

وزارة التربية
منطقة الفروانية التعليمية
مدرسة جليب الشيوخ الثانوية بنين
العام الدراسي 2025 / 2026م



حلقة نقاشية حل أوراق عمل 11

تقديم أ/ محمد العديله و أ/ نادي ذكي

إشراف ورعاية:

د. محمد راشد المطيري

مدير المدرسة

أ. أنور محمد الكندري

الموجه الأول

أ. نواف عيد العازمي

الموجه الفني

أنواع البيانات Data Types في SQLite

تصنيفات تُحدد شكل وطبيعة المعلومات التي يمكن تخزينها في كل عمود من أعمدة الجدول. تساعد في تنظيم المعلومات والتأكد من التعامل معها بطريقة صحيحة عند التخزين أو المعالجة.

نوع البيانات	الوصف	مثال
INTEGER	تخزين الأرقام الصحيحة.	سنة النشر: 2025 - العمر: 17 - عدد الكتب: 1500 - عدد الطلاب: 20.
REAL	تخزين الأرقام العشرية.	سعر الكتاب: 3.15 - الوزن: 100.0 - المسافة: 0.75.
TEXT	تخزين النصوص.	اسم المؤلف: S. Ali. عنوان الكتاب: Learn SQLite.
BLOB	تخزين كائنات ثنائية كبيرة Binary Large Object.	الصور: JPEG, PNG, GIF. المستندات: PDF, Word, Excel. الملفات الصوتية / المرئية: MP3, MP4.
NULL	يستخدم لتمثيل القيم الفارغة (بدون قيمة).	عنوان الكتاب غير مُدخل - تاريخ طباعة الكتاب غير متوفر - البريد الإلكتروني فارغ



تضم دولة الكويت العديد من المعالم السياحية، التراثية، والدينية المُميزة، ونحتاج إلى حفظ معلوماتها في قاعدة بيانات رقمية، تتضمن أشهر تلك المعالم.

عائنا بيانات الجدول landmark بقاعدة البيانات Kuwait_landmarks.db ، وذلك باتباع الخطوات التالية:

1. شغل برنامج DB Browser for SQLite.

2. استدع قاعدة البيانات Kuwait_landmarks.db من مجلد أوراق العمل.

3. عاين هيكلية الجدول landmark :

◀ أسماء الأعمدة. :

.....

4. عاين بيانات الجدول landmark ، وتعرف على مكوناته ثم استكمل ما يلي:

◀ موقع "Al-Tahrir Tower" location :

◀ سنة التأسيس year "The Avenues mall" :

◀ عدد معالم دولة الكويت في الجدول :

◀ فئة category المَعلم "Souq Al-Mubarakiya" :

5. أغلق قاعدة البيانات Kuwait_landmarks.db

6. إنهاء البرنامج.

	id	name	location	category	year	price	info
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	Al-Tahrir Tower	Central Kuwait City	Architecture	1996	0.0	Al-Tahrir Tower is one of the most ...
2	2	Kuwait National Museum	Al-Murqab Area	Historical	1983	2.5	The museum houses cultural and ...
3	3	The Avenues Mall	Al-Rai Area	Shopping	2007	0.0	One of the largest and most popular...
4	4	Failaka Island	Off the eastern coast of Kuwait	Historical	1950	5.0	An island with archaeological sites...
5	5	Souq Al-Mubarakiya	Central Kuwait City	Traditional	1897	0.0	A traditional market offering a ...
6	6	Sheikh Jaber Al-Ahmad Bridge	Connecting Kuwait City to the south	Modern Infrastructure	2019	0.0	A remarkable modern bridge linking ...
7	7	Kuwait National Library	Al-Murqab Area	Cultural	1994	0.0	A prominent library housing an ...
8	8	Museum of Modern Art	Salmiya Area	Art	1980	1.0	Home to contemporary and modern art...
9	9	Sheikh Jaber Al-Ahmad Cultural ...	Kuwait City	Cultural	2016	0.0	It is a cultural center in Kuwait, ...
10	10	Grand Mosque of Kuwait	Kuwait City	Religious Architecture	1986	0.0	The Grand Mosque is the largest ...

