# **SQL QUERIES**

Aims: At the end of this group of four lectures, you should be able to use SQL to query your database.

#### Data retrieval in SQL

- Single table queries
- Multi-table queries
- Set operations
- Aggregate functions and grouping
- Complex queries

Reading: Elmasri & Navathe, Chapters 6 & 7



### SELECT STATEMENT

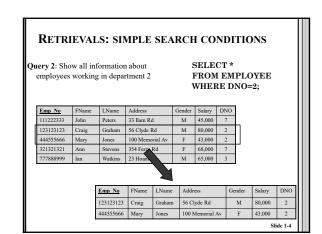
- o Not the same as the SELECT operation of the relational algebra
- o Six clauses only the first two mandatory
- o SELECT <attribute list> SELECT cattribute list>
  FROM 
  [WHERE <condition>]
  [GROUP BY <grouping attributes>]
  [HAVING <group condition>]
  [ORDER BY <attribute list>]
- <attribute list> is a list of attribute names whose values are to be retrieved by the query
- o is a list of the relation names required to process the query o <condition> is a conditional (Boolean) expression that identifies the tuples to be retrieved by the query
- A query is evaluated by first taking the tables in FROM, applying the WHERE clause, then GROUP BY and HAVING, then SELECT clause, and finally ORDER BY

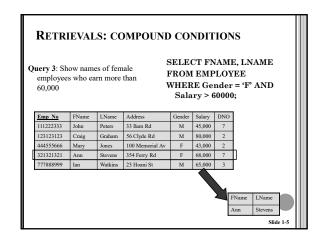


## SIMPLE SQL QUERIES

- ${\bf o}$  Basic SQL queries correspond to using the SELECT, PROJECT, and JOIN operations of the relational
- o Query 1: Show employee numbers only. SELECT EMP\_NO FROM EMPLOYEE;

,								
Emp No	FName	LName	Address	Gender	Salary	DNO	l	EMP NO
111222333	John	Peters	33 Ilam Rd	M	45,000	7		111222333
123123123	Craig	Graham	56 Clyde Rd	M	80,000	2		123123123
444555666	Mary	Jones	100 Memorial Av	F	43,000	2		444555666
321321321	Ann	Stevens	354 Ferry Rd	F	68,000	7		321321321
777888999	Ian	Watkins	23 Hoani St	M	65,000	3		777888999





### SIMPLE SQL QUERIES

- o Query 4: Comparison search condition List the names of all directors born in or after 1920.
- o Query 5: Compound search condition (AND) List the titles and numbers of all movies that have won at least one Academy Award and have been made in or after 1988.



#### SIMPLE SQL QUERIES (CONT)

- Query 6: Compound search condition (OR)
  List the titles of all comedies or dramas.
- Query 7: NOT operator

  List the titles of all movies that have a critics rating.

### ARITHMETIC OPERATIONS

- ${\bf o}$  The standard arithmetic operators +, -, \*, / can be applied to numeric values in an SQL query result
- o Query 8: Calculated fields

 $Produce\ a\ list\ of\ customer\ names,\ numbers\ and\ bonuses.$ 



# Tables as Sets in $\mathbf{SQL}$

- ${\bf o}$  SQL does not automatically eliminate duplicate tuples in query results
- $\bullet$  ALL (default) or DISTINCT in the SELECT clause
- o Query 9:

Produce a list of all stars that acted in movies no 5 or 6.

# SIMPLE SQL QUERIES (CONT)

- Query 10: range search (BETWEEN)
   List the numbers and titles of all movies made between 2000 and 2005.
- Query 11: set membership (IN)

  List the numbers and titles of all movies whose type is comedy or drama.



### PATTERN MATCHING

- o att [NOT] LIKE pattern
- ${\bf o}\ \ {\rm Patterns}\ {\rm are}\ {\rm formed}\ {\rm by}\ {\rm using}\ {\rm two}\ {\rm reserved}\ {\rm characters};$
- % replaces an arbitrary number of characters (wild card)
- \_ replaces a single arbitrary character (position marker)
- o Works with attributes of various data types
- o Query 12: Find all customers who live in Ilam.
- ${\tt o~Query~13:}~ Retrieve~all~directors~who~were~born~during~the~1950s.$
- Escape character: 'Dark\\_Night%' escape '\'



# REGULAR EXPRESSIONS

- o Regular expressions added in SQL3
- ${\tt o} \ \ {\tt REGEXP\_LIKE} \ (att, pattern, \ [param]) \\$
- ${\bf o}\,$  Only string attributes can be used
- o param
  - 'i': case-insensitive
- 'c' : case-sensitive'x' : ignore whitespace
- o Query 14: Find all stars whose last name is Marais or Moranis.



