

More Complex SQL Retrieval Queries

الأكثر تعقيدًا SQL استعلامات استرجاع

Additional features allow users to specify more complex retrievals from database: ■

■ تسمح الميزات الإضافية للمستخدمين بتحديد عمليات استرداد أكثر تعقيدًا من قاعدة البيانات:

■ Nested queries, joined tables, outer joins, aggregate functions, and grouping

■ الاستعلامات المتداخلة والجداول المرتبطة والصلات الخارجية والوظائف التجميعية والتجميع

Comparisons Involving NULL and Three-Valued Logic

مقارنات تتضمن منطقًا فارغًا وثلاثي القيم

■ Meanings of NULL معاني NULL

■ **Unknown value** قيمة غير معروفة

■ **Unavailable or withheld value** قيمة غير متاحة أو محتجزة

■ **Not applicable attribute** سمة غير قابلة للتطبيق

■ Each individual NULL value considered to be different from every other NULL value

■ تعتبر كل قيمة NULL فردية مختلفة عن كل قيمة NULL أخرى

■ SQL uses a three-valued logic: يستخدم SQL منطقًا ثلاثي القيم:

■ TRUE, FALSE, and UNKNOWN

Comparisons Involving NULL and Three-Valued Logic (cont'd.)

Table 5.1 Logical Connectives in Three-Valued Logic

(a)	AND	TRUE	FALSE	UNKNOWN
	TRUE	TRUE	FALSE	UNKNOWN
	FALSE	FALSE	FALSE	FALSE
	UNKNOWN	UNKNOWN	FALSE	UNKNOWN
(b)	OR	TRUE	FALSE	UNKNOWN
	TRUE	TRUE	TRUE	TRUE
	FALSE	TRUE	FALSE	UNKNOWN
	UNKNOWN	TRUE	UNKNOWN	UNKNOWN
(c)	NOT			
	TRUE	FALSE		
	FALSE	TRUE		
	UNKNOWN	UNKNOWN		

Comparisons Involving NULL and Three-Valued Logic (cont'd.)

SQL allows queries that check whether an attribute value is NULL ■

SQL يسمح بالاستعلامات التي تتحقق مما إذا كانت قيمة السمة فارغة

IS or IS NOT NULL ■

Query 18. Retrieve the names of all employees who do not have supervisors.

```
Q18:  SELECT  Fname, Lname
      FROM    EMPLOYEE
      WHERE   Super_ssn IS NULL;
```

Nested Queries, Tuples, and Set/Multiset Comparisons

استعلامات متداخلة ، مجموعات ، ومقارنات مجموعة / متعددة

■ Nested queries استعلامات متداخلة

■ Complete select-from-where blocks within WHERE clause of another query

■ أكمل كتل التحديد من حيث داخل عبارة WHERE لاستعلام آخر

■ Outer query الاستعلام الخارجي

■ Comparison operator IN عامل المقارنة IN

■ Compares value v with a set (or multiset) of values V يقارن القيمة v بمجموعة (أو مجموعة متعددة) من القيم V .

■ Evaluates to TRUE if v is one of the elements in V

يتم التقييم إلى TRUE إذا كان v أحد العناصر الموجودة في V

Nested Queries (cont'd.)

Q4A: SELECT DISTINCT Pnumber
 FROM PROJECT
 WHERE Pnumber IN
 (SELECT Pnumber
 FROM PROJECT, DEPARTMENT, EMPLOYEE
 WHERE Dnum=Dnumber AND
 Mgr_ssn=Ssn AND Lname='Smith')

 OR
 Pnumber IN
 (SELECT Pno
 FROM WORKS_ON, EMPLOYEE
 WHERE Essn=Ssn AND Lname='Smith');

Nested Queries (cont'd.)

Use tuples of values in comparisons ■

■ استخدم مجموعات القيم في المقارنات

Place them within parentheses ■

■ ضعهم بين قوسين

```
SELECT DISTINCT Essn
FROM WORKS_ON
WHERE (Pno, Hours) IN ( SELECT Pno, Hours
                        FROM WORKS_ON
                        WHERE Essn='123456789' );
```

Nested Queries (cont'd.)