1- شغل برنامج DB Browser (SQLite)

1- افتح قاعدة البيانات من زر فتح
Open Database

3- انتقل إلى تبويب استعراض البيانات
Browse Data

4- حدد اسم الجدول للمعاينة من قائمة جداول - Tables

DB Browser for SQLite - G:\My Drive\Docutments\School Work\2025-2026\Python Practice\U1-lesson1-Introduction to Database\Kuwait_landmarks.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Undo Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Table: landmark

	id	name	location	category	year	price	info
1	1	Al-Tahrir Tower	Central Kuwait City	Architecture	1996	0.0	Al-Tahrir Tower is one of the most ...
2	2	Kuwait National Museum	Al-Murqab Area	Historical	1983	2.5	The museum houses cultural and ...
3	3	The Avenues Mall	Al-Rai Area	Shopping	2007	0.0	One of the largest and most popular...
4	4	Failaka Island	Off the eastern coast of Kuwait	Historical	1950	5.0	An island with archaeological sites...
5	5	Souq Al-Mubarakiya	Central Kuwait City	Traditional	1897	0.0	A traditional market offering a ...
6	6	Sheikh Jaber Al-Ahmad Bridge	Connecting Kuwait City to the south	Modern Infrastructure	2019	0.0	A remarkable modern bridge linking ...
7	7	Kuwait National Library	Al-Murqab Area	Cultural	1994	0.0	A prominent library housing an ...
8	8	Museum of Modern Art	Salmiya Area	Art	1980	1.0	Home to contemporary and modern art...
9	9	Sheikh Jaber Al-Ahmad Cultural ...	Kuwait City	Cultural	2016	0.0	It is a cultural center in Kuwait, ...
10	10	Grand Mosque of Kuwait	Kuwait City	Religious Architecture	1986	0.0	The Grand Mosque is the largest ...

1 - 10 of 10

عدد السجلات

Go to: 1



تضم دولة الكويت العديد من المعالم السياحية، التراثية، والدينية المُميزة، ونحتاج إلى حفظ معلوماتها في قاعدة بيانات رقمية، تتضمن أشهر تلك المعالم.

عائنا بيانات الجدول landmark بقاعدة البيانات Kuwait_landmarks.db ، وذلك باتباع الخطوات التالية:

1. شغل برنامج DB Browser for SQLite.

2. استدع قاعدة البيانات Kuwait_landmarks.db من مجلد أوراق العمل.

3. عاين هيكلية الجدول landmark:

◀ أسماء الأعمدة.

..... :
id, name, location, category, year, price

4. عاين بيانات الجدول landmark ، وتعرف على مكوناته ثم استكمل ما يلي:

Central Kuwait City	:	موقع "Al-Tahrir Tower" location	◀
2007	:	سنة التأسيس year "The Avenues mall"	◀
10	:	عدد معالم دولة الكويت في الجدول	◀
Traditional	:	فئة category المَعلم "Souq Al-Mubarakiya"	◀

5. أغلق قاعدة البيانات Kuwait_landmarks.db

6. إنهاء البرنامج.

صيغة التعليمة البرمجية للاتصال بقاعدة البيانات SQLite في Python

كائن (مُتغير) للاتصال بقاعدة البيانات

Database Connection Object



دالة من مكتبة sqlite3

لإنشاء / فتح اتصال بقاعدة البيانات



```
connection = sqlite3 . connect ('database_name.db')
```



مكتبة قواعد البيانات



اسم قاعدة البيانات

- إذا كانت قاعدة البيانات غير موجودة يتم إنشاؤها تلقائياً.
- الامتداد الافتراضي لملف قاعدة البيانات من نوع SQLite هو .db.

إغلاق الاتصال `connection.close()`

- يجب التأكد من إغلاق الاتصال بقاعدة البيانات بشكل صحيح، للأسباب التالية:
 - ◀ منع تسرب الموارد (مثل الذاكرة).
 - ◀ ضمان سلامة البيانات وعدم فقد البيانات.
 - ◀ تجنب حدوث أخطاء قد تعيق فتح قاعدة البيانات مرة أخرى، نتيجة بقاء الاتصال بها مفتوحاً دون إغلاق.



ورقة عمل (2)

أنشئ قاعدة بيانات Kuwait_landmarks.db لتوثيق معالم دولة الكويت، باتباع الخطوات التالية:

1. أنشئ مشروعاً جديداً باسم landmark_database، مستخدماً بيئة التطوير المناسبة IDE.