# NULLS IN SQL QUERIES

- o Meanings of NULL
  - Unknown value
  - · Unavailable or withheld value
  - Not applicable attribute
- ${\bf o}\:$  SQL uses IS NULL or IS NOT NULL to compare NULLs because it considers each NULL value distinct from other NULL values, so equality comparison is not appropriate
- Query 15: NULL search condition
   List all directors who are still living.



#### THREE-VALUED LOGIC AND TRUE FALSE UNKNOWN FALSE FALSE FALSE FALSE UNKNOWN UNKNOWN UNKNOWN FALSE FALSE UNKNOWN FALSE TRUE FALSE UNKNOWN UNKNOWN UNKNOWN NOT FALSE TRUE UNKNOWN UNKNOWN

### ORDER BY

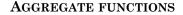
- $\boldsymbol{o}$  Sort the tuples based on the values of some attribute(s)
- $oldsymbol{\circ}$  ASC (ascending, default) or DESC (descending)
- Query 16: Sorting results (ORDER BY)

  Produce a list of all movies, arranged in descending order of the number of Academy Awards won.



# ORDER BY (CONT.)

o Query 17: Multiple column ordering
List the numbers, names, addresses and join dates of
all members. Sort the output by last name descending
and by first name ascending.



- $\circ \mathit{SUM}, \mathit{MIN}, \mathit{MAX}, \mathit{AVG}, \mathit{COUNT}$
- NULL values ignored by aggregate functions
   except COUNT(\*)
- o ALL and DISTINCT
- o The SELECT clause may contain:
  - Column names
  - Aggregate functions
  - Constants
  - Expressions containing the above



### AGGREGATE FUNCTIONS: EXAMPLES

- Query 18: How many movies won more than four Academy Awards?
- Query 19: Find how many comedies there are and how many AA they won



#### AGGREGATE FUNCTIONS: EXAMPLES

• Query 20: How many types of movies are there in the database?

#### GROUPING

- f o Often we want to apply the aggregate functions to groups of tuples in a relation
- Each group of tuples consists of the set of tuples that have the same value for the grouping attribute(s)
- o The function is applied to each group independently
- ${\bf \circ}$  GROUP BY clause specifies the grouping attributes
- Query 21: Find the number of movies in each category and the total of AA won in each of them.



### GROUPING (CONT.)

- Query 22: Show how many movies of each type each director has made.
- Query 23: For each director show how many different types of movies he/she has directed.



# HAVING CLAUSE

- Sometimes we want to retrieve the values of these functions for only those groups that satisfy certain conditions
- o The HAVING clause is used for specifying a selection condition on groups (rather than on individual tuples)
- Query 24: For each director, list the director's number and the total number of awards won by comedies he or she directed if that number is greater than 1.



#### NESTING OF QUERIES

- o A complete SELECT query, called a *nested query* (a subquery), can be specified within the WHERE clause of another query (the outer query)
- Query 25: List the numbers and names of all members who have rented more DVDs than average.



### NESTED QUERIES

- Query 26: List the titles of all movies directed by Stanley Kubrick.
- Query 27: Show the names of all stars who acted in movie no 137.



#### RULES FOR NESTED QUERIES

- ${\bf o}$  ORDER BY not permitted in the nested query.
- The subquery SELECT list must consist of a single attribute or expression, except for queries using EXISTS.
- o By default, attribute names in a subquery refer to the table names in the FROM clause of the subquery.
- A subquery can appear on both sides of a comparison, but some DBMSs allow nested queries only on the right-hand side.
- ${\bf o}$  A subquery may not be used as an operand in an expression.
- $oldsymbol{\circ}$  In general, there can be several levels of nesting.



#### CORRELATED NESTED QUERIES

- o If a condition in the WHERE-clause of a *nested* query references an attribute of a relation declared in the *outer query*, the two queries are said to be *correlated*
- The result of a correlated nested query is different for each tuple (or combination of tuples) of the relation(s) the outer query



#### CORRELATED NESTED QUERIES (CONT)

• Query 28: For each director, his/her number and the title of his/her movie that got most Academy Awards.



# **EXISTS**

- o EXISTS is used to check whether the result of a correlated nested query is empty (contains no tuples) or not
- Query 29: Find the numbers and names of all directors who have directed at least one comedy.



