DATABASE SYSTEM LABS
- BASIC SQL III

李旭东

LEEXUDONG@NANKAI.EDU.CN

NANKAI UNIVERSITY

OBJECTIVES

- Basic Queries
- Natural join
- Rename operation
- String operation
- Ordering the display of tuples
- Set operation
- Aggregate Functions
- Nested Subqueries

Basic Queries

BASIC QUERIES

- select * from course;
- select name

  from instructor

  where dept_name = ‘Comp. Sci.’;
- select name, course_id

  from instructor , teaches

  where instructor.ID = teaches.ID

Natural join

NATURAL JOIN

- select * from teaches natural join course;
- select * from teaches join course on teaches.course_id = course.course_id;

Rename operation

NATURAL JOIN

- select avg(budget) as avg_budget from department;

- select distinct T.name
  from instructor as T, instructor as S
  where T.salary > S.salary and S.dept_name = ‘Comp. Sci.’

String operation

STRING OPERATION

- select name from instructor where name like '%on%’
- select * from department where dept_name like '_in%'

Ordering the display of tuples

ORDERING THE DISPLAY OF TUPLES

- select distinct name
  from instructor
  order by name
- select *
  from instructor
  order by salary desc;

Set operation

SET OPERATION

– UNION

- Find courses that ran in Fall 2009 or in Spring 2010

  (select course_id from section where semester = ‘Fall’ and year = 2009)

  union

  (select course_id from section where semester = ‘Spring’ and year = 2010)

SET OPERATION

– INTERSECT

- Find courses that ran in Fall 2009 and in Spring 2010

  (select course_id from section where semester = ‘Fall’ and year = 2009)

  intersect

  (select course_id from section where semester = ‘Spring’ and year = 2010)

SET OPERATION

– EXCEPT

- Find courses that ran in Fall 2009 but not in Spring 2010

  (select course_id from section where semester = ‘Fall’ and year = 2009)

  except

  (select course_id from section where semester = ‘Spring’ and year = 2010)

Aggregate Functions

AGGREGATE FUNCTIONS

- select dept_name, avg (salary)
  from instructor
  group by dept_name
  having avg (salary) > 42000;

Nested Subqueries

NESTED SUBQUERIES

- select count(distinct ID)
  from takes
  where (course_id, sec_id, semester, year)
  in (select course_id, sec_id, semester, year
  from teaches
  where teaches.ID= 43779);

THANKS !

leexudong@nankai.edu.cn