2. أنشئ ملف Python جديد باسم data_database.py.

▶ استدع مكتبة sqlite3.

▶ أنشئ اتصال بقاعدة البيانات.

▶ أغلق الاتصال بقاعدة البيانات.

3. نفذ التعليمات البرمجية.

4. تأكد من إنشاء ملف قاعدة البيانات بمجلد المشروع.

• تطوير البرنامج: طور البرنامج ليشمل معالجة الاستثناءات.



ورقة عمل (2)

أنشئ قاعدة بيانات Kuwait_landmarks.db لتوثيق معالم دولة الكويت، باتباع الخطوات التالية:

1. أنشئ مشروعًا جديدًا باسم landmark_database، مستخدمًا بيئة التطوير المناسبة IDE.

2. أنشئ ملف Python جديد باسم data_database.py.

▶ استدع مكتبة sqlite3.

▶ أنشئ اتصال بقاعدة البيانات.

▶ أغلق الاتصال بقاعدة البيانات.

3. نفذ التعليمات البرمجية.

4. تأكد من إنشاء ملف قاعدة البيانات بمجلد المشروع.

#الحل الأول

```
import sqlite3
```

```
connection = sqlite3.connect("Kuwait_landmarks.db")  
connection.close()
```

• تطوير البرنامج: طور البرنامج ليشمل معالجة الاستثناءات.



ورقة عمل (2)

#الحل المطور

`import sqlite3``connection = None``try:` `connection = sqlite3.connect("Kuwait_landmarks.db")` `print("connection successful")``except Exception as e:` `print("error: connection failed")` `print(e)``finally:` `if connection is not None:` `connection.close()` `print("connection closed")`

أنشئ قاعدة بيانات Kuwait_landmarks.db لتوثيق معالم دولة الكويت، باتباع الخطوات التالية:

1. أنشئ مشروعًا جديدًا باسم landmark_database، مستخدمًا بيئة التطوير المناسبة IDE.

2. أنشئ ملف Python جديد باسم data_database.py.

▶ استدع مكتبة sqlite3.

▶ أنشئ اتصال بقاعدة البيانات.

▶ أغلق الاتصال بقاعدة البيانات.

3. نفذ التعليمات البرمجية.

4. تأكد من إنشاء ملف قاعدة البيانات بمجلد المشروع.

• تطوير البرنامج: طور البرنامج ليشمل معالجة الاستثناءات.

صيغة التعليمة البرمجية لإنشاء جدول في قاعدة البيانات SQLite في Python

تنفيذ أمر SQL
داخل قاعدة البيانات

اسم الجدول عبارة شرطية اختيارية أمر SQL لإنشاء جدول

▼ ▼ ▼

```
cursor.execute("""
```

```
CREATE TABLE IF NOT EXISTS table_name (
```

```
    column1 data_type constraints,
```

```
    column2 data_type constraints,
```

```
    ...
```

قُيود اختيارية نوع البيانات أسماء الأعمدة

▲ ▲ ▲

```
)
```

```
""")
```



ورقة عمل (3)

إنشاء جدول جديد landmark بقاعدة البيانات Kuwait_landmarks.db لتخزين بيانات بعض معالم دولة الكويت، مُتبعًا الخطوات التالية:

1. استدع المشروع landmark_table.

2. افتح ملف data_table.py.

3. استكمل التعليمة البرمجية لإنشاء جدول landmark. وفقًا للبيانات التالية:

اسم العمود	النوع	القيود
id	INTEGER	المفتاح الأساسي PRIMARY KEY
name	TEXT	غير فارغ NOT NULL
location	TEXT	-
category	TEXT	-
year	INTEGER	-
price	REAL	-

جدول (3-4) جدول landmark بقاعدة البيانات Kuwait_landmarks.db.

4. نفذ التعليمات البرمجية.

5. عاين هيكلية الجدول باستخدام برنامج DB Browser.

```
import sqlite3

connection = sqlite3.connect('Kuwait_landmarks.db')

cursor = connection.cursor()
```

```
connction.commit()

connection.close()
```



ورقة عمل (3)

إنشاء جدول جديد landmark بقاعدة البيانات Kuwait_landmarks.db لتخزين بيانات بعض معالم دولة الكويت، مُتبعًا الخطوات التالية:

1. استدع المشروع landmark_table.

