

استفاده از توابع و شرایط SQL در SQLite

SQLite امکانات قدرتمندی برای جستجو و پردازش داده‌ها دارد. در این بخش، استفاده از دستورات شرطی و توابع SQL را توضیح می‌دهیم.

۱. استفاده از دستورات شرطی در جستجو

دستورات شرطی مانند `WHERE`, `AND`, `OR`, `IN`, `BETWEEN` برای محدود کردن نتایج در جستجوها استفاده می‌شوند.

۱.۱. WHERE - جستجو با یک شرط خاص

مثال: دریافت کاربری که نامش "محمدجواد" است.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT * FROM users WHERE name = ?", ("محمدجواد",))
user = cursor.fetchone()

print(user)

conn.close()
```

📌 توضیح: شرط `WHERE name = ?` باعث می‌شود فقط سطرهای که نامش "محمدجواد" است، برگردانده شود.

۱.۲. AND - ترکیب چند شرط

مثال: دریافت کاربران بالای ۲۵ سال که ایمیل دارند.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT * FROM users WHERE age > ? AND email IS NOT NULL", (25,))
users = cursor.fetchall()

for user in users:
    print(user)

conn.close()
```

📌 توضیح: `AND` مشخص می‌کند که هر دو شرط باید برقرار باشند.

۱.۳. OR - یکی از شرطها برقرار باشد

مثال: دریافت کاربرانی که یا نامشان "محمدجواد" است یا سن آنها کمتر از ۲۵ سال است.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT * FROM users WHERE name = ? OR age < ?", ("25", "محمدجواد"))
users = cursor.fetchall()

for user in users:
    print(user)

conn.close()
```

📌 توضیح: اگر یکی از شرطها برقرار باشد، داده برگردانده می‌شود.

۱.۴. IN - جستجو در یک مجموعه مقادیر

مثال: دریافت کاربران با نام "محمدجواد"، "علی" یا "سارا".

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT * FROM users WHERE name IN (?, ?, ?)", ("سارا", "علی", "محمدجواد"))
users = cursor.fetchall()

for user in users:
    print(user)

conn.close()
```

📌 توضیح: IN باعث می‌شود که مقدار ستون name یکی از مقادیر مشخص شده باشد.

۱.۵. BETWEEN - جستجو در یک محدوده عددی

مثال: دریافت کاربران با سن بین ۲۰ تا ۳۰ سال.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT * FROM users WHERE age BETWEEN ? AND ?", (20, 30))
users = cursor.fetchall()

for user in users:
    print(user)

conn.close()
```

📌 توضیح: BETWEEN بازه‌ای از مقادیر را مشخص می‌کند. این شرط شامل دو مقدار ابتدا و انتها نیز می‌شود.

۲. استفاده از توابع SQL در SQLite

SQLite دارای توابع داخلی زیادی برای انجام پردازش روی داده‌ها است. برخی از مهم‌ترین آن‌ها عبارت‌اند از:

- ✓ COUNT () - شمارش تعداد سطرها
- ✓ SUM () - محاسبه مجموع مقادیر
- ✓ AVG () - میانگین مقادیر
- ✓ MAX () - بیشترین مقدار
- ✓ MIN () - کمترین مقدار

۲.۱. COUNT () - شمارش تعداد سطرها

مثال: شمارش تعداد کاربران ثبت‌شده در جدول.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT COUNT(*) FROM users")
count = cursor.fetchone()[0]

print("تعداد کاربران:", count)

conn.close()
```

📌 توضیح: این دستور تعداد کل کاربران را نمایش می‌دهد.

۲.۲. SUM () - مجموع یک ستون

مثال: محاسبه مجموع سن کاربران.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT SUM(age) FROM users")
total_age = cursor.fetchone()[0]

print("مجموع سن کاربران:", total_age)

conn.close()
```

📌 توضیح: SUM (age) تمام مقادیر ستون age را جمع می‌کند.

۲.۳. AVG () - میانگین مقادیر

مثال: محاسبه میانگین سن کاربران.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT AVG(age) FROM users")
average_age = cursor.fetchone()[0]

print("میانگین سن کاربران", average_age)

conn.close()
```

📌 توضیح: `AVG (age)` میانگین مقادیر ستون `age` را محاسبه می‌کند.

۲.۴. MAX () - بیشترین مقدار در یک ستون

مثال: دریافت بیشترین سن در بین کاربران.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT MAX(age) FROM users")
max_age = cursor.fetchone()[0]

print("بیشترین سن", max_age)

conn.close()
```

📌 توضیح: `MAX (age)` مقدار بیشینه ستون `age` را برمی‌گرداند.

۲.۵. MIN () - کمترین مقدار در یک ستون

مثال: دریافت کمترین سن در بین کاربران.

```
conn = sqlite3.connect("test.db")
cursor = conn.cursor()

cursor.execute("SELECT MIN(age) FROM users")
min_age = cursor.fetchone()[0]

print("کمترین سن", min_age)

conn.close()
```

📌 توضیح: `MIN (age)` مقدار کمینه ستون `age` را برمی‌گرداند.

جمع‌بندی

- ✓ `WHERE` برای فیلتر کردن داده‌ها استفاده می‌شود.
- ✓ `AND` و `OR` برای ترکیب چند شرط به کار می‌روند.
- ✓ `IN` برای جستجو در مجموعه‌ای از مقادیر استفاده می‌شود.
- ✓ `BETWEEN` برای جستجو در بازه عددی کاربرد دارد.
- ✓ توابع `MIN` , `MAX` , `AVG` , `SUM` , `COUNT` () برای پردازش آماری داده‌ها استفاده می‌شوند.

با استفاده از این مفاهیم، می‌توانید جستجوهای پیچیده‌تر و پردازش داده‌های بهتری در SQLite انجام دهید. اگر سوالی داشتی بپرس! 🚀