BUSINESS MANEGMENT SOWTWARE

```
import streamlit as st
from streamlit_option_menu import option_menu
from streamlit_extras.dataframe_explorer import dataframe_explorer
import pandas as pd
import mysql.connector as mc
from matplotlib import pyplot as plt
from PIL import Image
import subprocess
```

```
st.set_page_config(
page_title="BUSINESS MANEGMENT",
page_icon=":boat:",
layout="wide",
)
```

```
def login_page():
    st.markdown('<style>div.block-container{padding-top:2rem}</style>',unsafe_allow_html=True)
    A,B=st.columns(2)
    with B:
        a,b=st.columns(2)
        with a:
            st.subheader("WELCOME")
            st.image("BOAT3.jpg")
            username=st.text_input("USERNAME")
            password=st.text_input("PASSWORD",type="password")
            LOGIN_BUTTON=st.button("LOGIN")

    with A:
        a,b=st.columns(2)
        with b:
            i=Image.open("login.jpg")
            ni=i.resize((400,550))
            st.image(ni,use_column_width=True)

            if LOGIN_BUTTON:
                if username=="maloypariari" and password=="20100172":
                    st.success("SUCCESSFUL")
                    subprocess.Popen(["streamlit","run","MAIN.py"]).wait()
                else:
                    st.error("CHECK DETAILS AGAIN")
```

```
def main_page():
    st.markdown('<style>div.block-container{padding-top:2rem}</style>',unsafe_allow_html=True)
    colors=["#7380ec","#ff7782","#41f1b6","#ffbb55","#111e88","#353949"]


    mycon=mc.connect(host="localhost",user="root",password="MPMSSP3639",database="BUISNESS")
    mycur=mycon.cursor()

    m0="CREATE DATABASE IF NOT EXISTS BUISNESS"
    m1="CREATE TABLE IF NOT EXISTS INVENTORY(PRODUCT_ID INT(11) PRIMARY KEY,NAME VARCHAR(22),
        PRICE INT(11),QUANTITY INT(11))"
    m2="CREATE TABLE IF NOT EXISTS SALES(DATE DATE,VALUE INT(11),PROFIT INT(11),LOSS INT(11),GROWTH FLOAT(7,2)
    m3="CREATE TABLE IF NOT EXISTS EMPLOYEES(EMPLOYEE_ID INT(11) PRIMARY KEY,
        NAME VARCHAR(22),JOB VARCHAR(22),HIREDATE DATE,SALARY INT(11),COMMISSION INT(11),DEPT_NO INT(11))"
    m4="CREATE TABLE IF NOT EXISTS PRODUCTS(PRODUCT_ID INT(11) PRIMARY KEY,NAME VARCHAR(22),START_DATE DATE)"
    m5="CREATE TABLE IF NOT EXISTS ORDERS(ORDER_NO INT(11) PRIMARY KEY,CUSTOMER_NAME VARCHAR(22),
        PRODUCTS VARCHAR(22),AMOUNT INT(11),DATE DATE)"
    m6="CREATE TABLE IF NOT EXISTS ACCOUNTS(DATE DATE,CREDIT INT(11),DEBIT INT(11),BALANCE FLOAT(7,2))"

    mycur.execute(m0)
    mycur.execute(m1)
    mycur.execute(m2)
    mycur.execute(m3)
    mycur.execute(m4)
    mycur.execute(m5)
    mycur.execute(m6)

    with st.sidebar:
        a=st.sidebar.image("BOAT3.jpg",caption="PLUG INTO NIRWANA")

        SELECTED = option_menu(
        menu_title=None,
        options=["HOME","INVENTORY","SALES","EMPLOYEES","PRODUCTS","ORDERS","ACCOUNTS","LOGOUT"],
        styles={"container": {"padding": "5!important", "background-color": "#f6f6f9","position":"relative"},
            "icon": {"color": "black", "font-size": "23px"},
            "nav-link": {"font-size": "16px", "text-align": "left", "margin":"0px", "--hover-color": "#7380ec"
            "nav-link-selected": {"background-color": "#7d8da1"}},
        icons=["house-fill","box-seam","bar-chart","person-bounding-box","receipt-cutoff","cart","bank","arrow
        )
#
#
    if SELECTED=="HOME":
        col1,col2,col3=st.columns(3)

        with col1:
            sql="UPDATE ACCOUNTS SET BALANCE=CREDIT-DEBIT"
            mycur.execute(sql)
            DATA2=pd.read_sql("SELECT SUM(BALANCE) AS 'ACCOUNT BALANCE   ' FROM ACCOUNTS",con=mycon)