2. افتح ملف data_table.py.

3. استكمل التعليمة البرمجية لإنشاء جدول landmark. وفقًا للبيانات التالية:

اسم العمود	النوع	القيود
id	INTEGER	المفتاح الأساسي PRIMARY KEY
name	TEXT	غير فارغ NOT NULL
location	TEXT	-
category	TEXT	-
year	INTEGER	-
price	REAL	-

جدول (4-3) landmark بقاعدة البيانات Kuwait_landmarks.db.

4. نفذ التعليمات البرمجية.

5. عاين هيكلية الجدول باستخدام برنامج DB Browser.

```
import sqlite3

connection = sqlite3.connect('Kuwait_landmarks.db')

cursor = connection.cursor()

cursor.execute('''
    CREATE TABLE IF NOT EXISTS landmark(
        id INTEGER PRIMARY KEY,
        name TEXT NOT NULL,
        location TEXT,
        category TEXT,
        year INTEGER,
        price REAL
    )
''')

connection.commit()

connection.close()
```

صيغة التعليمة البرمجية لإضافة البيانات SQLite في Python

يُنفذ أمر SQL
داخل قاعدة البيانات

اسم الجدول

عناصر مؤقتة placeholders
للقيم المطلوب إدخالها

`cursor.execute ("INSERT INTO table_name (column1, column2, ...) VALUES (?, ?, ...)", (value1, value2, ...))`

أمر SQL
لإضافة بيانات للجدول

أسماء الأعمدة

القيم المطلوب إدخالها

```
cursor.execute(''  
    INSERT INTO book(title, author, year, category, price)  
    VALUES( ?, ?, ?, ?, ?) '' ,  
    ('Arabic', 'Arabic SD', 2001, 'Art/Science', None)  
)
```



ورقة عمل (4)

إضافة سجل لجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_insert.

2. افتح ملف data_insert1.py.

3. استكمل التعليمة البرمجية لإضافة سجل وفقاً للبيانات التالية:

Column names	id	name	location	category	year	price
Data values		Abdullah AlSalem Centre	Al-Shaab Area	Culture	2018	

جدول رقم (3-4) يوضح بيانات سجل مطلوب إضافته لجدول landmark بقاعدة البيانات Kuwait_landmarks.db.

4. نفذ التعليمات البرمجية.

5. عاين بيانات الجدول باستخدام برنامج DB Browser.

```
import sqlite3
```

```
# Connect to the database
```

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

```
cursor = connection.cursor()
```

```
# Insert data into table book
```

```
cursor.execute(
```

```
)
```

```
# Commit changes
```

```
connection.commit()
```

```
# Close database
```

```
connection.close()
```




ورقة عمل (4)

إضافة سجل لجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_insert.

2. افتح ملف data_insert1.py.

3. استكمل التعليمة البرمجية لإضافة سجل وفقاً للبيانات التالية:

Column names	id	name	location	category	year	price
Data values		Abdullah AlSalem Centre	Al-Shaab Area	Culture	2018	

جدول رقم (3-4) يوضح بيانات سجل مطلوب إضافته لجدول landmark بقاعدة البيانات Kuwait_landmarks.db.

4. نفذ التعليمات البرمجية.

5. عاين بيانات الجدول باستخدام برنامج DB Browser.

```
import sqlite3
```

```
# Connect to the database
```

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

```
cursor = connection.cursor()
```

```
# Insert data into table book
```

```
cursor.execute('''
    INSERT INTO landmark (name, location, category, year, price)
    VALUES (?, ?, ?, ?, ?)
''',
    ('Abdullah Alsalem Center', 'Al-Shaab Area', 'Culture', 2018, None)
)
```

```
# Commit changes
```

```
connection.commit()
```

```
# Close database
```

```
connection.close()
```

