

زیرپرسش‌ها (Subqueries) در SQL به شما این امکان را می‌دهند که یک پرس‌وجو را داخل یک پرس‌وجوی دیگر قرار دهید. این می‌تواند برای انجام عملیات پیچیده‌تر و انتخاب داده‌هایی که به صورت مستقیم در دسترس نیستند، بسیار مفید باشد.

در ادامه نحوه استفاده از زیرپرسش‌ها و دستورات مرتبط مانند `IN`، `EXISTS` و `ANY` برای انجام عملیات پیچیده‌تر در SQL و پایتون را توضیح می‌دهیم.

1. استفاده از زیرپرسش‌ها (Subqueries)

زیرپرسش‌ها می‌توانند در بخش‌های مختلف یک دستور SQL استفاده شوند. معمولاً آن‌ها در بخش‌های `SELECT`، `WHERE` یا `FROM` به کار می‌روند.

1.1. زیرپرسش در بخش `WHERE`

در اینجا یک زیرپرسش در بخش `WHERE` استفاده می‌شود تا مقادیر مورد نظر را فیلتر کند.

```
import mysql.connector

# اتصال به پایگاه داده
conn = mysql.connector.connect(
    host="localhost",
    user="root",
    password="password",
    database="your_database"
)

cursor = conn.cursor()

# WHERE زیرپرسش در بخش
cursor.execute("""
    SELECT customer_id, name
    FROM customers
    WHERE customer_id IN (
        SELECT customer_id
        FROM orders
        WHERE total_amount > 1000
    )
""")

result = cursor.fetchall()

for row in result:
    print(row)
```

در این مثال:

- زیرپرسش `(SELECT customer_id FROM orders WHERE total_amount > 1000)` مشتریانی را انتخاب می‌کند که سفارش‌هایی با مجموع مبلغ بیشتر از 1000 دارند.
- سپس پرس‌وجوی اصلی تنها این مشتریان را نمایش می‌دهد.

1.2. زیرپرسش در بخش `SELECT`

گاهی اوقات می‌توان از زیرپرسش‌ها در بخش `SELECT` برای محاسبه مقادیر مختلف در همان زمان استفاده کرد.

```
cursor.execute("""
    SELECT customer_id,
           (SELECT COUNT(*) FROM orders WHERE customer_id = customers.customer_id) AS order_count
    FROM customers
""")
result = cursor.fetchall()

for row in result:
    print(row)
```

در اینجا:

- از یک زیرپرسش برای شمارش تعداد سفارش‌های هر مشتری استفاده شده است. این شمارش برای هر ردیف در نتیجه اصلی انجام می‌شود.

2. استفاده از `ANY`, `IN`, `EXISTS` در زیرپرسش‌ها

این دستورات به شما کمک می‌کنند تا پرس‌وجوهای پیچیده‌تری بسازید.

2.1. استفاده از `IN`

دستور `IN` برای مقایسه یک مقدار با مجموعه‌ای از مقادیر استفاده می‌شود. این معمولاً برای انتخاب مقادیر از یک زیرپرسش مفید است.

```
cursor.execute("""
    SELECT customer_id, name
    FROM customers
    WHERE customer_id IN (
        SELECT customer_id
        FROM orders
        WHERE status = 'Completed'
    )
""")
result = cursor.fetchall()

for row in result:
    print(row)
```

در اینجا:

- زیرپرسش `'SELECT customer_id FROM orders WHERE status = 'Completed'` تمامی مشتریانی را که سفارش‌هایشان کامل شده است، انتخاب می‌کند.
- سپس پرس‌وجوی اصلی تمامی این مشتریان را از جدول `customers` می‌آورد.

2.2. استفاده از `EXISTS`

دستور `EXISTS` برای بررسی وجود یک یا چند ردیف در نتیجه زیرپرسش استفاده می‌شود. این مفید است زمانی که فقط وجود داده‌ها مهم است.

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

```

SELECT customer_id, name
FROM customers
WHERE EXISTS (
    SELECT 1
    FROM orders
    WHERE orders.customer_id = customers.customer_id
    AND orders.status = 'Shipped'
)
)
"""
result = cursor.fetchall()

for row in result:
    print(row)

```

در اینجا:

- زیرپرسش بررسی می‌کند که آیا برای هر مشتری سفارش با وضعیت "Shipped" وجود دارد یا خیر.
- اگر چنین سفارشی وجود داشته باشد، آن مشتری در نتایج نمایش داده می‌شود.

2.3. استفاده از ANY

دستور ANY برای مقایسه یک مقدار با هر یک از مقادیر بازگشتی توسط زیرپرسش استفاده می‌شود. این معمولاً برای مقایسه با مجموعه‌ای از مقادیر مفید است.

```

cursor.execute("""
SELECT customer_id, name
FROM customers
WHERE total_spent > ANY (
    SELECT total_amount
    FROM orders
    WHERE status = 'Completed'
)
)
"""
result = cursor.fetchall()

for row in result:
    print(row)

```

در اینجا:

- دستور ANY بررسی می‌کند که آیا `total_spent` یک مشتری بیشتر از هر یک از `total_amount` هایی است که از زیرپرسش بدست آمده است.

3. استفاده از زیرپرسش‌ها در PostgreSQL

در PostgreSQL، نحوه استفاده از زیرپرسش‌ها مشابه MySQL است. فقط باید از کتابخانه `psycopg2` برای اتصال به PostgreSQL استفاده کنید.

3.1. مثال زیرپرسش در PostgreSQL

```

import psycopg2

# PostgreSQL اتصال به پایگاه داده
conn = psycopg2.connect(

```

```

dbname="your_database",
user="postgres",
password="password",
host="localhost"
)

cursor = conn.cursor()

# استفاده از زیرپرسش
cursor.execute("""
    SELECT customer_id, name
    FROM customers
    WHERE customer_id IN (
        SELECT customer_id
        FROM orders
        WHERE status = 'Completed'
    )
""")
result = cursor.fetchall()

for row in result:
    print(row)

```

4. نتیجه‌گیری

استفاده از زیرپرسش‌ها در SQL به شما این امکان را می‌دهد که پرس‌وجوهای پیچیده‌تری ایجاد کنید که در آن‌ها می‌توانید نتایج یک پرس‌وجو را در پرس‌وجوی دیگر استفاده کنید. دستورات `IN`، `EXISTS` و `ANY` به شما کمک می‌کنند تا از قدرت زیرپرسش‌ها بهره‌برداری بیشتری داشته باشید و عملیات پیچیده‌تری انجام دهید. این دستورات به طور گسترده در پایتون و هنگام کار با پایگاه داده‌ها استفاده می‌شوند.