATION
    if not notes.strip():
        st.warning("Please enter transaction notes.")
    elif not location.strip():
        st.warning("Please enter a location.")
    elif amount is None:
        st.warning("Please enter a valid amount.")
    elif payment_mode == "Select Payment Mode":
        st.warning("Please select a payment mode.")
    elif transaction_type == "Select Transaction Type":
        st.warning("Please select a transaction type.")
    elif not date_str:
        st.warning("Please select a date.")
    else:
        pred, top3 = predict_category(
            notes,
            amount,
            payment_mode,
            location,
            transaction_type,
            date_str
        )

        st.success(f"Predicted Category: **{pred}**")

        st.subheader("Top 3 Predictions:")
        for cat, prob in top3:
            st.write(f"- **{cat}** → {prob * 100:.2f}%")
```

```
Overwriting app.py
```