Use other comparison operators to compare a single

value v استخدم عوامل المقارنة الأخرى لمقارنة قيمة واحدة

= ANY (or = SOME) operator ■

Returns TRUE if the value v is equal to some value •

• تُرجع TRUE إذا كانت القيمة v تساوي بعض القيمة

in the set V and is hence equivalent to IN

في المجموعة V وبالتالي يعادل IN

Other operators that can be combined with ANY ■
(or SOME): >, >=, <, <=, and <>

```
SELECT      Lname, Fname
FROM        EMPLOYEE
WHERE       Salary > ALL ( SELECT      Salary
                           FROM        EMPLOYEE
                           WHERE       Dno=5 );
```


Nested Queries (cont'd.)

Avoid potential errors and ambiguities ■

تجنب الأخطاء والغموض المحتملة ■

Create tuple variables (aliases) for all tables referenced in ■

SQL query قم بإنشاء متغيرات (tuple أسماء مستعارة) لجميع الجداول المشار إليها في استعلام SQL

Query 16. Retrieve the name of each employee who has a dependent with the same first name and is the same sex as the employee.

```
Q16:  SELECT      E.Fname, E.Lname
        FROM      EMPLOYEE AS E
        WHERE      E.Ssn IN ( SELECT      Essn
                                FROM      DEPENDENT AS D
                                WHERE      E.Fname=D.Dependent_name
                                AND E.Sex=D.Sex );
```

Correlated Nested Queries

الاستعلامات المتداخلة المرتبطة

Correlated nested query ■

استعلام متداخل مرتبط ■

Evaluated once for each tuple in the outer query ■

تقييم مرة واحدة لكل مجموعة في الاستعلام الخارجي ■

The EXISTS and UNIQUE Functions in SQL

وفريدة من نوعها EXISTS

وظائف في SQL
وظيفة EXISTS

EXISTS function ■

- Check whether the result of a correlated nested query is empty or not ■
مترابط فارغة أم لا
تحقق مما إذا كانت نتيجة استعلام متداخل

EXISTS and NOT EXISTS ■

- Typically used in conjunction with a correlated nested query ■
يتم استخدامه عادةً مع استعلام متداخل مترابط

SQL function UNIQUE (Q) ■

- Returns TRUE if there are no duplicate tuples in the result of query Q ■
مكررة في نتيجة الاستعلام Q
TRUE في حالة عدم وجود مجموعات

Explicit Sets and Renaming of Attributes in SQL

مجموعات صريحة وإعادة تسمية السمات في

SQL

Can use explicit set of values in WHERE ■

clause يمكن استخدام مجموعة صريحة من القيم في جملة
WHERE

Use qualifier AS followed by desired new ■
name استخدم المؤهل AS متبوعًا بالاسم الجديد المطلوب

Rename any attribute that appears in the result of ■
a query أعد تسمية أي سمة تظهر في نتيجة الاستعلام

```
Q8A:  SELECT  E.Lname AS Employee_name, S.Lname AS Supervisor_name
        FROM    EMPLOYEE AS E, EMPLOYEE AS S
        WHERE   E.Super_ssn=S.Ssn;
```

Joined Tables in SQL and Outer Joins

والصلات SQL الجداول المنضمة في
الخارجية

■ **Joined table** انضم إلى الجدول

■ Permits users to specify a table resulting from a join operation in the FROM clause of a query

■ يسمح للمستخدمين بتحديد جدول ناتج عن عملية ربط في عبارة FROM للاستعلام

■ **The FROM clause in Q1A** بند FROM في Q1A

■ **Contains a single joined table** يحتوي على جدول مرتبط واحد

```
Q1A:  SELECT  Fname, Lname, Address
        FROM    (EMPLOYEE JOIN DEPARTMENT ON Dno=Dnumber)
        WHERE   Dname='Research';
```

Joined Tables in SQL and Outer Joins (cont'd.)

- Specify different types of join تحديد أنواع مختلفة من الصلة
- NATURAL JOIN انضمام طبيعي
- Various types of OUTER JOIN أنواع مختلفة من OUTER JOIN
- NATURAL JOIN on two relations R and S
- NATURAL JOIN على علاقتين R و S.
- No join condition specified لم يتم تحديد شرط الانضمام
- Implicit EQUIJOIN condition for each pair of attributes with same name from R and S
- شرط EQUIJOIN الضمني لكل زوج من السمات بنفس الاسم من R و S.

