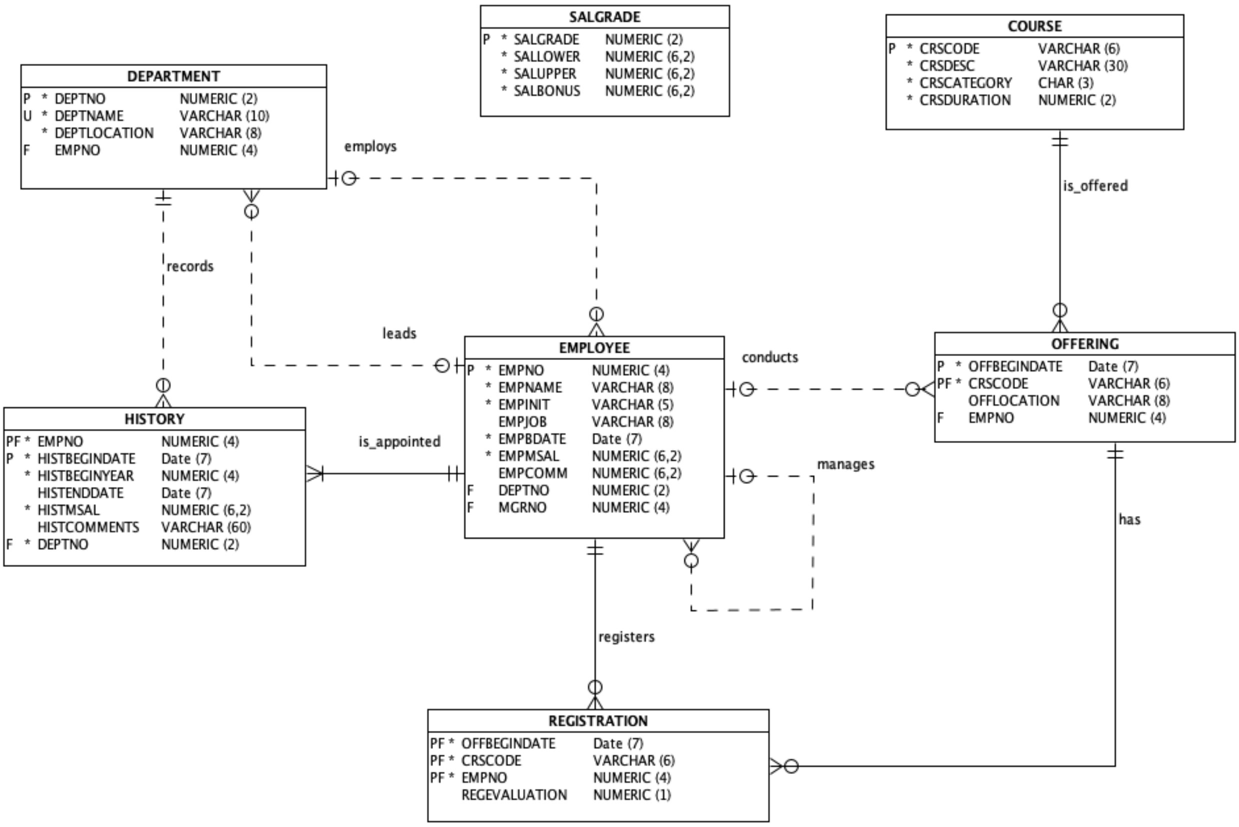
SQL FINAL



```
/*
1. Display the full details of employees who earn less than $1000, order
the output  by  the employee number.
*/

SELECT
    empno,
    empname,
    empinit,
    empjob,
    to_char(empbdate,'dd Mon yyyy'),
    empmsal,
    empcomm,
    deptno,
    mgrno
FROM
    payroll.employee
WHERE
    empmsal < 1000
ORDER BY
    empno;
```

```sql
/*
2. List the department number of departments that have employees, order
the output by the department number .
*/

SELECT DISTINCT
    deptno
FROM
    payroll.employee
ORDER BY deptno;
```

```sql
/*
3. Display the full details of trainers (employee job is TRAINER) who earn
less than 2500 each month and are working in department 20. Order the
output by  employee number.