#### EXISTS (CONT.)

• Query 30: Find the names of directors who made no dramas.



### MULTITABLE QUERIES

- A query containing a nested query and using the = or IN conditions can *always* be expressed as a single query.
- ${\tt o}$  In that case, the FROM clause of the query may contain more than one table
- ${\bf o}$  If there is more than one table in FROM, join conditions are necessary
- o Query 26 revisited:

List the titles of all movies directed by Stanley Kubrick.



#### UNSPECIFIED WHERE CLAUSE

- o A missing WHERE-clause indicates no condition
- o All tuples of the relations in the FROM clause are selected
- This is equivalent to the condition WHERE TRUE
- If more than one relation is specified FROM and there is no join condition, then the CARTESIAN PRODUCT of tuples is selected
- It is extremely important not to overlook specifying any selection and join conditions in the WHERE clause; otherwise, incorrect and very large relations may result
- Query 31: Select all combinations of movie titles and DVD codes.



#### MULTIPLE TABLE JOIN

- ${\bf o}$  The join condition may be specified in WHERE
- SQL2 also allows joins to be specified in FROM FROM table1 join-type JOIN table2 ON att1=att2
- o Query 26 revisited:

List the titles of all movies directed by Stanley Kubrick.



#### MULTIPLE TABLE JOIN

• Query 32: List the names and addresses of all customers currently renting Mel Gibson's movies.



#### ALIASES

- o Some queries need to refer to the same relation twice
- o In this case, an *alias* (i.e. alternative name) is given to the relation
- o Can also use the AS keyword to specify aliases
- Aliasing can also be used in any SQL query for convenience
- Query 33: Find the list of any pairs of stars who have the same last name.



### SPECIFYING JOINS IN SQL2

- o FROM table1 join-type JOIN table2 ON att1=att2
- o Join-type: join, natural join, left outer join, right outer join, full outer join, cross join
- Outer join in Oracle:
  - Additionally, the (+) operator can be used
  - Use (+) after the join attribute which might need NULLs to be added
- Query 34: List names and numbers of all members. For those who are currently renting DVDs, list the number of DVDs too.



## JOINED RELATIONS IN SQL2

- ${\bf \circ}\ {\rm Can}\ {\rm specify}\ {\rm a}$  "joined relation" in the FROM clause
- ${\bf o}\$  Looks like any other relation but is the result of a join
- Query 26 revisited: List the titles of all movies directed by Stanley Kubrick.

SELECT TITLE

FROM MOVIE, DIRECTOR
WHERE DIRECTOR=DNUMBER and LNAME='Kubrick'
and FNAME='Stanley'

can be written as:

SELECT TITLE

FROM (select \* from MOVIE join DIRECTOR on DIRECTOR=DNUMBER) MOVDIR

WHERE movdir.lname='Kubrick' and movdir.fname='Stanlegs'



### SET OPERATIONS

- o UNION
- o EXCEPT (MINUS in Oracle)
- o INTERSECT
- o The resulting relations of these set operations are sets of tuples; duplicate tuples are eliminated from the result
- ${\bf o}$  Corresponding multiset operations: UNION ALL, EXCEPT ALL, INTERSECT ALL
- ${\bf o}$  Arguments must be  $union\ compatible\ relations$



# **SET OPERATIONS (CONT)**

 Query 35: List the DVD codes and movie numbers for all DVDs on which the movie is a comedy or that are rented by Mark Haley.



# **SET OPERATIONS (CONT)**

• Query 36: Find the numbers of directors who have directed both comedies and dramas.



# **SET OPERATIONS (CONT)**

• Query 37: Find the numbers of directors who have directed comedies, but not dramas.

# ANY/ALL PREDICATE

- ${\bf o}$  Used with a comparison operator
- o In some DBMSs, SOME used instead of ANY
- ${\bf o}$ !=ALL is equivalent to NOT IN
- o = ANY is equivalent to IN
- Query 38: List numbers and titles of all movies that were nominated for more awards than any movie directed by Woody Allen.



### **DIVISION IN SQL**

• Query 39: Find the names of directors who have directed at least one movie of each type.



# DUAL TABLE

- o Available to all users
- $\boldsymbol{o}$  Has one column (dummy, varchar(1)) and one row with a value  $\boldsymbol{X}$
- $oldsymbol{\circ}$  Useful for getting a constant expression Select sysdate from dual;