            st.info(DATA2)

            st.write("ACCOUNT HISTORY")
            sql="SELECT * FROM ACCOUNTS"
            DATA=pd.read_sql(sql,columns=["CREDIT","DEBIT","BALANCE"],con=mycon)
            st.area_chart(DATA,x="DATE",height=270,use_container_width=True)

            st.write("ORDERS")
            sql="SELECT CUSTOMER_NAME,AMOUNT FROM ORDERS"
            DATA=pd.read_sql(sql,con=mycon)
            with st.expander(label="RECENT"):
                st.table(DATA)

        with col2:
            sql="SELECT SUM(VALUE) AS 'COMPANY VALUATION   ' FROM SALES"
            DATA=pd.read_sql(sql,con=mycon)
            st.warning(DATA)

            st.write("SALES")
            sql="SELECT PROFIT,LOSS,DATE FROM SALES"
            DATA=pd.read_sql(sql,columns=["PROFIT","LOSS"],con=mycon)
            st.bar_chart(DATA,x="DATE",height=270,use_container_width=True)

            st.write("PRODUCTS")
            sql="SELECT NAME,START_DATE FROM PRODUCTS"
            DATA=pd.read_sql(sql,columns=["NAME","START_DATE"],con=mycon)
            with st.expander(label="AVAILABLE"):
                st.table(DATA)

        with col3:
            c1,c2=st.columns(2)
            with c2:
                st.time_input("⏰ TIME")
            with c1:
                st.date_input("📅 DATE")

            sql="SELECT JOB,COUNT(EMPLOYEE_ID) AS 'NJOB' FROM EMPLOYEES GROUP BY JOB"
            DATA=pd.read_sql(sql,con=mycon)
            fig = plt.figure()
            plt.pie(DATA["NJOB"],colors=colors,labels=DATA["NJOB"])
            plt.legend(title="__ JOBS __",labels=DATA["JOB"])
            plt.title("EMPLOYEES")
            st.pyplot(fig,clear_figure=True)
            st.pyplot(fig,clear_figure=True)

            i=Image.open("home.gif")
            ni=i.resize((450,290))
            st.image(ni,use_column_width=True)
#
#
    if SELECTED=="INVENTORY":
        st.subheader("MANAGE INVENTORY")
        T1,T2,T3,T4=st.tabs(["ADD ITEMS","DELETE ITEMS","UPDATE ITEMS","SHOW ITEMS"])

        with T1:
            II=st.number_input("PRODUCT ID NUMBER",step=1,format="%i",min_value=0)
            IN=st.text_input("PRODUCT NAME")
            IP=st.number_input("PRODUCT PRICE",step=1000,format="%i",min_value=0)
            IQ=st.slider("PRODUCT QUANTITY",step=10,format="%i",min_value=0)
            IN=IN.upper()

            sql="INSERT INTO INVENTORY VALUES({},'{}',{},{})".format(II,IN,IP,IQ)
            BUTTON=st.button(label="ADD")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="ADDED A NEW RECORD")
                else:
                    st.error("ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T2:
            R1,R2=st.columns(2)
            with R1:
                CONDATR=st.radio("CONDITIONAL ATRIBUTE",["PRODUCT_ID","NAME","PRICE"])
            with R2:
                CONDVAL=st.text_input("CONDITIONAL VALUE")

            sql=("DELETE FROM INVENTORY WHERE {}='{}'").format(CONDATR,CONDVAL)
            BUTTON=st.button(label="DELETE")
            st.image("delete.gif")

            if BUTTON:
                    mycur.execute(sql)
                    mycon.commit()
                    if mycur._check_executed()==None:
                        st.success(body="DELETED A RECORD")
                    else:
                        st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T3:
            R1,R2=st.columns(2)
            with R1:
                ATR=st.radio("ATTRIBUTE TO UPDATE",["PRICE","QUANTITY"])
                VAL=st.number_input("NEW VALUE",step=1,format="%i",min_value=0)
            with R2:
                CONDATR=st.radio("CONDITION ATTRIBUTE",["PRODUCT_ID","PRICE"])
                CONDVAL=st.number_input("CONDITION VALUE",step=1,format="%i",min_value=0)

            sql=("UPDATE INVENTORY SET {}={} WHERE {}={}").format(ATR,VAL,CONDATR,CONDVAL)
            BUTTON=st.button(label="UPDATE")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="UPDATED A RECORD")
                else:
```

```
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T4:
            DATA=pd.read_sql("SELECT * FROM INVENTORY",con=mycon)
            NDATA=dataframe_explorer(DATA)
            st.table(NDATA)
#
#
    if SELECTED=="SALES":
        st.subheader("MANAGE SALES")
        T1,T2,T3,T4=st.tabs(["ADD SALES","DELETE SALES","REVENUE","SHOW SALES"])

        with T1:
            SD=st.date_input("SALES DATE")
            SV=st.slider("SALES VALUE",step=100,format="%i",min_value=0,max_value=100000)
            SP=st.slider("SALES PROFIT",step=100,format="%i",min_value=0,max_value=100000)
            SL=st.slider("SALES LOSS",step=100,format="%i",min_value=0,max_value=100000)

            sql="INSERT INTO SALES(DATE,VALUE,PROFIT,LOSS) VALUES('{}',{},{},{})".format(SD,SV,SP,SL)
            BUTTON=st.button(label="ADD")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="ADDED A NEW RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T2:
            R1,R2=st.columns(2)
            with R1:
                CONDATR=st.radio("CONDITIONAL ATRIBUTE",["DATE"])
            with R2:
                CONDVAL=st.date_input("CONDITIONAL VALUE")

            sql=("DELETE FROM SALES WHERE {}='{}'").format(CONDATR,CONDVAL)
            BUTTON=st.button(label="DELETE")
            st.image("delete.gif")

            if BUTTON:
                    mycur.execute(sql)
                    mycon.commit()
                    if mycur._check_executed()==None:
                        st.success(body="DELETED A RECORD")
                    else:
                        st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T3:
            R1,R2=st.columns(2)
            with R1:
                SDR1=st.date_input("FROM DATE")
            with R2:
                SDR2=st.date_input("TO DATE")

            sql="SELECT SUM(PROFIT)-SUM(LOSS) FROM SALES WHERE DATE BETWEEN '{}' AND '{}'".format(SDR1,SDR2)
            BUTTON=st.button(label="FIND REVENUE")

            if BUTTON:
                mycur.execute(sql)
                for x in mycur.fetchall():
                    for a in x:
                        if a>=0:
                            st.success(a)
                        else:
                            st.error(a)

        with T4:
            sql="SELECT DATE,PROFIT,LOSS,(PROFIT/VALUE)*100 AS 'GROWTH' FROM SALES"
            DATA=pd.read_sql(sql,con=mycon)
            NDATA=dataframe_explorer(DATA)
            st.table(NDATA)
#
#
    if SELECTED=="EMPLOYEES":
        st.subheader("MANAGE EMPLOYEES")
        T1,T2,T3,T4=st.tabs(["ADD EMPLOYEES","DELETE EMPLOYEES","UPDATE EMPLOYEES","SHOW EMPLOYEES"])

        with T1:
            EI=st.number_input("EMPLOYEE ID",step=1,format="%i",min_value=0)
            EN=st.text_input("EMPLOYEE NAME")
            EJ=st.selectbox("EMPLOYEE JOB",["MANAGER","HR","SALES MAN","ANALYST","PRESIDENT","CLERK"])
            EHD=st.date_input("EMPLOYEE HIREDATE")
            ES=st.number_input("EMPLOYEE SALARY",step=1000,format="%i",min_value=0)
            EC=st.number_input("EMPLOYEE COMMISION",step=1000,format="%i",min_value=0)
            EDN=st.slider("EMPLOYEE DEPT_NO",step=1,format="%i",min_value=0,max_value=10)