*/

SELECT
    empno,
    empname,
    empinit,
    empjob,
    to_char(empbdate,'dd Mon yyyy'),
    empmsal,
    empcomm,
    deptno,
    mgrno
FROM
    payroll.employee
WHERE
    upper(empjob) = 'TRAINER'
    AND empmsal < 2500
    AND deptno = 20
ORDER BY
    empno;
```

```sql
/*
4. Display the name, job, monthly salary and commission of employees whose
monthly salary is higher than their commission. Rename the columns: Name,
Job, Monthly Salary and Commission. Order the output by name and within
name by descending monthly salary.
*/

SELECT
    empname  AS "Name",
    empjob   AS "Job",
    empmsal  AS "Monthly Salary",
    empcomm  AS "Commission"
FROM
    payroll.employee
WHERE
    empmsal > empcomm
ORDER BY
    empname,
    empmsal DESC;
```

```
/*
5. Display the name and job of employees whose job ends with the letter R.
Order the output by name and within name by job.
*/

SELECT
    empname,
    empjob
FROM
    payroll.employee
WHERE
    upper(empjob) LIKE '%R'
ORDER BY
    empname,
    empjob;
```

```
/*
6. Display the name and job of employees that have a name starting with
“J”, “K” or “M”. Order the output by name and within name by job.
*/

SELECT
    empname,
    empjob
FROM
    payroll.employee
WHERE
        upper(empname) LIKE 'J%'
    OR
        upper(empname) LIKE 'K%'
    OR
        upper(empname) LIKE 'M%'
ORDER BY
    empname,
    empjob;
```

```sql
/*
7. Display the name, job, date of birth and monthly salary of employees
who were born before 1990 and earn more than 1500 each month. Order the
output by name and within name by monthly salary.
*/

SELECT
    empname,
    empjob,
    to_char(empbdate, 'Dd-Mon-YYYY') AS birthdate,
    empmsal
FROM
    payroll.employee
WHERE
        empbdate < TO_DATE('01-JAN-1990', 'dd-MON-YYYY')
    AND empmsal > 1500
ORDER BY
    empname,
    empmsal;
```

```sql
/*
8. Display the name, job, date of birth and salary of employees that don't
have a commission. Order the output by name, and within name by date of
birth.
*/

SELECT
    empname,
    empjob,
    to_char(empbdate, 'Dd-Mon-YYYY') AS birthdate,
    empmsal
FROM
    payroll.employee
WHERE
    empcomm IS NULL
ORDER BY
    empname,
    empbdate;
```

```sql
/*
9. Display the employee name, job, department name, location and monthly
salary of employees that work in NEW YORK. Order the output by name, and
within name by job.
*/

SELECT
```

```
    e.empname,
    e.empjob,
    d.deptname,
    d.deptlocation,
    e.empmsal
FROM
    payroll.employee e join payroll.department d on e.deptno = d.deptno
WHERE
    upper(d.deptlocation) = 'NEW YORK'
ORDER BY e.empname;
```

```
/*
10. Display the name and job of employees who do not work in NEW YORK or
CHICAGO. Order the output by name, and within name by job.
*/

SELECT
    e.empname,
    e.empjob,
    d.deptname,
    d.deptlocation
FROM
         payroll.employee e
    JOIN payroll.department d
    ON e.deptno = d.deptno
WHERE
    upper(d.deptlocation) NOT IN ( 'NEW YORK', 'CHICAGO' )
ORDER BY
    e.empname,
    empjob;
```

```
/*
11. Display the name, job, date of birth and salary of employees who were
born in the first half of the 70s. Display the output in birth date order
and within birthdate by name.
*/

SELECT
    empname,
    empjob,
    to_char(empbdate, 'Dd-Mon-YYYY') AS birthdate,
    empmsal
FROM
    payroll.employee
WHERE
    empbdate BETWEEN TO_DATE('01-JAN-1970', 'DD-MON-YYYY') AND
TO_DATE('31-DEC-1974',
    'DD-MON-YYYY')
ORDER BY
    empbdate,
    empname;
```

```
/*
12.  Display the name, job and salary of employees who earn less than 1500
or greater than 3000 per month. Order the output by name, and within name
by monthly salary.
