CODE – For managing the Book Records

```
import streamlit as st
from create import create
from database import create_table
from delete import delete
from read import read
from update import update

def main():
    st.title("Library Management System")
    menu = ["Add Books", "View Books", "Edit Books", "Remove Books"]
    choice = st.sidebar.radio("Menu", menu)

    create_table()
    if choice == "Add Books":
        st.subheader("Enter Book Details:")
        create()

    elif choice == "View Books":
        st.subheader("View created records")
        read()

    elif choice == "Edit Books":
        st.subheader("Update created records")
        update()

    elif choice == "Remove Books":
        st.subheader("Delete created records")
        delete()

    else:
        st.subheader("About tasks")


if __name__ == '__main__':
    main()
```

CODE – For managing the Student Membership

```
import streamlit as st
from createst import create
from databasest import create_table
from deletest import delete
from readst import read
from updatest import update

def main():
    st.title("Student Membership Details")
    menu = ["Add Member", "View all members", "Edit members", "Remove members"]
    choice = st.sidebar.radio("Here is the list of operations you can perform", menu)

    create_table()
    if choice == "Add Member":
        st.subheader("Enter student Details:")
```

```
        create()

    elif choice == "View all members":
        st.header("View added records")
        read()

    elif choice == "Edit members":
        st.subheader("Update added records")
        update()

    elif choice == "Remove members":
        st.subheader("Delete added records")
        delete()

    else:
        st.subheader("About tasks")


if __name__ == '__main__':
    main()
```

CODE – For generating Report Records

```
import streamlit as st
from createre import create
from databasere import create_table
from deletere import delete
from readre import read
from updatere import update

def main():
    st.title("Book Borrow Report")
    menu = ["Generate Report", "View all reports", "Extend date", "Delete report"]
    choice = st.sidebar.radio("Here is the list of operations you can perform", menu)

    create_table()
    if choice == "Generate Report":
        st.subheader("Enter report details:")
        create()

    elif choice == "View all reports":
        st.subheader("View present reports")
        read()

    elif choice == "Extend date":
        st.subheader("Extend date")
        update()

    elif choice == "Delete report":
        st.subheader("Book has been returned")
        delete()

    else:
```

```
    st.subheader("About tasks")


if __name__ == '__main__':
    main()
```

Create

```
import streamlit as st
from database import add_data


def create():
    col1, col2, col3, col4 = st.columns(4)
    with col1:
        num = st.text_input("Book No")
        author = st.text_input("Author")
    with col2:
        title = st.text_input("Book Title")
        publisher = st.text_input("Publisher")
    with col3:
        genre = st.text_input("Genre")
        availability = st.selectbox("Availability",["Yes","No","Not supplying
anymore"])
    with col4:
        des = st.text_input("mini description")
        copies = st.number_input("Number of copies")


    if st.button("Add this book to the library"):
        add_data(num,title,genre,des,author,publisher,availability,copies)
        st.success("Successfully added the book: {}".format(title))
```