صيغة التعليمة البرمجية لإضافة أكثر من سجل

```
cursor.executemany('''  
    INSERT INTO book(title,author,year,category,price)  
    VALUES(?,?,?,?,?)''',  
[  
    ('ICT','ICT SD',2001,'Art/Science',None),  
    ('Physics','Science SD',2007,'Science',1.500),  
    ('English','English SD',2009,'Art/Science',None)  
]  
)
```

```

import sqlite3

connection = None

try:
    connection=sqlite3.connect('Kuwait_landmarks.db')
    cursor=connection.cursor()

    # Insert data into landmark table
    insert_sql="....."

    landmarks_list = [(('Failaka Island', 'Off the eastern coast of Kuwait', 'Historical Landmark', 1950,5),
                        ('Souq Al-Mubarakiya', 'Central Kuwait City', 'Traditional Market', 1897,0),
                        ('Museum of Modern Art', 'Salmiya Area', 'Art Museum', 1980,1))]

    cursor.executemany(.....)

    connection.commit()

except Exception as e:
    print("Error:", e)

finally:
    if connection is not None:
        connection.close()

```

إضافة عدة سجلات مرة واحدة لجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_insert.
2. افتح ملف data_insert2.py.
3. استكمل التعليمة البرمجية لإضافة الصفوف المخصصة للمتغير landmarks_list.
4. نفذ التعليمات البرمجية.
5. عاين بيانات الجدول باستخدام برنامج DB Browser.

إضافة عدة سجلات مرة واحدة لجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_insert.

2. افتح ملف data_insert2.py.

3. استكمل التعليمة البرمجية لإضافة الصفوف المخصصة للمتغير landmarks_list.

4. نفذ التعليمات البرمجية.

5. عاين بيانات الجدول باستخدام برنامج DB Browser.

```
import sqlite3

connection = None

try:
    connection=sqlite3.connect('Kuwait_landmarks.db')
    cursor=connection.cursor()

    # Insert data into landmark table
    insert_sql= "INSERT INTO landmark (name, location, category, year, price) VALUES (?, ?, ?, ?, ?)"

    landmarks_list = [(('Failaka Island', 'Off the eastern coast of Kuwait', 'Historical Landmark', 1950, 5),
                        ('Souq Al-Mubarakiya', 'Central Kuwait City', 'Traditional Market', 1897, 0),
                        ('Museum of Modern Art', 'Salmiya Area', 'Art Museum', 1980, 1))]

    cursor.executemany(insert_sql, landmarks_list)

    connection.commit()

except Exception as e:
    print("Error:", e)

finally:
    if connection is not None:
        connection.close()
```

صيغة التعليمة البرمجية لإنشاء استعلامات SQLite في Python

يُنفذ أمر SQL داخل قاعدة البيانات

أسماء الأعمدة المطلوب استرجاعها

العمود المُستخدم في جملة الشرط

قيمة الشرط

```
cursor.execute ("SELECT column1, column2, ... FROM tabel_name WHERE condition_column = ?", (condition_value,))
```

أمر SQL لاسترجاع البيانات

اسم الجدول

عُنصر مُؤقت placeholder لقيمة الشرط

طرق استخدام دوال fetch

```
cursor.execute("SELECT * FROM book")
books=cursor.fetchall()
# print records in separate lines
for book in books:
    print(book)
```

Fetch Functions

fetchone()

تسترجع صفًا واحدًا فقط من النتائج في كل مرة.

fetchmany(size)

تسترجع عددًا محددًا من الصفوف حسب المعامل size.

fetchall()

تسترجع جميع الصفوف دفعة واحدة.

العوامل الشرطية في SQLite



النوع: خارج قائمة قيم
العوامل الشرطية:

NOT IN (value1, value2, value3, ...)

مثال:

price **NOT IN** (10,20,25)



النوع: ضمن قائمة قيم
العوامل الشرطية:

IN (value1, value2, value3, ...)

مثال:

price **IN** (10,20,25)



النوع: الاختلاف

العوامل الشرطية: **!=** **<>**

مثال:

price **<>** 20 price **!=** 20



النوع: المساواة

العامل الشرطي: **=**

مثال:

price **=** 20



النوع: التحقق من القيم غير الفارغة
العوامل الشرطية:

IS NOT NULL

مثال: **IS NOT NULL**

title **IS NOT NULL** **AND** price **IS NOT NULL**



النوع: التحقق من القيم الفارغة
العوامل الشرطية:

IS NULL

مثال:

title **IS NULL**



النوع: أكبر من أو يساوي /
أصغر من أو يساوي

العوامل الشرطية: **>=** **<=**

مثال:

price **>=** 20 price **<=** 20



النوع: أكبر من / أصغر من

العوامل الشرطية: **>** **<**

مثال:

price **>** 20 price **<** 20



النوع: لا يُطابق نمط جزئي نصي
العوامل الشرطية:

NOT LIKE 'pattern' →

% : تُمثل مجموعة أحرف مجهولة _ : تُمثل حرف مجهول

مثال: 'com%' **NOT LIKE** title عنوان (كتاب) لا يبدأ ب com



النوع: يُطابق نمط جزئي نصي
العوامل الشرطية:

LIKE 'pattern' →

مثال: 'com%' **LIKE** title عنوان (كتاب) يبدأ ب com



النوع: خارج النطاقات
العوامل الشرطية:

NOT BETWEEN value1 **AND** value2

مثال:

price **NOT BETWEEN** 10 **AND** 20



النوع: داخل النطاقات
العوامل الشرطية:

BETWEEN value1 **AND** value2

مثال:

price **BETWEEN** 10 **AND** 20



ورقة عمل (6)

الاستعلام عن جميع البيانات من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db
مستخدمًا التعليمة fetchall، مُتبعًا الخطوات التالية:

```
import sqlite3

# Connect to the database
connection = sqlite3.connect('Kuwait_landmarks.db')
cursor = connection.cursor()

# Data Retrieval
cursor.execute(
    )
rows = cursor.fetchall()
for row in rows:

# Close the database
connection.close()
```

1. استدع المشروع landmark_select.

2. افتح ملف data_select1.py.

3. استكمل التعليمات البرمجية لتنفيذ التالي:

◀ استعلم عن جميع صفوف الجدول.

◀ اطبع الصفوف التي تم استرجاعها (الاستعلام عنها).

4. نفذ التعليمات البرمجية.

5. عاين نتائج الاستعلام.

◀ عدد الصفوف:



ورقة عمل (6)

الاستعلام عن جميع البيانات من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db مستخدماً التعليمة fetchall، مُتبعاً الخطوات التالية:

1. استدع المشروع landmark_select.

2. افتح ملف data_select1.py.

3. استكمل التعليمات البرمجية لتنفيذ التالي:

◀ استعلم عن جميع صفوف الجدول.

◀ اطبع الصفوف التي تم استرجاعها (الاستعلام عنها).

4. نفذ التعليمات البرمجية.

5. عاين نتائج الاستعلام.

10

◀ عدد الصفوف:

```
import sqlite3

# Connect to the database
connection = sqlite3.connect('Kuwait_landmarks.db')
cursor = connection.cursor()

# Data Retrieval
cursor.execute( "SELECT * FROM landmark" )
rows = cursor.fetchall()

for row in rows:
    print(row)

# Close the database
connection.close()
```


الاستعلام عن بيانات محددة من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db مُستخدمًا التعليمة fetchall، مُتبعًا الخطوات التالية:

```
import sqlite3
```

```
# Connect to the database
```

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

```
cursor = connection.cursor()
```

```
# Data Retrieval
```

```
cursor.execute(
```

```
# Fetch the Data
```

```
# Output the Data
```

```
# Close the database
```

```
connection.close()
```

1. استدع المشروع landmark_select.

2. افتح ملف data_select2.py.

3. استكمل التعليمات البرمجية اللازمة لتنفيذ الاستعلامات التالية:

استعلم عن عمودي (اسم معلم الكويت name، وسنة الإنشاء year) لجميع صفوف المعالم الثقافية Cultural فقط.

استخدم دالة fetchall() لاسترجاع البيانات وتخزينها في متغير records.

اكتب التعليمة البرمجية للحلقة التكرارية للمتغير records لطباعة كل صف في سطر منفصل.

4. نفذ التعليمات البرمجية.

5. عاين نتائج الاستعلام.

الاستعلام عن بيانات محددة من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db مُستخدمًا التعليمة fetchall، مُتبعًا الخطوات التالية:

```
import sqlite3
```

```
# Connect to the database
```

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

```
cursor = connection.cursor()
```

```
# Data Retrieval
```

```
cursor.execute( "SELECT name, year FROM landmark WHERE category=?", ("Cultural",) )
```

```
# Fetch the Data
```

```
records = cursor.fetchall()
```

```
# Output the Data
```

```
for record in records:
    print(record)
```

```
# Close the database
```

```
connection.close()
```

1. استدع المشروع landmark_select.