*/

SELECT
    empname,
    empjob,
    empmsal
FROM
    payroll.employee
WHERE
    empmsal NOT BETWEEN 1500 AND 3000
ORDER BY
    empname,
    empmsal;
```

```
/*
13. Display the name, job and manager number of employees who have a
manager.
Order the output by manager number and within a given manager by employee
name.
*/

SELECT
    empname,
    empjob,
    mgrno
FROM
    payroll.employee
WHERE
    mgrno IS NOT NULL
ORDER BY
    mgrno,
    empname;
```

```
/*
14. Display the name, job, department name, department location and
monthly salary of employees who either work in DALLAS or as a MANAGER, and
earn more than 2500. Order the output by name and within name by monthly
```

```
salary.
*/

SELECT
    e.empname,
    e.empjob,
    d.deptname,
    d.deptlocation,
    e.empmsal
FROM
         payroll.employee e
    JOIN payroll.department d
    ON ( e.deptno = d.deptno )
WHERE
    ( upper(d.deptlocation) = 'DALLAS'
      OR upper(e.empjob) = 'MANAGER' )
    AND e.empmsal > 2500
ORDER BY
    e.empname,
    empmsal;
```

```
/*
15. Display the name, job, monthly salary and salary grade of all
employees. Display the list in monthly salary order within salary grade
order.
*/

SELECT
    e.empname,
    e.empjob,
    e.empmsal,
    s.salgrade
FROM
    payroll.employee e
    JOIN payroll.salgrade s ON (
        e.empmsal BETWEEN s.sallower AND s.salupper
    )
ORDER BY
    s.salgrade,
    e.empmsal;
```

```
/*
16. Display the name and location of ALL departments, and the name of
their employees. Display the output in employee name order within
department name order.
*/
```

```
SELECT
    d.deptname,
    d.deptlocation,
    e.empname
FROM
    payroll.employee e
    RIGHT OUTER JOIN payroll.department d ON (
        e.deptno = d.deptno
    )
ORDER BY
    d.deptname,
    e.empname;
```

```
/*
17. Display the name of ALL employees, their job and the name of their
manager. List the output in employee name order within manager name order.
*/

SELECT
    e.empname "Employee",
    e.empjob AS "Emp Job",
    m.empname AS "Manager"
FROM
    payroll.employee e
    LEFT OUTER JOIN payroll.employee m ON (
        e.mgrno = m.empno
    )
ORDER BY
    m.empname,
    e.empname;
/*
-- here NVL(m.empname,'NA') could be used to display a value rather than
leave the managers name null for the director
*/
```

```
/*
18. For each employee display their employment history. In the listing
include the employees’ name, the name of the department they worked for,
the begin and end date and their monthly salary. Display the output in
begin date order (most recent at the top of the list) within employee name
order.
*/

SELECT
    e.empname,
    d.deptname,
    to_char(histbegindate,'DD-Mon-YYY') as HISTBEGIN,
    to_char(histenddate,'Dd-Mon-YYYY') as HISTEND,
    h.histmsal
FROM
    payroll.employee e
    JOIN payroll.history h ON (
        e.empno = h.empno
    )
    JOIN payroll.department d ON (
        d.deptno = h.deptno
    )
ORDER BY
    e.empname,
    h.histbegindate DESC;
```

```
/*
19. Display the employee name, empjob, monthly salary and annual salary of
all employees (annual salary is monthly salary x 12). Order the output by
annual salary with the highest value shown first and within annual salary
by name.
*/

SELECT
    empname,
    empjob,
    empmsal,
    empmsal * 12 AS "Annual Salary"
FROM
    payroll.employee
ORDER BY
    "Annual Salary" DESC,
    empname;
```

```
/*
20. Display the employee name, empjob, monthly salary, empcommission and
annual income (salary and empcommission) of all employees. Commission is
paid on a monthly basis. Order the output by name,and within by annual
income.
