## Part II B

**Q.1**

```
SQL> SELECT E.fname || ' ' || E.lname "Employee", S.name "Skill",
  2         COUNT(T.emp_num) "Times Trained",MIN(T.date_acquired) "Date",
  3         TRUNC(MONTHS_BETWEEN(SYSDATE,MAX(T.date_acquired)),0) "Month Passed"
  4  FROM employee E JOIN training T ON E.emp_num = T.emp_num
  5                  JOIN skill S    ON T.code = S.code
  6  GROUP BY E.fname, E.lname, S.name;
```

| Employee | Skill | Times Trained | Date | Month Passed |
|---|---|---|---|---|
| Lebron Jason | Python | 1 | 01-NOV-21 | 1 |
| David Davis | Decision Making | 1 | 01-DEC-21 | 0 |
| Lawrence Curry | Writing | 1 | 01-JUN-20 | 18 |
| Mike Jordan | Speaking | 1 | 01-JAN-15 | 83 |
| Klay Durant | C++ | 1 | 01-DEC-21 | 0 |
| Kris Paul | Java | 2 | 01-DEC-20 | 0 |
| John Magic | Java | 1 | 01-FEB-12 | 118 |
| Ray Tatum | Java | 1 | 01-AUG-19 | 28 |
| Koby Bryant | C++ | 1 | 01-OCT-12 | 110 |
| Frank Lee | Java | 1 | 01-DEC-21 | 0 |
| Matt Lopez | Speaking | 1 | 01-DEC-21 | 0 |
| Derek Wade | Leadership | 1 | 01-MAY-19 | 31 |
| Stephen Book | Python | 1 | 01-DEC-21 | 0 |
| Chris Bosh | Decision Making | 1 | 01-JUL-21 | 5 |
| Paris Harden | Decision Making | 1 | 01-JUL-20 | 17 |
| Kevin Thompson | Java | 1 | 01-DEC-21 | 0 |
| Kyle Anderson | Speaking | 1 | 01-DEC-21 | 0 |

```
17 rows selected.
```

**Q.2**

```
SQL> SELECT LEVEL,
  2        LPAD(' ', 3*(LEVEL - 1))|| E.emp_num || ' ' || E.fname || ' ' || E.lname "Employee",
  3         D.name Department
  4  FROM employee E JOIN department D USING (dept_code)
  5  START WITH E.emp_num = '1000'
  6  CONNECT BY PRIOR E.emp_num = E.super_ID;
```

```
     LEVEL Employee                       DEPARTMENT
---------- ------------------------------ --------------
         1 1000 Steve King                Executive
         2    1001 Mike Jordan            Executive
         3       1003 Koby Bryant         Marketing
         4          1008 Ray Tatum        Marketing
         4          1009 Stephen Book     Marketing
         3       1004 Lebron Jason        Finance
         4          1010 Klay Durant      Finance
         4          1011 David Davis      Finance
         3       1005 Derek Wade          Human Resources
         4          1012 Chris Bosh       Human Resources
         4          1013 Kris Paul        Human Resources
         2    1002 John Magic             Executive
         3       1006 Lawrence Curry      IT
         4          1014 Kevin Thompson   IT
         4          1015 Frank Lee        IT
         3       1007 Paris Harden        Strategy
         4          1016 Matt Lopez       Strategy
         4          1017 Kyle Anderson    Strategy

18 rows selected.
```

**Q.3**

```
SQL> SELECT P.name Project, P.start_date "Start Date",
  2      (CASE EXTRACT(MONTH FROM A.date_assigned)
  3      WHEN 1 THEN 'JAN'
  4      WHEN 2 THEN 'FEB'
  5      WHEN 3 THEN 'MAR'
  6      WHEN 4 THEN 'APR'
  7      WHEN 5 THEN 'MAY'
  8      WHEN 6 THEN 'JUN'
  9      WHEN 7 THEN 'JUL'
 10      WHEN 8 THEN 'AUG'
 11      WHEN 9 THEN 'SEPT'
 12      WHEN 10 THEN 'OCT'
 13      WHEN 11 THEN 'NOV'
 14      WHEN 12 THEN 'DEC'
```

```
 15      END)
 16      "MONTH",
 17       COUNT(A.emp_num) "# of Employee", SUM(A.hours_used) "Hours"
 18  FROM project P JOIN assignment A ON P.proj_number = A.proj_number
 19  WHERE P.Total_Cost IS NULL
 20  GROUP BY GROUPING SETS ((P.name, P.start_date, EXTRACT(MONTH FROM A.date_assigned)),P.name)
 21  ORDER BY P.name;

PROJECT                        Start Date MONTH # of Employee      Hours
------------------------------ ---------- ------ ------------ ----------
Ace Solution                   12-OCT-21  OCT              3        120
Ace Solution                   12-OCT-21  NOV              3        130
Ace Solution                   12-OCT-21  DEC              1
Ace Solution                                               7        250
```