Database

```
import mysql.connector

mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="Safiyasql@13",
    database="lib"
)
c = mydb.cursor()


def create_table():
    c.execute('CREATE TABLE IF NOT EXISTS reports(repnum INT, sname TEXT, bnum
INT, sid INT, issued DATE, returnd DATE)')


def add_data(repnum,sname,bnum,sid,issued,returnd):
    c.execute('INSERT INTO reports(repnum,sname,bnum,sid,issued,returnd)
VALUES (%s,%s,%s,%s,%s,%s)',(repnum,sname,bnum,sid,issued,returnd))
    mydb.commit()
```

```
def view_all_data():
    c.execute('SELECT * FROM reports')
    data = c.fetchall()
    return data


def view_only_repnum():
    c.execute('SELECT repnum FROM reports')
    data = c.fetchall()
    return data


def get_repnum(repnum):
    c.execute('SELECT * FROM reports WHERE repnum="{}"'.format(repnum))
    data = c.fetchall()
    return data


def
edit_data(nrepnum,nsname,nbnum,nsid,nissued,nreturnd,repnum,sname,bnum,sid,iss
ued,returnd):
    c.execute("UPDATE reports SET repnum=%s, sname=%s, bnum=%s, sid=%s,
issued=%s, returnd=%s  WHERE repnum=%s and sname=%s and bnum=%s and sid=%s and
issued=%s and returnd=%s ",
(nrepnum,nsname,nbnum,nsid,nissued,nreturnd,repnum,sname,bnum,sid,issued,retur
nd))
    mydb.commit()
    data = c.fetchall()
    return data


def delete_data(repnum):
    c.execute('SET FOREIGN_KEY_CHECKS=0')
    c.execute('DELETE FROM reports WHERE repnum="{}"'.format(repnum))
    mydb.commit()

import mysql.connector

mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="Safiyasql@13",
    database="lib"
)
c = mydb.cursor()


def create_table():
    c.execute('CREATE TABLE IF NOT EXISTS studs(sid INT, name TEXT, address
TEXT,phone INT, expiry DATE, mem TEXT)')


def add_data(sid,name,address,phone,expiry,mem):
```

```
    c.execute('INSERT INTO studs(sid,name,address,phone,expiry,mem) VALUES
(%s,%s,%s,%s,%s,%s)',(sid,name,address,phone,expiry,mem))
    mydb.commit()


def view_all_data():
    c.execute('SELECT * FROM studs')
    data = c.fetchall()
    return data


def view_only_sid():
    c.execute('SELECT sid FROM studs')
    data = c.fetchall()
    return data


def get_sid(sid):
    c.execute('SELECT * FROM studs WHERE sid="{}"'.format(sid))
    data = c.fetchall()
    return data


def
edit_stud_data(nsid,nname,naddress,nphone,nexpiry,nmem,sid,name,address,phone,
expiry,mem):
    c.execute("UPDATE studs SET sid=%s, name=%s, address=%s, phone=%s,
expiry=%s, mem=%s WHERE sid=%s and name=%s and address=%s and phone=%s and
expiry=%s and mem=%s ",
(nsid,nname,naddress,nphone,nexpiry,nmem,sid,name,address,phone,expiry,mem))
    mydb.commit()
    data = c.fetchall()
    return data


def delete_data(sid):
    c.execute('SET FOREIGN_KEY_CHECKS=0')
    c.execute('DELETE FROM studs WHERE sid="{}"'.format(sid))
    mydb.commit()
```

Update

```
import datetime
import pandas as pd
import streamlit as st
from databasest import view_all_data, view_only_sid, get_sid, edit_stud_data


def update():
    result = view_all_data()
    df = pd.DataFrame(result, columns=['Student ID Number','Student
Name','Address', 'Phone Number', 'Expiry date','Membership for a year'])
    with st.caption("Current Students"):
```

```
        st.dataframe(df)
    list_of_studs = [i[0] for i in view_only_sid()]
    selected_stud = st.selectbox("Student details to Edit", list_of_studs)
    selected_result = get_sid(selected_stud)
    if selected_result:
        sid = selected_result[0][0]
        name = selected_result[0][1]
        address = selected_result[0][2]
        phone = selected_result[0][3]
        expiry = selected_result[0][4]
        mem = selected_result[0][5]


    col1, col2, col3 = st.columns(3)
    with col1:
        sid = st.text_input("Student ID Number",sid)
        phone = st.number_input("Phone Number",phone)
    with col2:
        name = st.text_input("Student Name",name)
        expiry = st.date_input("Membership Expiry Date",expiry)
    with col3:
        address = st.text_input("Address",address)
        mem = st.selectbox("1 year membership",["Yes","No"])


        if st.button("Update Student Details"):
            edit_stud_data(nsid,nname,naddress,nphone,nexpiry,nmem,sid,name,ad
dress,phone,expiry,mem)
            st.success("Successfully updated details for student:: {} to
::{}".format(name,nname))

    result2 = view_all_data()
    df2 = pd.DataFrame(result2, columns=['Student ID Number','Student
Name','Address', 'Phone Number', 'Expiry date','Membership for a year'])
    with st.expander("Updated data"):
        st.dataframe(df2)
```

Read

```
import pandas as pd
import streamlit as st
from database import view_all_data


def read():
    result = view_all_data()
    df = pd.DataFrame(result, columns=['Book no','Book Title','Genre',
'Description', 'Author','Publisher', 'Availability', 'copies'])
    with st.caption("View all Book Records"):
        st.dataframe(df)
```