*/

SELECT
    empname,
    empjob,
    empmsal,
    empcomm,
    12 * ( empmsal + nvl(empcomm, 0) ) AS "Annual Income"
FROM
    payroll.employee
```

```
ORDER BY
    empname,
    "Annual Income";
```

```
/*
1. Display for all employees their number, name, job, monthly salary,
their current annual salary (not including commission) and what their
annual salary would be if they were given a 10% pay rise. Order the output
by employee number.
*/

SELECT
    empno,
    empname,
    empjob,
    empmsal,
    empmsal * 12          AS annual_salary,
    empmsal * 12 * 1.1    AS ann_sal_with_rise
FROM
    payroll.employee
ORDER BY
    empno;
```

```
/*
2. Display the name of all employees, their birthdate and their age in
years. Order the output by birthdate, within birthdate order the output by
name.
*/

SELECT
    empname,
    to_char(empbdate, 'DD-Mon-YYYY')                       AS birthdate,
    floor(months_between(sysdate, empbdate) / 12)          AS "Age in Yrs"
FROM
    payroll.employee
ORDER BY
    empbdate,
    empname;
```

```
/*
3. Display for all employees, their number, name, job, monthly salary,
commission (which is paid monthly) and their current annual salary
(including commission). Order the output by  employee number.
*/

SELECT
```

```
    empno,
    empname,
    empjob,
    empmsal,
    empcomm,
    ( empmsal + nvl(empcomm, 0) ) * 12 AS total_salary
FROM
    payroll.employee
ORDER BY
    empno;
```

```
/*
4. Display all employees details in the following format: EMPLOYEE N.
Smith IS A Trainer AND WORKS IN THE Training DEPARTMENT.

INITCAP() 函数将每个单词的第一个字母设置为大写，所有其他字母设置为小写。单词由空格或
非字母数字字符分隔。

SELECT INITCAP('steven king') "Emp. Name"
FROM DUAL;

>> Emp. Name
-----------
Steven King
*/
SELECT
    'EMPLOYEE '
     || empinit
     || '. '
     || initcap(empname)
     || ' IS A '
     || initcap(empjob)
     || ' AND WORKS IN THE '
     || initcap(deptname)
     || ' DEPARTMENT.' as empdetails
FROM
    payroll.employee e
    JOIN payroll.department d ON (
        e.deptno = d.deptno
    )
ORDER BY empname;

SELECT
    'EMPLOYEE '
     || empinit
     || '. '
     || initcap(empname)
     || ' IS A '
     || initcap(nvl(empjob,'----'))
```

```
        || ' AND WORKS IN THE '
        || initcap(nvl(deptname,'----'))
        || ' DEPARTMENT.' as empdetails
FROM
    payroll.employee e
    left outer JOIN payroll.department d ON (
        e.deptno = d.deptno
    )
ORDER BY e.empno;
```

```
/*
5. Display the name of all employees, their birthdate and their age in
months. Order the output by age in months (with the oldest employee
first), within age in months order the output by name. The age in months
must be shown with one decimal point and right aligned.
*/

SELECT
    empname,
    to_char(empbdate, 'dd-Mon-yyyy')
AS birthdate,
    lpad(to_char(months_between(sysdate, empbdate), '990.0'), 10)
AS age_in_months
FROM
    payroll.employee
ORDER BY
    age_in_months DESC,
    empname;

/*
LPAD() 函数用另一个字符串向左填充一个字符串，达到一定的长度。
SELECT LPAD("SQL Tutorial", 20, "ABC");
>> ABCABCABSQL Tutorial
*/
```

```
/*
6. Display the employees who were born in February. Order the output by
employeename.