**Q.4**

**// In this problem, there is only one employee works on one project during the first quarter of this year, so we just need to set his bonus amount as $200**

```
SQL> ALTER TABLE Employee
  2  ADD (BONUS_AMT NUMBER(4) DEFAULT 0 NOT NULL);

Table altered.

SQL> UPDATE Employee E
  2  SET BONUS_AMT = 200
  3  WHERE Emp_Num = (SELECT emp_num
  4          FROM project P JOIN assignment A USING (Proj_Number)
  5                         JOIN employee   E USING (Emp_Num)
  6          WHERE EXTRACT(MONTH FROM P.start_date) <= 3  AND
  7                EXTRACT(MONTH FROM P.start_date) >= 1 AND
  8                EXTRACT(YEAR FROM P.start_date) = EXTRACT(YEAR FROM SYSDATE)
  9          GROUP BY emp_num
 10          HAVING SUM(A.hours_used) >= 150);

1 row updated.

SQL> SELECT * FROM employee;
```

```
   EMP_NUM LNAME      FNAME      DOB       HIRE_DATE   SUPER_ID  DEPT_CODE  BONUS_AMT
---------- ---------- ---------- --------- --------- ---------- ---------- ----------
      1000 King       Steve      19-DEC-74 19-JUN-10                     6          0
      1001 Jordan     Mike       12-FEB-74 01-AUG-11       1000          6          0
      1002 Magic      John       23-AUG-78 01-AUG-11       1000          6          0
      1003 Bryant     Koby       14-AUG-59 01-AUG-12       1001          1          0
      1004 Jason      Lebron     30-DEC-84 01-JUN-21       1001          2          0
      1005 Wade       Derek      30-JAN-84 01-DEC-12       1001          3          0
      1006 Curry      Lawrence   06-JAN-77 14-JAN-15       1002          4          0
      1007 Harden     Paris      22-JAN-78 12-NOV-11       1002          5          0
      1008 Tatum      Ray        19-MAR-80 13-JUN-11       1003          1        200
      1009 Book       Stephen    06-AUG-90 01-NOV-21       1003          1          0
      1010 Durant     Klay       22-JAN-88 01-MAY-11       1004          2          0
      1011 Davis      David      09-APR-83 01-NOV-21       1004          2          0
      1012 Bosh       Chris      21-AUG-85 01-APR-11       1005          3          0
      1013 Paul       Kris       23-JAN-78 11-AUG-11       1005          3          0
      1014 Thompson   Kevin      09-APR-90 01-MAY-21       1006          4          0
      1015 Lee        Frank      02-JUN-95 14-JUN-21       1006          4          0
      1016 Lopez      Matt       23-JAN-78 14-JUN-12       1007          5          0
      1017 Anderson   Kyle       19-NOV-89 01-JUL-21       1007          5          0

18 rows selected.
```

**Q.5**

```
SQL> SELECT E.emp_num || ': ' ||E.fname ||' '|| E.lname "Employee", E.Hire_Date, T.name Training, T.date_acquired,
  2         (T.date_acquired - E.hire_date) "Days",COUNT(DISTINCT A.proj_number) as "# of Projects"
  3  FROM employee E LEFT JOIN training T   ON E.emp_num = T.emp_num
  4                  LEFT JOIN assignment A ON T.emp_num = A.emp_num
  5  WHERE EXTRACT(YEAR FROM E.hire_date) = EXTRACT(YEAR FROM SYSDATE) AND
  6        EXTRACT(MONTH FROM E.hire_date) > 3 AND
  7        EXTRACT(MONTH FROM E.hire_date) <= 6
  8  GROUP BY E.emp_num, E.fname,E.lname,E.Hire_Date,T.name,T.date_acquired;


Employee                       HIRE_DATE TRAINING                       DATE_ACQU       Days # of Projects
-----------------------------  --------- ------------------------------ --------- ---------- -------------
1004: Lebron Jason             01-JUN-21 Python Programming Bootcamp    01-NOV-21        153             2
1014: Kevin Thompson           01-MAY-21 Java Programming Bootcamp      01-DEC-21        214             1
1015: Frank Lee                14-JUN-21 Java Programming Bootcamp      01-DEC-21        170             0
```