            EN,EJ=EN.upper(),EJ.upper()

            sql="INSERT INTO EMPLOYEES VALUE({},'{}','{}','{}',{},{},{})".format(EI,EN,EJ,EHD,ES,EC,EDN)
            BUTTON=st.button(label="ADD")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="ADDED A NEW RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T2:
            R1,R2=st.columns(2)
            with R1:
                CONDATR=st.radio("CONDITIONAL ATTRIBUTE",["EMPLOYEE_ID","JOB"])
            with R2:
                CONDVAL=st.text_input("CONDITIONAL VALUE TO DELETE ")
            CONDATR=CONDATR.upper()

            sql="DELETE FROM EMPLOYEES WHERE {}='{}'".format(CONDATR,CONDVAL)
            BUTTON=st.button(label="DELETE")
            st.image("delete.gif")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="DELETED A RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T3:
            R1,R2=st.columns(2)
            with R1:
                ATR=st.radio("ATTRIBUTE TO UPDATE",["DEPT_NO","SALARY"])
                VAL=st.text_input("VALUE TO UPDATE")
            with R2:
                CONDATR=st.radio("CONDITIONAL ATTRIBUTE",["EMPLOYEE_ID","NAME","JOB","DEPT_NO","SALARY"])
                CONDVAL=st.text_input("CONDITIONAL VALUE")
                CONDVAL=st.text_input("CONDITIONAL VALUE ")
            VAL,CONDVAL=VAL.upper(),CONDVAL.upper()

            sql="UPDATE EMPLOYEES SET {}='{}' WHERE {}='{}'".format(ATR,VAL,CONDATR,CONDVAL)
            BUTTON=st.button(label="UPDATE")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="UPDATED A RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T4:
            DATA=pd.read_sql("SELECT * FROM EMPLOYEES",con=mycon)
            NDATA=dataframe_explorer(DATA)
            st.table(NDATA)
#
#
    if SELECTED=="PRODUCTS":
        st.subheader("MANAGE PRODUCTS")
        T1,T2,T3,T4=st.tabs(["ADD PRODUCTS","DELETE PRODUCTS","DISTINCT PRODUCTS","SHOW PRODUCTS"])

        with T1:
            PID=st.number_input("PRODUCT ID ",step=1,format="%i",min_value=0)
            PN=st.text_input("PRODUCT NAME ")
            PSD=st.date_input("PRODUCT START DATE")
            PN=PN.upper()

            sql="INSERT INTO PRODUCTS VALUES({},'{}','{}')".format(PID,PN,PSD)
            BUTTON=st.button(label="ADD")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="ADDED A NEW RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T2:
            R1,R2=st.columns(2)
            with R1:
                CONDATR=st.radio("CONDITION ATTRIBUTE TO DELETE",["PRODUCT_ID","NAME"])
            with R2:
                CONDVAL=st.text_input("CONDITION VALUE TO DELETE ")
            CONDATR,CONDVAL=CONDATR.upper(),CONDVAL.upper()

            sql=("DELETE FROM PRODUCTS WHERE {}='{}'").format(CONDATR,CONDVAL)
            BUTTON=st.button(label="DELETE")
            st.image("delete.gif")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="DELETED A RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T3:
            sql="SELECT DISTINCT NAME FROM PRODUCTS"
            DATA=pd.read_sql(sql,con=mycon)
            st.table(data=DATA)

        with T4:
            DATA=pd.read_sql("SELECT * FROM PRODUCTS",con=mycon)
            NDATA=dataframe_explorer(DATA)
            st.table(NDATA)
 #
 #
    if SELECTED=="ORDERS":
        st.subheader("MANAGE ORDERS")
        T1,T2,T3,T4=st.tabs(["ADD ORDERS","DELETE ORDERS","ANALYS ORDERS","SHOW ORDERS"])

        with T1:
            OD=st.date_input("SALES DATE")
            OI=st.number_input("ORDER ID ",step=1000,format="%i",min_value=0)
            OCN=st.text_input("CUSTOMER NAME ")