EXTRACT() Extract the month from a date:

SELECT EXTRACT(MONTH FROM "2017-06-15");
>> 6
*/

SELECT
```

```
    empname,
    to_char(empbdate, 'dd-Mon-yyyy') AS birthdate
FROM
    payroll.employee
WHERE
    EXTRACT(MONTH FROM empbdate) = 2
ORDER BY
    empname;
```

```


/*
7. Display the employee name, salary and commission (using the GREATEST
function) for those employees who earn more commission than their monthly
salary. Order the output by employee name and within name by monthly take
home pay (salary plus commission).

GREATEST() Return the greatest value of the list of arguments:
*/

SELECT
    empname,
    empmsal,
    empcomm
FROM
    payroll.employee
WHERE
    empcomm = greatest(empcomm, empmsal)
ORDER BY
    empname,
    ( empmsal + empcomm );
```

```
/*
8. Display the name of all employees and their birthdate in the following
format: EMPLOYEE N. Smith was born on FRIDAY the 17 of DECEMBER , 1982.
Order the output by employee name.

 INITCAP() 函数将每个单词的第一个字母设置为大写，所有其他字母设置为小写。单词由空格或
非字母数字字符分隔

RTRIM() 从字符串中删除尾随空格：

SELECT RTRIM('SQL Tutorial    ') AS RightTrimmedString;
>>SQL Tutorial
*/

SELECT
```

```
    'EMPLOYEE '
     || empinit
     || '. '
     || initcap(empname)
     || ' was born on '
     || rtrim(TO_CHAR(empbdate,'DAY') )
     || ' the '
     || EXTRACT(DAY FROM empbdate)
     || ' of '
     || rtrim(TO_CHAR(empbdate,'MONTH') )
     || ','
     || EXTRACT(YEAR FROM empbdate) as "EMP DOB Details"
FROM
    payroll.employee
ORDER BY
    empname;
```

```
/*
9. Display the name of the employees who have registered for a course and
the
number of times they have registered. Order the output by employee number.

`GROUP BY`语句将具有相同值的行分组到汇总行中，例如“查找每个国家/地区的客户数量”。

`GROUP BY`语句通常与聚合函数 ( `COUNT()`, `MAX()`, `MIN()`, `SUM()`, `AVG()`)
一起使用，以按一列或多列对结果集进行分组
SELECT _column_name(s)_
FROM _table_name_
WHERE _condition_
GROUP BY _column_name(s)
ORDER BY _column_name(s);_

SELECT COUNT(CustomerID), Country
FROM Customers
GROUP BY Country
ORDER BY COUNT(CustomerID) DESC;

>>COUNT(CustomerID) Country
13USA

11Germany

11France

*/

SELECT
    e.empno,
    e.empname,
    COUNT(*) AS nbr_registrations
FROM
         payroll.employee e
    JOIN payroll.registration r
    ON ( e.empno = r.empno )
GROUP BY
    e.empno,
    e.empname
ORDER BY
    empno;
```

```
/*
10. Who is the oldest employee? Show the employee number, name and date of
birth. Order the output by employee number.
*/

SELECT
    empno,
    empname,
    to_char(empbdate, 'DD-Mon-YYYY')
FROM
    payroll.employee
WHERE
    empbdate = (
        SELECT
            MIN(empbdate)
        FROM
            payroll.employee
    )
ORDER BY
    empno;
```

```
/*
11. For each department, list the department number and name, the number
of employees, the minimum and maximum monthly salary, the total monthly
salary and the average salary paid to their employees. Name the columns:
NbrOfEmployees, MinSalary, MaxSalary, TotalSalary, AvgSalary. Order the
output by department number.
*/

SELECT
    d.deptno,
    d.deptname,
    COUNT(e.empno)                           "NbrOfEmployees",
    MIN(empmsal)                             "MinSalary",
    MAX(empmsal)                             "MaxSalary",
    SUM(empmsal)                             "TotalSalary",
    to_char(AVG(empmsal), '9999.99')          "AvgSalary"
FROM
    payroll.employee      e
    RIGHT OUTER JOIN payroll.department    d
    ON ( e.deptno = d.deptno )
GROUP BY
    d.deptno,
    d.deptname
ORDER BY
    deptno;
```

```
/*
12. Display the department number, jobs available in that department and
the total monthly salary paid for each job. Order the output by department
number and within a department by job.