**Q.6**

**// As you can check our Project table, for the projects that are discontinued**

1. **For the completed projects, the projects have the same name and same total cost but different project number.**
2. **For the on-going projects, the projects' total cost is null.**

```
SQL> SELECT P.name client, MIN(P.start_date) "Date",
  2         (CASE
  3          WHEN COUNT(P.name) > 1 THEN 'completed'
  4          WHEN COUNT(P.name) = 1 THEN 'on-going'
  5          END) "Status"
  6  FROM  project P
  7  WHERE P.name IN (
  8         SELECT P.name
  9         FROM project P
 10         GROUP BY P.name
 11         HAVING COUNT(P.name) > 1) OR
 12         P.total_cost IS NULL
 13  GROUP BY P.name;

CLIENT                         Date      Status
------------------------------ --------- ---------
Netflip Hiring                 08-AUG-21 completed
Ace Solution                   12-OCT-21 on-going
```

**Q.7**

```
SQL> SELECT (CASE EXTRACT(MONTH FROM P.start_date)
  2         WHEN 1 THEN 'Q1'
  3         WHEN 2 THEN 'Q1'
  4         WHEN 3 THEN 'Q1'
  5         WHEN 4 THEN 'Q2'
  6         WHEN 5 THEN 'Q2'
  7         WHEN 6 THEN 'Q2'
  8         WHEN 7 THEN 'Q3'
  9         WHEN 8 THEN 'Q3'
 10         WHEN 9 THEN 'Q3'
 11         WHEN 10 THEN 'Q4'
 12         WHEN 11 THEN 'Q4'
```

```
 13         WHEN 12 THEN 'Q4'
 14         END) "Qrt",
 15         COUNT(DISTINCT P.proj_number)"Number of Projects",
 16         COUNT(DISTINCT A.emp_num) "Number of Employees",
 17         SUM(A.hours_used)/COUNT(DISTINCT A.proj_number) "Avg Hours/Project"
 18  FROM project P JOIN assignment A ON (P.proj_number = A.proj_number)
 19  WHERE EXTRACT(YEAR FROM P.start_date) = EXTRACT(YEAR FROM SYSDATE)
 20  GROUP BY EXTRACT(MONTH FROM P.start_date);

Qrt Number of Projects Number of Employees Avg Hours/Project
--- ----------------- ------------------ -----------------
Q1                  1                  2               250
Q2                  2                  3                90
Q3                  1                  4               240
Q4                  1                  5               250
```