Joined Tables in SQL and Outer Joins (cont'd.)

Inner join صلة داخلية ■

- Default type of join in a joined table ■

- النوع الافتراضي للانضمام في جدول مرتبط ■

- Tuple is included in the result only if a matching tuple exists in the other relation ■

- يتم تضمين Tuple في النتيجة فقط في حالة وجود مجموعة مطابقة في العلاقة الأخرى ■

LEFT OUTER JOIN ترك صلة خارجية ■

- Every tuple in left table must appear in result ■

- يجب أن تظهر كل مجموعة في الجدول الأيسر في النتيجة ■

- If no matching tuple إذا لم يكن هناك بنية مطابقة ■

- Padded with NULL values for attributes of right table •

- مبطن بقيم فارغة لسمات الجدول الأيمن •

Joined Tables in SQL and Outer Joins (cont'd.)

RIGHT OUTER JOIN حق الانضمام الخارجي ■

Every tuple in right table must appear in result ■

يجب أن تظهر كل مجموعة في الجدول الأيمن في النتيجة ■

If no matching tuple إذا لم يكن هناك بنية مطابقة ■

Padded with NULL values for the attributes of left •

table مبطن بقيم فارغة لسمات الجدول الأيسر

FULL OUTER JOIN الانضمام الخارجي الكامل ■

Can nest join specifications ■

يمكن أن تتداخل مع المواصفات ■

Aggregate Functions in SQL

وظائف مجمعة في

SQL

Used to summarize information from multiple tuples into a single-tuple summary ■

يستخدم لتلخيص المعلومات من عدة مجموعات في ملخص مفرد ■

Grouping التجمع ■

Create subgroups of tuples before ■

summarizing أنشئ مجموعات فرعية من المجموعات
قبل التلخيص

Built-in aggregate functions وظائف مجمعة مدمجة ■

COUNT, SUM, MAX, MIN, and AVG ■

Functions can be used in the `SELECT` clause or in a `HAVING` clause ■

Aggregate Functions in SQL (cont'd.)

NULL values discarded when aggregate functions are applied to a particular column

يتم تجاهل القيم الفارغة عند تطبيق الدالات التجميعية على عمود معين

Query 20. Find the sum of the salaries of all employees of the 'Research' department, as well as the maximum salary, the minimum salary, and the average salary in this department.

```
Q20:  SELECT  SUM (Salary), MAX (Salary), MIN (Salary), AVG (Salary)
        FROM    (EMPLOYEE JOIN DEPARTMENT ON Dno=Dnumber)
        WHERE    Dname='Research';
```

Queries 21 and 22. Retrieve the total number of employees in the company (Q21) and the number of employees in the 'Research' department (Q22).

```
Q21:  SELECT  COUNT (*)
        FROM    EMPLOYEE;
```

```
Q22:  SELECT  COUNT (*)
        FROM    EMPLOYEE, DEPARTMENT
        WHERE    DNO=DNUMBER AND DNAME='Research';
```

Grouping: The GROUP BY and HAVING Clauses

GROUP BY و HAVING التجميع: بنود

- **Partition** relation into subsets of tuples

- علاقة التقسيم إلى مجموعات فرعية من المجموعات

- Based on **grouping attribute(s)**

- بناءً على سمات التجميع

- Apply function to each such group independently تطبيق

- الوظيفة على كل مجموعة بشكل مستقل

- **GROUP BY** clause

- Specifies grouping attributes يحدد سمات التجميع

- If NULLs exist in grouping attribute

- إذا كانت القيم الفارغة موجودة في خاصية التجميع

- Separate group created for all tuples with a NULL value in grouping attribute

- تم تكوين مجموعة منفصلة لجميع المجموعات بقيمة NULL في سمة التجميع

Grouping: The GROUP BY and HAVING Clauses (cont'd.)