*/

SELECT
    deptno,
    empjob,
    SUM(empmsal)
FROM
    payroll.employee
GROUP BY
    deptno,
    empjob
ORDER BY deptno,empjob;
```

```
/*
13. Which employee earns more than the average salary? Show the employee
number, name and monthly salary. Order the output by employee number.
*/

SELECT
    empno,
    empname,
    empmsal
FROM
    payroll.employee
WHERE
    empmsal > (
        SELECT
            AVG(empmsal)
        FROM
            payroll.employee
    )
```

```
ORDER BY
    empno;
```

```
/*
14. Which department has the greatest average monthly salary? Show the
department no, name and average monthly salary. Order the output by
department number.

ROUND() 函数将数字四舍五入到指定的小数位数
SELECT ROUND(235.415, 2) AS RoundValue;
>>RoundValue
235.420
*/

SELECT
    d.deptno,
    d.deptname,
    round(AVG(empmsal),2) "AvgSalary"
FROM
    payroll.employee e
    JOIN payroll.department d ON (
        e.deptno = d.deptno
    )
GROUP BY
    d.deptno,
    d.deptname
HAVING
    AVG(empmsal) = (
        SELECT
            MAX(AVG(empmsal) )
        FROM
            payroll.employee
        GROUP BY
            deptno
    )
ORDER BY
    deptno;
```

```
/*
15. Which course has the most offerings? Show the course code, description
and number of offerings. Order the output by the number of offerings in
descending order and within the number of offerings by the course code.
*/

SELECT
    c.crscode,
    c.crsdesc,
    COUNT(*) "NbrOfferings"
FROM
         payroll.course c
    JOIN payroll.offering o
    ON ( c.crscode = o.crscode )
GROUP BY
    c.crscode,
    c.crsdesc
HAVING
    COUNT(*) = (
        SELECT
            MAX(COUNT(*))
        FROM
            payroll.offering
        GROUP BY
            crscode
    )
ORDER BY
    COUNT(*) DESC,
    crscode;
```

```
/*
16. Display the name, job and date of birth of employees who perform the
same job as SCOTT and were born in the same year. Do not include SCOTT in
the output. Order the output by employee name.
*/

SELECT
    e.empname,
    e.empjob,
    to_char(empbdate, 'dd-Mon-yyyy')
FROM
    payroll.employee e
WHERE
        ( e.empjob,
          EXTRACT(YEAR FROM empbdate) ) = (
            SELECT
                e.empjob,
                EXTRACT(YEAR FROM empbdate)
            FROM
                payroll.employee e
            WHERE
                upper(empname) = 'SCOTT'
        )
    AND upper(e.empname) <> 'SCOTT'
ORDER BY
    empname;
```

```
/*
17. Using the MINUS statement, which employees have never registered in a
course. Show their employee number and name. Order the output by employee
number.

`SELECT DISTINCT`语句用于仅返回不同（不同）的值。
在表中，一列通常包含许多重复值；有时您只想列出不同的（不同的）值。
SELECT DISTINCT _column1_, _column2, ..._
FROM _table_name_;
*/

SELECT
    empno,
    empname
FROM
    payroll.employee
MINUS
SELECT DISTINCT
    e.empno,
    empname
FROM
        payroll.registration r
    JOIN payroll.employee e
    ON ( r.empno = e.empno )
ORDER BY
    empno;
```

```
/*
18. Using the INTERSECT statement, which employees have both registered
for and conducted courses. Show the employee number and name. Order the
output by employee number.
*/

SELECT DISTINCT
    e.empno AS "TrainerAndStudent",
    empname
FROM
        payroll.offering o
    JOIN payroll.employee e
    ON ( o.empno = e.empno )
INTERSECT
SELECT DISTINCT
    e.empno,
    empname
FROM
        payroll.registration r
    JOIN payroll.employee e
    ON ( r.empno = e.empno )
```

```
ORDER BY
    "TrainerAndStudent";
```