**Q.8**

**// In order to make number of training's number of skills data as '-------' ,**
**I assume that each employee will not take more than ten training programs.**

```
SQL> SELECT DECODE(emp_num,NULL,'----',emp_num) AS "ID",
  2         DECODE((E.fname || ' ' || E.lname),' ','Number of Trainings:',
  3                (E.fname || ' ' || E.lname)) "Employee Name",
  4         SUM(DECODE(code,1,1,0)) "Speaking",
  5         (CASE
  6               WHEN E.fname || ' ' || E.lname = ' ' THEN '-------'
  7               WHEN SUM(DECODE(code,1,1,0)) >= 1 THEN TO_CHAR(MAX(T.date_acquired),'MM/DD/YY')
  8               ELSE '-------'
  9               END) AS "Lastest Date Acquired",
 10         SUM(DECODE(code,2,1,0)) "Java",
 11         (CASE
 12               WHEN E.fname || ' ' || E.lname = ' ' THEN '-------'
 13               WHEN SUM(DECODE(code,2,1,0)) >= 1 THEN TO_CHAR(MAX(T.date_acquired),'MM/DD/YY')
 14               ELSE '-------'
 15               END) AS "Lastest Date Acquired",
 16         SUM(DECODE(code,3,1,0)) "C++",
 17         (CASE
 18               WHEN E.fname || ' ' || E.lname = ' ' THEN '-------'
 19               WHEN SUM(DECODE(code,3,1,0)) >= 1 THEN TO_CHAR(MAX(T.date_acquired),'MM/DD/YY')
```

```
 20               ELSE '-------'
 21               END) AS "Lastest Date Acquired",
 22         SUM(DECODE(code,4,1,0)) "Python",
 23         (CASE
 24               WHEN E.fname || ' ' || E.lname = ' ' THEN '-------'
 25               WHEN SUM(DECODE(code,4,1,0)) >= 1 THEN TO_CHAR(MAX(T.date_acquired),'MM/DD/YY')
 26               ELSE '-------'
 27               END) AS "Lastest Date Acquired",
 28         SUM(DECODE(code,5,1,0)) "Leadership",
 29         (CASE
 30               WHEN E.fname || ' ' || E.lname = ' ' THEN '-------'
 31               WHEN SUM(DECODE(code,5,1,0)) >= 1 THEN TO_CHAR(MAX(T.date_acquired),'MM/DD/YY')
 32               ELSE '-------'
 33               END) AS "Lastest Date Acquired",
 34         SUM(DECODE(code,6,1,0)) "Writing",
 35         (CASE
 36               WHEN E.fname || ' ' || E.lname = ' ' THEN '-------'
 37               WHEN SUM(DECODE(code,6,1,0)) >= 1 THEN TO_CHAR(MAX(T.date_acquired),'MM/DD/YY')
 38               ELSE '-------'
 39               END) AS "Lastest Date Acquired",
 40         SUM(DECODE(code,7,1,0)) "Decision Making",
 41         (CASE
 42               WHEN E.fname || ' ' || E.lname = ' ' THEN '-------'
 43               WHEN SUM(DECODE(code,7,1,0)) >= 1 THEN TO_CHAR(MAX(T.date_acquired),'MM/DD/YY')
 44               ELSE '-------'
 45               END) "Lastest Date Acquired",
 46         (CASE
 47               WHEN COUNT (code) > 10 THEN '-------'
 48               ELSE TO_CHAR(COUNT (code))
 49               END) "Number of Skills"
 50  FROM employee E JOIN training T USING (emp_num)
 51                  JOIN skill    S USING (code)
 52  GROUP BY GROUPING SETS ((emp_num,E.fname,E.lname),());
```

| ID | Employee Name | Speaking | Latest Date Acquired | Java | Latest Date Acquired | C++ | Latest Date Acquired | Python | Latest Date Acquired | Leadership | Latest Date Acquired | Writing | Latest Date Acquired | Decision Making | Latest Date Acquired | Number of Skills: |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1001 | Mike Jordan | 1 | 01/01/15 | 0 | ------- | 0 | ------- | 0 | ------- | 0 | ------- | 0 | ------- | 0 | ------- | 1 |
| 1002 | John Magic | 0 | ------- | 1 | 02/01/12 | 0 | ------- | 0 | ------- | 0 | ------- | 0 | ------- | 0 | ------- | 1 |

```
1003 Koby Bryant                0 -------       0 -------       1 10/01/12        0 -------         0 -------         0 -------              0 -------  1
1004 Lebron Jason               0 -------       0 -------       0 -------         1 11/01/21        0 -------         0 -------              0 -------  1
1005 Derek Wade                 0 -------       0 -------       0 -------         0 -------         1 05/01/19        0 -------              0 -------  1
1006 Lawrence Curry             0 -------       0 -------       0 -------         0 -------         0 -------         1 06/01/20             0 -------  1
1007 Paris Harden               0 -------       0 -------       0 -------         0 -------         0 -------         0 -------              1 07/01/20 1
1008 Ray Tatum                  0 -------       1 08/01/19      0 -------         0 -------         0 -------         0 -------              0 -------  1
1009 Stephen Book               0 -------       0 -------       0 -------         1 12/01/21        0 -------         0 -------              0 -------  1
1010 Klay Durant                0 -------       0 -------       1 12/01/21        0 -------         0 -------         0 -------              0 -------  1
1011 David Davis                0 -------       0 -------       0 -------         0 -------         0 -------         0 -------              1 12/01/21 1
1012 Chris Bosh                 0 -------       0 -------       0 -------         0 -------         0 -------         0 -------              1 07/01/21 1
1013 Kris Paul                  0 -------       2 12/01/21      0 -------         0 -------         0 -------         0 -------              0 -------  2
1014 Kevin Thompson             0 -------       1 12/01/21      0 -------         0 -------         0 -------         0 -------              0 -------  1
1015 Frank Lee                  0 -------       1 12/01/21      0 -------         0 -------         0 -------         0 -------              0 -------  1
1016 Matt Lopez                 1 12/01/21      0 -------       0 -------         0 -------         0 -------         0 -------              0 -------  1
1017 Kyle Anderson              1 12/01/21      0 -------       0 -------         0 -------         0 -------         0 -------              0 -------  1
---- Number of Trainings:       3 -------       6 -------       2 -------         2 -------         1 -------         1 -------              3 -------  ------
```