HAVING clause ■

Provides a condition on the summary information ■

■ يوفر شرطاً على المعلومات الموجزة

Query 28. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than \$40,000.

```
Q28:  SELECT  Dnumber, COUNT (*)
        FROM    DEPARTMENT, EMPLOYEE
        WHERE   Dnumber=Dno AND Salary>40000 AND
              ( SELECT  Dno
                FROM    EMPLOYEE
                GROUP BY Dno
                HAVING  COUNT (*) > 5)
```

Discussion and Summary of SQL Queries

SQL مناقشة وملخص استعلامات

```
SELECT <attribute and function list>  
FROM <table list>  
[ WHERE <condition> ]  
[ GROUP BY <grouping attribute(s)> ]  
[ HAVING <group condition> ]  
[ ORDER BY <attribute list> ];
```

Specifying Constraints as Assertions and Actions as Triggers

تحديد القيود على أنها تأكيدات وإجراءات كمشغلات

■ **CREATE ASSERTION** إنشاء تأكيد

- Specify additional types of constraints outside scope of built-in relational model constraints

حدد أنواعًا إضافية من القيود خارج نطاق قيود النموذج العلائقي المضمنة

■ **CREATE TRIGGER** إنشاء المشغل

- Specify automatic actions that database system will perform when certain events and conditions occur

- حدد الإجراءات التلقائية التي سيقوم نظام قاعدة البيانات بتنفيذها عند حدوث أحداث وشروط معينة

Specifying General Constraints as Assertions in SQL

SQL تحديد القيود العامة كتأكدات في

CREATE ASSERTION ■

Specify a query that selects any tuples that violate the desired condition ■

حدد استعلامًا يحدد أي مجموعات تنتهك الشرط المطلوب ■

Use only in cases where it is not possible to use CHECK on attributes and domains ■
التحقق على السمات والمجالات

```
CREATE ASSERTION SALARY_CONSTRAINT
CHECK ( NOT EXISTS ( SELECT *
                     FROM   EMPLOYEE E, EMPLOYEE M,
                     DEPARTMENT D
                     WHERE  E.Salary>M.Salary
                          AND E.Dno=D.Dnumber
                          AND D.Mgr_ssn=M.Ssn ) );
```

Introduction to Triggers in SQL

SQL مقدمة في المشغلات في

CREATE TRIGGER statement ■

إنشاء بيان TRIGGER ■

Used to monitor the database ■
تستخدم لمراقبة قاعدة البيانات

Typical trigger has three components: ■

يحتوي المشغل النموذجي على ثلاثة مكونات: ■

Event(s) حدث (أحداث) ■

Condition شرط ■

Action عمل ■

Views (Virtual Tables) in SQL

SQL طرق العرض (الجدول الافتراضية) في

■ Concept of a view in SQL

■ مفهوم وجهة نظر في SQL

■ Single table derived from other tables

■ جدول مفرد مشتق من جداول أخرى

■ Considered to be a virtual table

■ يعتبر أن يكون الجدول الافتراضي

Specification of Views in SQL

SQL مواصفات طرق العرض في

■ **CREATE VIEW** command أمر إنشاء عرض

■ Give table name, list of attribute names, and a query to specify the contents of the view

■ قم بتزويد اسم الجدول وقائمة بأسماء السمات واستعلام لتحديد محتويات طريقة العرض

```
V1:  CREATE VIEW  WORKS_ON1
      AS SELECT    Fname, Lname, Pname, Hours
          FROM      EMPLOYEE, PROJECT, WORKS_ON
          WHERE     Ssn=Essn AND Pno=Pnumber;

V2:  CREATE VIEW  DEPT_INFO(Dept_name, No_of_emps, Total_sal)
      AS SELECT    Dname, COUNT (*), SUM (Salary)
          FROM      DEPARTMENT, EMPLOYEE
          WHERE     Dnumber=Dno
          GROUP BY  Dname;
```

Specification of Views in SQL (cont'd.)

