Class Number Name Data

**Experiment topic：**

Experiment 3 Simple queries (1)

1. **Experimental objectives**

1）Master the solution of attaching a database into the lab environment.

2）Master the method of using SELECT.

3）Master the methods of managing simple queries.

1. **Experimental tasks**
   1. Experimental contents

4.1 Attaching the projemp database

1) Download the projemp database file (projemp.mdf) from the Lab Resources

folder on Blackboard to your download folder.

2) Copy & Paste projemp.mdf from your download folder to

C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\MSSQL\DATA

In general, always use ‘Copy & Paste’ when moving database files - do NOT use

‘Drag & Drop’. If necessary, click ‘continue’ if you are asked for administrator

permissions.

3) Open SQL Server Management Studio (SSMS) and connect to the server.

4) In Object Explorer, right click on Databases, and click ‘Attach…’ on the

pop-up menu.

5) Click ‘Add…’ on the dialogue box, select projemp.mdf when the list of

available databases appears, then click ‘OK’ on both dialogue boxes to

attach the database.

6) Expand the Databases tab in Object Explorer. projemp should now be

attached and available for use.

7) Use database file projemp on SQL Server - Execute USE projemp; in a New

Query window

Click on New Query on the top menu to create a new query.

USE projemp;

GO Placed before each query ensures the correct database is current4.2 Complete the following queries using SELECT clause

1) Display all the records in the EXP table;

2) Get the employee names and ages of those employees in department number

'd1'. (11 records)

3) Get the employee number and name of all employees aged above 30 in

department number ‘d2’ (1 records)

NOTE: Values of a character datatype attribute (e.g., dno) must be shown in

inverted commas (single not double), whereas numeric datatype attributes are not.

4) Get the names of employees earning less than £20,000 or more than £30,000.

(13 records)

5) How many employees are in department number 'd3'? (1 records)

6) Get the total salary of all employees in department number ‘d1’ (adds up all

salaries in department ‘d1’ ) (1 records)

7) Get the employee names and their salaries in ascending order of salary.

NOTE: ASC is the default value if ASC or DESC is not entered; DESC sorts in

descending order

Try changing ASC to DESC and changing the ORDER BY attribute to ename

8) Get the employees who have salaries of £18,000 or £20,000 (Use of the IN

predicate) (2 records)

9) Get the names of employees who earn a salary above £25,000. (10 records)

10) Get the names of employees who earn a salary above £25,000 and are

in department number 'd1'. (6 records)

11) What is the average age of employees in department number 'd2'?(29)

12) List the names, salaries and ages of all employees in department number

'd1’ in descending order of age (oldest to youngest).(11 records)

13) How many employees have a salary greater than £20,000 and are in

department number 'd1?'(9)

2.2 Experimental preparation

Review the SQL main clauses that will be used in this experiment, and write down the right way to use the clauses in the examples.

SELECT clause: used to select the columns or expressions to be retrieved.

From clause: Specify the table or view from which to retrieve data.

WHERE clause: used to filter data that meets the specified conditions.

ORDER BY clause: used to sort the results by the specified column, default to ascending order (ASC).

IN clause: used to specify a list of values for matching in the query.

AVG function: used to calculate the average value.

DESC keyword: used to sort results in descending order.

Example of a SELECT clause:

1. SELECT column1, column2, ...

2. FROM table\_name;

3.

Example from clause:

1. SELECT column1, column2, ...

2. FROM table\_name

3. WHERE condition;

4.

Example of a WHERE clause:

1. SELECT column1, column2, ...

2. FROM table\_name

3. WHERE condition;

4.

Example of an ORDER BY clause:

1. SELECT column1, column2, ...

2. FROM table\_name

3. ORDER BY column\_name ASC|DESC;

4.

Example of IN clause:

SELECT column1, column2, ...

FROM table\_name

WHERE column\_name IN (value1, value2, ...);

AVG function example:

SELECT AVG(column\_name)

FROM table\_name;

DESC keyword example:

SELECT column1, column2, ...

FROM table\_name

ORDER BY column\_name DESC;

1. **Task solutions**

**Analyzes the tasks in the task book, and gives the SQL statements of all the tasks.**

SELECT \* FROM EMP;

SELECT ename, age

FROM EMP

WHERE dno = 'd1';

SELECT eno, ename

FROM EMP

WHERE dno = 'd2' AND age >= 30;

SELECT ename

FROM EMP

WHERE salary < 20000 OR salary > 30000;

SELECT COUNT(\*)

FROM EMP

WHERE dno = 'd3';

SELECT SUM(salary)

FROM EMP

WHERE dno = 'd1';

SELECT ename, salary

FROM EMP

ORDER BY salary ASC;

-- 8)

SELECT \*

FROM EMP

WHERE salary IN (18000, 20000);

-- 9)

SELECT ename

FROM EMP

GROUP BY ename

HAVING SUM(salary) > 25000;

-- 10)

SELECT ename

FROM EMP

WHERE salary > 25000 AND dno = 'd1';

**。。。差3个（11 12 13）**