**Q.9**

```
SQL> select d.name department, s.name skill, count(t.train_num) Num_Training,
  2            rank() over (order by count(t.train_num)) rank
  3  from department d join employee e on d.dept_code=e.dept_code
  4                join training t on t.emp_num=e.emp_num
  5                join skill s on s.code = t.code
  6  group by d.name, s.name;

DEPARTMENT      SKILL            NUM_TRAINING       RANK
--------------- --------------- ------------ ----------
Finance         C++                         1          1
Human Resources Leadership                  1          1
Marketing       Python                      1          1
Marketing       C++                         1          1
Executive       Java                        1          1
Finance         Python                      1          1
IT              Writing                     1          1
Executive       Speaking                    1          1
Human Resources Decision Making             1          1
Finance         Decision Making             1          1
Strategy        Decision Making             1          1
Marketing       Java                        1          1
Strategy        Speaking                    2         13
Human Resources Java                        2         13
```

```
IT              Java                        2         13

15 rows selected.
```

**Q.10**

**// Rank assignment numbers to get the first 3 records in subqueries**
**Divide result set into subsets using partition**

```
SQL> select proj_number||': '|| name project, sum(date_ended-date_assigned) total_days_worked
  2  from project JOIN assignment using (proj_number)
  3  WHERE proj_number IN
  4     (SELECT DISTINCT proj_number FROM
  5          (SELECT proj_number, sum(total_days_worked)
  6                  FROM(SELECT * FROM
  7                       (SELECT PROJ_number,assign_num, total_days_worked,
  8                       (CASE rank when 1 THEN 1
  9                                  WHEN 2 THEN 1
 10                                  WHEN 3 THEN 1
 11                                  ELSE 0
 12                                  END) FIRST_3_RECORDS
 13  FROM(
 14  SELECT proj_number, assign_num, total_days_worked,
 15  RANK() OVER (PARTITION BY proj_number
 16               ORDER BY total_days_worked desc) rank
 17  FROM (
 18     SELECT proj_number, assign_num, (date_ended-date_assigned) total_days_worked
 19  FROM
 20  project JOIN assignment USING (proj_number))))
 21  WHERE FIRST_3_RECORDS = 1)
 22  GROUP BY proj_number
 23  HAVING
 24  SUM (total_days_worked) >= 60)
 25  GROUP BY proj_number, name
 26  HAVING COUNT(assign_num) >=5;

PROJECT                                                                 TOTAL_DAYS_WORKED
----------------------------------------------------------------------- -----------------
1013: Netflip Hiring                                                                128
1010: Ace Solution                                                                  131
```

**Q.11**

```
SQL> select e.emp_num||': '||e.lname Employee, e.hire_date, e.DOB,
  2  NVL(d.name, 'None') dept_managing, s.num_supervising
  3  from employee e left join department d on e.emp_num=d.manager_id
  4  left join (select super_id, count(*) as num_supervising
  5  from employee
  6  group by super_id) s
  7  on(e.emp_num = s.super_id)
  8  where rownum < 5
  9  order by e.dob;

EMPLOYEE                       HIRE_DATE DOB       DEPT_MANAGING   NUM_SUPERVISING
------------------------------ --------- --------- --------------- ---------------
1003: Bryant                   01-AUG-12 14-AUG-59 Marketing                     2
1001: Jordan                   01-AUG-11 12-FEB-74 None                          3
1000: King                     19-JUN-10 19-DEC-74 Executive                     2
1006: Curry                    14-JAN-15 06-JAN-77 IT                            2
```