2. افتح ملف data_select2.py.

3. استكمل التعليمات البرمجية اللازمة لتنفيذ الاستعلامات التالية:

استعلم عن عمودي (اسم معلم الكويت name، وسنة الإنشاء year) لجميع صفوف

المعالم الثقافية Cultural فقط.

استخدم دالة fetchall() لاسترجاع البيانات وتخزينها في متغير records.

اكتب التعليمة البرمجية للحلقة التكرارية للمتغير records لطباعة كل صف في

سطر منفصل.

4. نفذ التعليمات البرمجية.

5. عاين نتائج الاستعلام.

صيغة التعليمة البرمجية لتحديث قاعدة البيانات SQLite في Python

يُنفذ أمر SQL
داخل قاعدة البيانات

اسم الجدول

عُنصر مؤقت placeholder
للقِيمة الجديدة

عُنصر مؤقت placeholder
لقيمة الشرط

قيمة الشرط

```
cursor.execute("UPDATE table_name SET column_name = ? WHERE condition_column = ?", (value, condition_value))
```

أمر SQL لتحديث البيانات

اسم العمود المطلوب
تحديث بياناته

العمود المُستخدم في
جملة الشرط

القيمة الجديدة

ملاحظة:

● يتم تحديث جميع الصفوف في الجدول إذا لم يتم تحديد شرط.



ورقة عمل (8)

تعديل / تحديث بيانات الجدول landmark بقاعدة البيانات Kuwait_landmarks.db ، باتباع الخطوات التالية:

1. استدع المشروع landmark_update.

2. افتح ملف data_update1.py.

3. استكمل التعليمات البرمجية لتنفيذ التالي:

◀ حدث بيانات الصف (id=1) لتصبح قيمة تذكرة الدخول تساوي دينارين (price = 2).

4. نفذ التعليمات البرمجية.

◀ عدد الصفوف التي تم تعديلها =

```
import sqlite3
connection = sqlite3.connect('Kuwait_landmarks.db')
cursor = connection.cursor()

# Update Data
cursor.execute(

)

print("Number of rows updated:", cursor.rowcount)
connection.commit()
connection.close()
```



ورقة عمل (8)

تعديل / تحديث بيانات الجدول landmark بقاعدة البيانات Kuwait_landmarks.db ، باتباع الخطوات التالية:

1. استدع المشروع landmark_update.

2. افتح ملف data_update1.py.

3. استكمل التعليمات البرمجية لتنفيذ التالي:

◀ حدث بيانات الصف (id=1) لتصبح قيمة تذكرة الدخول تساوي دينارين (price = 2).

4. نفذ التعليمات البرمجية.

◀ عدد الصفوف التي تم تعديلها =1.....

```
import sqlite3
connection = sqlite3.connect('Kuwait_landmarks.db')
cursor = connection.cursor()

# Update Data
cursor.execute( '''
    UPDATE landmark
    SET price=?
    WHERE id=?
    ''', (2,1)
)

print("Number of rows updated:", cursor.rowcount)
connection.commit()
connection.close()
```

تعديل / تحديث بيانات الجدول landmark بقاعدة البيانات Kuwait_landmarks.db ، باتباع الخطوات التالية:

```
import sqlite3
```

1. استدع المشروع .landmark_update

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

2. افتح ملف .data_update2.py

```
cursor = connection.cursor()
```

3. استكمل التعليمات البرمجية لتنفيذ التالي:

```
# Update Data
```

◀ تحديث جميع صفوف المعالم الثقافية أو التاريخية

```
cursor.execute(
```

◀ category = 'Historical', category = 'Cultural', ليصبح سعر الدخول مجاني (price=0).

4. نفذ التعليمات البرمجية.

```
)
```

◀ عدد الصفوف التي تم تعديلها =

```
print("Number of rows updated:", cursor.rowcount)
```

```
connection.commit()
```

```
connection.close()
```

تعديل / تحديث بيانات الجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

```
import sqlite3
```

1. استدع المشروع .landmark_update

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

2. افتح ملف .data_update2.py

```
cursor = connection.cursor()
```

3. استكمل التعليمات البرمجية لتنفيذ التالي:

```
# Update Data
```

◀ تحديث جميع صفوف المعالم الثقافية أو التاريخية

```
cursor.execute( '''
    UPDATE landmark
    SET price=?
    WHERE category=? OR category =?
    ''', (0, 'Cultural', 'Historical')
)
```

◀ category = 'Cultural', category = 'Historical' ليصبح سعر الدخول مجاني (price=0).