- Specify SQL queries on a view
- حدد استعلامات SQL في طريقة العرض
- View always up-to-date عرض محدث دائما
- Responsibility of the DBMS and not the user
- مسؤولية نظام إدارة قواعد البيانات وليس المستخدم
- **DROP VIEW** command إسقاط عرض الأمر
- Dispose of a view التخلص من وجهة نظر

View Implementation, View Update, and Inline Views

عرض التنفيذ وعرض التحديث وطرق العرض المضمنة

Complex problem of efficiently
implementing a view for querying
مشكلة
معقدة تتعلق بتنفيذ طريقة عرض للاستعلام بكفاءة

Query modification approach نهج تعديل الاستعلام

Modify view query into a query on underlying base tables
تعديل عرض الاستعلام في استعلام على الجداول الأساسية
الأساسية

Disadvantage: inefficient for views defined via
complex queries that are time-consuming to execute

العيب: غير فعال لطرق العرض المحددة من خلال الاستعلامات المعقدة التي
تستغرق وقتاً طويلاً في التنفيذ

View Implementation

عرض التنفيذ

■ View materialization approach عرض نهج التجسيد

- Physically create a temporary view table when the view is first queried
- إنشاء جدول عرض مؤقت فعليًا عند الاستعلام عن العرض لأول مرة
- Keep that table on the assumption that other queries on the view will follow
- احتفظ بهذا الجدول على افتراض أن الاستفسارات الأخرى في طريقة العرض ستتبع
- Requires efficient strategy for automatically updating the view table when the base tables are updated
- يتطلب إستراتيجية فعالة لتحديث جدول العرض تلقائيًا عند تحديث الجداول الأساسية

View Implementation (cont'd.)

Incremental update strategies ■

استراتيجيات التحديث المتزايد ■

■ DBMS determines what new tuples must be inserted, deleted, or modified in a materialized view table

■ يحدد نظام إدارة قواعد البيانات (DBMS) ما هي المجموعات الجديدة التي يجب إدراجها أو حذفها أو تعديلها في جدول عرض ملموس

View Update and Inline Views

عرض التحديث وطرق العرض المضمنة

- Update on a view defined on a single table without any aggregate functions
- التحديث على طريقة عرض محددة في جدول واحد بدون أي وظائف مجمعة

Can be mapped to an update on underlying base table ■
يمكن تعيينها لتحديث الجدول الأساسي الأساسي

■ View involving joins عرض يتضمن الصلات

■ Often not possible for DBMS to determine which of the updates is intended

■ في كثير من الأحيان لا يمكن لنظام إدارة قواعد البيانات (DBMS) تحديد أي التحديثات هو المقصود

View Update and Inline Views (cont'd.)

■ Clause **WITH CHECK OPTION** بند مع خيار الاختيار

■ Must be added at the end of the view definition

if a view is to be updated يجب إضافتها في نهاية

تعريف العرض إذا كان سيتم تحديث طريقة العرض

■ **In-line view**

■ Defined in the `FROM` clause of an SQL query

■ معرف في عبارة `FROM` لاستعلام SQL

Schema Change Statements in SQL

SQL عبارات تغيير المخطط في

■ Schema evolution commands

■ أوامر تطور المخطط

- Can be done while the database is operational
- يمكن القيام به أثناء تشغيل قاعدة البيانات
- Does not require recompilation of the database schema
- لا يتطلب إعادة تجميع مخطط قاعدة البيانات

The DROP Command

DROP command ■

Used to drop named schema elements, such as ■
tables, domains, or constraint

■ يستخدم لإسقاط عناصر المخطط المسماة ، مثل الجداول أو المجالات
أو القيد

Drop behavior options: ■
خيارات سلوك الإسقاط:

■ CASCADE and RESTRICT
و القيد CASCADE

Example: ■

■ DROP SCHEMA COMPANY CASCADE;

The ALTER Command

Alter table actions include: ■

Adding or dropping a column (attribute) ■

إضافة عمود أو إسقاطه (سمة) ■

Changing a column definition ■ تغيير تعريف العمود

Adding or dropping table constraints ■ إضافة أو إسقاط قيود الجدول

Example: ■

```
ALTER TABLE COMPANY.EMPLOYEE ADD COLUMN Job  
VARCHAR(12); ■
```

To drop a column ■ لإسقاط عمود

Choose either CASCADE or RESTRICT

اختر إما CASCADE أو RESTRICT

The ALTER Command (cont'd.)

- Change constraints specified on a table ■

- تغيير القيود المحددة في الجدول ■

- Add or drop a named constraint ■
إضافة أو إسقاط قيد مسمى

```
ALTER TABLE COMPANY.EMPLOYEE  
DROP CONSTRAINT EMPSUPERFK CASCADE;
```