**Q.12**

```
SQL> SELECT
  2  (CASE
  3  WHEN web_address LIKE '%.edu' then 'education institute'
  4  WHEN web_address LIKE '%.org' then 'non-for-Profit'
  5  WHEN web_address LIKE '%.com' then 'for-Profit'
  6  WHEN web_address LIKE '%.gov' then 'government'
  7  WHEN web_address IS NULL then 'not available'
  8  ELSE 'Other'
  9  end) client_type,
 10  COUNT(distinct(web_address)) AS "Number of each type",
 11  COUNT(DISTINCT project.proj_number)"Number of project"
 12  FROM client JOIN project
 13   ON client.client_id = project.client_id
 14  GROUP BY (CASE
 15  WHEN web_address LIKE '%.edu' then 'education institute'
 16  WHEN web_address LIKE '%.org' then 'non-for-Profit'
 17  WHEN web_address LIKE '%.com' then 'for-Profit'
```

```
 18  WHEN web_address LIKE '%.gov' then 'government'
 19  WHEN web_address IS NULL then 'not available'
 20  ELSE 'Other'
 21  end);

CLIENT_TYPE         Number of each type Number of project
------------------- ------------------- -----------------
education institute                   2                 2
for-Profit                            5                 6
government                            1                 1
non-for-Profit                        3                 5
```

**Q.13**

```
SQL> SELECT emp_num||': '||fname||' '||lname Employee, name Department, NVL(project_name,'NONE') Last_Project
  2  FROM (SELECT emp_num FROM employee
  3  MINUS
  4  SELECT DISTINCT emp_num FROM assignment
  5  WHERE  date_assigned < to_date('2021-08-01','YYYY-MM-DD') or date_assigned is NULL)
  6  LEFT JOIN
  7  (SELECT e.emp_num, a.proj_number, name project_name
  8  FROM( SELECT emp_num, MAX(date_assigned) maxdate
  9  FROM assignment
 10  GROUP BY emp_Num) e
 11  JOIN assignment a ON (e.emp_num = a.emp_num AND e.maxdate = a.date_assigned)
 12  JOIN project p ON (a.proj_number = p.proj_number)) USING (emp_num)
 13  JOIN employee USING(emp_num)
 14  JOIN department USING(dept_code)
 15  ORDER BY department.name, employee.lname;

EMPLOYEE                                                       DEPARTMENT      LAST_PROJECT
-------------------------------------------------------------- --------------- ------------------------------
1011: David Davis                                              Finance         NONE
1010: Klay Durant                                              Finance         NONE
1004: Lebron Jason                                             Finance         Netflip Hiring
1013: Kris Paul                                                Human Resources Netflip Hiring
1015: Frank Lee                                                IT              NONE
1014: Kevin Thompson                                           IT              Ace Solution
1009: Stephen Book                                             Marketing       NONE
1017: Kyle Anderson                                            Strategy        NONE
```

```
1016: Matt Lopez                                          Strategy        NONE
```

**Q.14**

```
SQL> select s.category skill_category, count(distinct t.train_num) num_training,
  2  count(distinct p.proj_number) num_projects
  3  from skill s join training t on s.code=t.code
  4  join project p on t.code=p.code
  5  group by s.category
  6  union all
  7  select 'Grand Total', count(distinct t.train_num) , count(distinct p.proj_number)
  8  from skill s join training t on s.code=t.code
  9  join project p on t.code=p.code;
SKILL_CATEGORY  NUM_TRAINING NUM_PROJECTS
--------------- ------------ ------------
Communication              4            3
Management                 4            7
Programming               10            4
Grand Total               18           14
```

**Q.15**

**// Extract constraint type from constraint_name**
**   Extract reference tables and columns from r_constraint_name**

```
SQL> break on table_name on column_name
SQL> column table_name format A10
SQL> column column_name format A13
SQL> column constraint_name format A28
SQL> column type format A04
SQL> column refer_table format A12
SQL> column refer_column format A12
SQL> column search_condition format A75
SQL> SELECT table_name, column_name, constraint_name, substr(constraint_name,-2) type,
  2  search_condition, substr(r_constraint_name,1, instr(r_constraint_name, '_')-1) refer_table,
  3  substr(r_constraint_name, instr(r_constraint_name, '_')+1,
  4  length(r_constraint_name)-instr(r_constraint_name,'_')-3) refer_column
  5  FROM user_constraints JOIN user_tab_columns USING (table_name)
  6  ORDER BY table_name, column_name;

TABLE_NAME COLUMN_NAME   CONSTRAINT_NAME             TYPE SEARCH_CONDITION                                                                 REFER_TABLE  REFER_COLUMN
```

