```
"""
Ryan Kennedy, Gabriel Walder
Cmdr. Schenk
Cloud Computing
7th Period
March 18, 2025
"""

import mysql.connector
import random

from author_record import AuthorRecord
from book_record import BookRecord

# CREATE DATABASE RyanKennedyAndGabrielWaldner;
# USE RyanKennedyAndGabrielWaldner;
# CREATE TABLE authors (
#     id integer auto_increment unique not null,
#     name text not null,
#     birth_year integer,
#     PRIMARY KEY (id)
# );
# CREATE TABLE books (
#     id integer auto_increment unique not null,
#     name text not null,
#     year_released integer,
#     page_amt integer,
#     price float,
#     author_id integer not null,
#     PRIMARY KEY (id),
#     FOREIGN KEY (author_id) REFERENCES authors(id)
# );

class Database():

    def __init__(self):
        # self.conn = mysql.connector.connect(host = "192.168.0.100", user = "student"
, passwd = "jchs", database = "RyanKennedyAndGabrielWaldner")
        self.conn = mysql.connector.connect(host = "127.0.0.1", user = "root", passwd
= "ryansmiles", database = "RyanKennedyAndGabrielWaldner")
        self.cursor = self.conn.cursor()

    def __del__(self):
        self.conn.commit()
        self.conn.close()

    def books_get_all_records(self):
        self.cursor.execute("SELECT * FROM books;")

        arr_data = self.cursor.fetchall()

        result = []

        if(len(arr_data) == 0):
            return result

        for record in arr_data:
            rec = BookRecord()
            rec.fill(record[0], record[1], record[2], record[3], record[4], record[5])
            result.append(rec)

        return result


    def authors_get_all_records(self):
        self.cursor.execute("SELECT * FROM authors;");

        arr_data = self.cursor.fetchall()
        result = []

        if(len(arr_data) == 0):
            return result

        for record in arr_data:
            rec = AuthorRecord()
            rec.fill(record[0], record[1], record[2])
            result.append(rec)


        return result


    def books_insert(self, record):
        self.cursor.execute("INSERT INTO books (name, year_released, page_amt, price, 
author_id) VALUES ('{}', {}, {}, {}, {});".format(record.name, record.year_released, r
ecord.page_amt, record.price, record.author_id))

    def authors_insert(self, record):
        self.cursor.execute("INSERT INTO authors (name, birth_year) VALUES ('{}', {});
".format(record.name, record.birth_year))

    def books_update(self, record):
        self.cursor.execute("UPDATE books SET name = '{}', year_released = {}, page_am
t = {}, price = {}, author_id ={} WHERE id = {};".format(record.name, record.year_rele
ased, record.page_amt, record.price, record.author_id, record.id))

    def authors_update(self, record):
        self.cursor.execute("UPDATE authors SET name = '{}', birth_year = {} WHERE id 
= {};".format(record.name, record.birth_year, record.id))

    def books_delete(self, id):
        self.cursor.execute("DELETE FROM books WHERE id = {};".format(id))

    def authors_delete(self, id):
        self.cursor.execute("DELETE FROM books WHERE author_id = {};".format(id))
        self.cursor.execute("DELETE FROM authors WHERE id = {};".format(id))
```