4. نفذ التعليمات البرمجية.

◀ عدد الصفوف التي تم تعديلها = 4

```
print("Number of rows updated:", cursor.rowcount)
```

```
connection.commit()
```

```
connection.close()
```

صيغة التعليمة البرمجية لحذف البيانات في SQLite Python

يُنفذ امر SQL
داخل قاعدة البيانات

اسم الجدول

عنصر مؤقت placeholder
لقيمة الشرط

`cursor.execute ("DELETE FROM table_name WHERE column_name = ?", (value))`

أمر SQL لحذف
صف / صفوف من الجدول

العمود المُستخدم في
جملة الشرط

قيمة الشرط

ملاحظة:

تُحذف جميع الصفوف في الجدول، إذا لم يتم تحديد شرط الحذف.



ورقة عمل (10)

حذف صفوف من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_delete.

2. افتح ملف data_delete1.py.

3. استكمل التعليمات البرمجية:

◀ احذف جميع صفوف معالم الكويت الثقافية. (category = "Cultural").

4. نفذ التعليمات البرمجية.

◀ عدد الصفوف التي تم حذفها =

```
import sqlite3
```

```
# Connect to the database
```

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

```
cursor = connection.cursor()
```

```
# Delete Data
```

```
cursor.execute(
```

```
)
```

```
print("Number of rows deleted:", cursor.rowcount)
```

```
# Commit changes
```

```
connection.commit()
```

```
# Close database
```

```
connection.close()
```



ورقة عمل (10)

حذف صفوف من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_delete.

2. افتح ملف data_delete1.py.

3. استكمل التعليمات البرمجية:

◀ احذف جميع صفوف معالم الكويت الثقافية. (category = "Cultural").

4. نفذ التعليمات البرمجية.

◀ عدد الصفوف التي تم حذفها =
2

```
import sqlite3
```

```
# Connect to the database
```

```
connection = sqlite3.connect('Kuwait_landmarks.db')
```

```
cursor = connection.cursor()
```

```
#Delete Data
```

```
cursor.execute( '''  
    DELETE FROM landmark  
    WHERE category=?  
    ''', ('Cultural',)  
)
```

```
print("Number of rows deleted:", cursor.rowcount)
```

```
# Commit changes
```

```
connection.commit()
```

```
# Close database
```

```
connection.close()
```

```

import sqlite3

# Connect to the database
connection = sqlite3.connect('Kuwait_landmarks.db')
cursor = connection.cursor()

#Delete Data
cursor.execute(

)

print("Number of rows deleted:", cursor.rowcount)

# Commit changes
connection.commit()

# Close database
connection.close()

```

حذف صفوف من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_delete.

2. افتح ملف data_delete2.py.

3. استكمل التعليمات البرمجية:

احذف الصفوف التي تاريخ إنشائها year قبل العام 2000م، وسعر تذكرة الدخول

price أكبر من دينار واحد.

4. نفذ التعليمات البرمجية.

عدد الصفوف التي تم حذفها =

```

import sqlite3

# Connect to the database
connection = sqlite3.connect('Kuwait_landmarks.db')
cursor = connection.cursor()

#Delete Data
cursor.execute( '''
    DELETE FROM landmark
    WHERE year<? AND price=?
    ''', (2000, 1)
)

print("Number of rows deleted:", cursor.rowcount)

# Commit changes
connection.commit()

# Close database
connection.close()

```

حذف صفوف من الجدول landmark بقاعدة البيانات Kuwait_landmarks.db، باتباع الخطوات التالية:

1. استدع المشروع landmark_delete.

2. افتح ملف data_delete2.py.

3. استكمل التعليمات البرمجية:

احذف الصفوف التي تاريخ إنشائها year قبل العام 2000م، وسعر تذكرة الدخول

price أكبر من دينار واحد.

4. نفذ التعليمات البرمجية.

عدد الصفوف التي تم حذفها =1.....