| | | | | | | |
|---|---|---|---|---|---|---|
| ASSIGNMENT | ASSIGN_NUM | ASSIGNMENT_HOURS_USED_CK | CK | Hours_Used >= 0 | | |
| | | ASSIGNMENT_PROJ_NUMBER_FK | FK | | PROJECT | PROJ_NUMBER |
| | | ASSIGNMENT_DATE_ENDED_CK | CK | Date_Ended >= Date_Assigned | | |
| | | ASSIGNMENT_EMP_NUMB_FK | FK | | EMPLOYEE | EMP_NUM |
| | | ASSIGNMENT_ASSIGN_NUM_PK | PK | | | |
| | DATE_ASSIGNED | ASSIGNMENT_PROJ_NUMBER_FK | FK | | PROJECT | PROJ_NUMBER |
| | | ASSIGNMENT_ASSIGN_NUM_PK | PK | | | |
| | | ASSIGNMENT_DATE_ENDED_CK | CK | Date_Ended >= Date_Assigned | | |
| | | ASSIGNMENT_EMP_NUMB_FK | FK | | EMPLOYEE | EMP_NUM |
| | | ASSIGNMENT_HOURS_USED_CK | CK | Hours_Used >= 0 | | |
| | DATE_ENDED | ASSIGNMENT_PROJ_NUMBER_FK | FK | | PROJECT | PROJ_NUMBER |
| | | ASSIGNMENT_EMP_NUMB_FK | FK | | EMPLOYEE | EMP_NUM |
| | | ASSIGNMENT_ASSIGN_NUM_PK | PK | | | |
| | | ASSIGNMENT_HOURS_USED_CK | CK | Hours_Used >= 0 | | |
| | | ASSIGNMENT_DATE_ENDED_CK | CK | Date_Ended >= Date_Assigned | | |
| | EMP_NUM | ASSIGNMENT_ASSIGN_NUM_PK | PK | | | |
| | | ASSIGNMENT_EMP_NUMB_FK | FK | | EMPLOYEE | EMP_NUM |
| | | ASSIGNMENT_HOURS_USED_CK | CK | Hours_Used >= 0 | | |
| | | ASSIGNMENT_DATE_ENDED_CK | CK | Date_Ended >= Date_Assigned | | |
| | | ASSIGNMENT_PROJ_NUMBER_FK | FK | | PROJECT | PROJ_NUMBER |
| | HOURS_USED | ASSIGNMENT_ASSIGN_NUM_PK | PK | | | |
| | | ASSIGNMENT_HOURS_USED_CK | CK | Hours_Used >= 0 | | |
| | | ASSIGNMENT_PROJ_NUMBER_FK | FK | | PROJECT | PROJ_NUMBER |
| | | ASSIGNMENT_EMP_NUMB_FK | FK | | EMPLOYEE | EMP_NUM |
| | | ASSIGNMENT_DATE_ENDED_CK | CK | Date_Ended >= Date_Assigned | | |
| | PROJ_NUMBER | ASSIGNMENT_ASSIGN_NUM_PK | PK | | | |
| | | ASSIGNMENT_HOURS_USED_CK | CK | Hours_Used >= 0 | | |
| | | ASSIGNMENT_EMP_NUMB_FK | FK | | EMPLOYEE | EMP_NUM |
| | | ASSIGNMENT_PROJ_NUMBER_FK | FK | | PROJECT | PROJ_NUMBER |
| | | ASSIGNMENT_DATE_ENDED_CK | CK | Date_Ended >= Date_Assigned | | |
| CLIENT | CITY | CLIENT_STATE_CK | CK | STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL','IN' | | |
| | | CLIENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | CLIENT_STATE_NN | NN | "STATE" IS NOT NULL | | |
| | | CLIENT_INDUSTRY_CK | CK | INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment') | | |
| | | CLIENT_CLIENT_ID_PK | PK | | | |
| | CLIENT_ID | CLIENT_STATE_NN | NN | "STATE" IS NOT NULL | | |
| | | CLIENT_STATE_CK | CK | STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL','IN' | | |
| | | CLIENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | CLIENT_INDUSTRY_CK | CK | INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment') | | |
| | | CLIENT_CLIENT_ID_PK | PK | | | |
| | CONTACT_FNAME | CLIENT_STATE_CK | CK | STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL' | | |

```
                                        ,'IN'

             CLIENT_CLIENT_ID_PK        PK
             CLIENT_STATE_NN            NN   "STATE" IS NOT NULL
             CLIENT_NAME_NN             NN   "NAME" IS NOT NULL
             CLIENT_INDUSTRY_CK         CK   INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment')
CONTACT_LNAME CLIENT_CLIENT_ID_PK       PK
             CLIENT_STATE_CK            CK   STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL'
                                        ,'IN'

             CLIENT_STATE_NN            NN   "STATE" IS NOT NULL
             CLIENT_INDUSTRY_CK         CK   INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment')
             CLIENT_NAME_NN             NN   "NAME" IS NOT NULL
INDUSTRY     CLIENT_STATE_NN            NN   "STATE" IS NOT NULL
             CLIENT_CLIENT_ID_PK        PK
             CLIENT_STATE_CK            CK   STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL'
                                        ,'IN'

             CLIENT_INDUSTRY_CK         CK   INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment')
             CLIENT_NAME_NN             NN   "NAME" IS NOT NULL
NAME         CLIENT_STATE_CK            CK   STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL'
                                        ,'IN'

             CLIENT_INDUSTRY_CK         CK   INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment')
             CLIENT_CLIENT_ID_PK        PK
             CLIENT_NAME_NN             NN   "NAME" IS NOT NULL
             CLIENT_STATE_NN            NN   "STATE" IS NOT NULL
PHONE        CLIENT_NAME_NN             NN   "NAME" IS NOT NULL
             CLIENT_STATE_CK            CK   STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL'
                                        ,'IN'

             CLIENT_STATE_NN            NN   "STATE" IS NOT NULL
             CLIENT_INDUSTRY_CK         CK   INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment')
             CLIENT_CLIENT_ID_PK        PK
STATE        CLIENT_STATE_CK            CK   STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL'
                                        ,'IN'

             CLIENT_STATE_NN            NN   "STATE" IS NOT NULL
             CLIENT_CLIENT_ID_PK        PK
             CLIENT_INDUSTRY_CK         CK   INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment')
             CLIENT_NAME_NN             NN   "NAME" IS NOT NULL
STREET       CLIENT_STATE_CK            CK   STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL'
                                        ,'IN'

             CLIENT_STATE_NN            NN   "STATE" IS NOT NULL
             CLIENT_NAME_NN             NN   "NAME" IS NOT NULL
```