            OP=st.selectbox("PRODUCTS PURCHASED",["A","B","C","D","E"])
            OA=st.slider("ORDER AMOUNT",step=1000,format="%i",min_value=100000)
            OCN=OCN.upper()

            sql="INSERT INTO ORDERS VALUES({},'{}','{}',{},'{}')".format(OI,OCN,OP,OA,OD)
            BUTTON=st.button(label="ADD")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="ADDED A NEW RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T2:
            R1,R2=st.columns(2)
            with R1:
                CONDATR=st.radio("CONDITION ATTRIBUTE",["ORDER_NO","CUSTOMER_NAME"])
            with R2:
                CONDVAL=st.text_input("CONDITION VALUE")
            CONDVAL=CONDVAL.upper()

            sql=("DELETE FROM ORDERS WHERE {}='{}'").format(CONDATR,CONDVAL)
            BUTTON=st.button(label="DELETE")
            st.image("delete.gif")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="DELETED A RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T3:
            DATA=pd.read_sql("SELECT PRODUCTS,SUM(AMOUNT) AS 'TOTAL AMOUNT' FROM ORDERS GROUP BY PRODUCTS",con
            NDATA=dataframe_explorer(DATA)
            st.table(NDATA)

        with T4:
            DATA=pd.read_sql("SELECT * FROM ORDERS",con=mycon)
            NDATA=dataframe_explorer(DATA)
            st.table(NDATA)
#
#
    if SELECTED=="ACCOUNTS":
        st.subheader("MANAGE ACCOUNTS")
        T1,T2,T3=st.tabs(["ADD TRANSACTION","DELETE TRANSACTION","BALANCE"])

        with T1:
            AD=st.date_input("TRANSACTION DATE ")
            R1,R2=st.columns(2)
            with R1:
                ACR=st.slider("CREDIT AMOUNT ",step=1000,format="%i",min_value=0,max_value=100000)
            with R2:
                ADR=st.slider("DEBIT AMOUNT ",step=1000,format="%i",min_value=0,max_value=100000)

            sql="INSERT INTO ACCOUNTS(DATE,CREDIT,DEBIT) VALUES('{}',{},{})".format(AD,ACR,ADR)
            BUTTON=st.button(label="ADD")

            if BUTTON:
                mycur.execute(sql)
                mycon.commit()
                if mycur._check_executed()==None:
                    st.success(body="ADDED TRANSACTION RECORD")
                else:
                    st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T2:
            R1,R2=st.columns(2)
            with R1:
                CONDATR=st.radio("CONDITIONAL ATRIBUTE",["DATE"])
            with R2:
                CONDVAL=st.date_input("CONDITIONAL VALUE")

            sql=("DELETE FROM ACCOUNTS WHERE {}='{}'").format(CONDATR,CONDVAL)
            BUTTON=st.button(label="DELETE")
            st.image("delete.gif")

            if BUTTON:
                    mycur.execute(sql)
                    mycon.commit()
                    if mycur._check_executed()==None:
                        st.success(body="DELETED A RECORD")
                    else:
                        st.error(body="ERROR")
            else:
                st.info("TAKE YOUR TIME")

        with T3:
            sql="UPDATE ACCOUNTS SET BALANCE=CREDIT-DEBIT"
            mycur.execute(sql)
            DATA=pd.read_sql("SELECT * FROM ACCOUNTS",con=mycon)
            NDATA=dataframe_explorer(DATA)
            st.table(NDATA)
            DATA2=pd.read_sql("SELECT SUM(BALANCE) AS 'NET BALANCE : ' FROM ACCOUNTS",con=mycon)
            st.warning(DATA2)
#
#
    if SELECTED=="LOGOUT":
        st.subheader("LOGOUT SUCCESSFULLY THANK YOU")
        st.image("logout.gif")
        subprocess.Popen(["streamlit","run","LOGIN.py"]).wait()
        mycon.close()
```

```
if __name__=="__main__":
    login_page()
# USERNAME: maloypariari
# PASSWORD: 20100172
```