| | | | | | | |
|---|---|---|---|---|---|---|
| | | CLIENT_CLIENT_ID_PK | PK | | | |
| | | CLIENT_INDUSTRY_CK | CK | INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment') | | |
| | WEB_ADDRESS | CLIENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | CLIENT_STATE_NN | NN | "STATE" IS NOT NULL | | |
| | | CLIENT_STATE_CK | CK | STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL','IN' | | |
| | | CLIENT_CLIENT_ID_PK | PK | | | |
| | | CLIENT_INDUSTRY_CK | CK | INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment') | | |
| | ZIP_CODE | CLIENT_STATE_NN | NN | "STATE" IS NOT NULL | | |
| | | CLIENT_STATE_CK | CK | STATE IN ('AL', 'AK','AZ','AR','CA','CO','CT','DE','FL','GA','HI','ID','IL','IN' | | |
| | | CLIENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | CLIENT_CLIENT_ID_PK | PK | | | |
| | | CLIENT_INDUSTRY_CK | CK | INDUSTRY IN ('Internet', 'E-Commerce', 'Banking', 'Investment') | | |
| DEPARTMENT | DEPT_CODE | DEPARTMENT_MANAGER_ID_CK | CK | Manager_ID > 0 | | |
| | | DEPARTMENT_LOCATION_NN | NN | "LOCATION" IS NOT NULL | | |
| | | DEPARTMENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | DEPARTMENT_DEPT_CODE_PK | PK | | | |
| | | DEPARTMENT_LOCATION_CK | CK | Location IN ('A100', 'B101', 'B103', 'C303', 'A102','D101') | | |
| | LOCATION | DEPARTMENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | DEPARTMENT_LOCATION_CK | CK | Location IN ('A100', 'B101', 'B103', 'C303', 'A102','D101') | | |
| | | DEPARTMENT_LOCATION_NN | NN | "LOCATION" IS NOT NULL | | |
| | | DEPARTMENT_MANAGER_ID_CK | CK | Manager_ID > 0 | | |
| | | DEPARTMENT_DEPT_CODE_PK | PK | | | |
| | MANAGER_ID | DEPARTMENT_LOCATION_CK | CK | Location IN ('A100', 'B101', 'B103', 'C303', 'A102','D101') | | |
| | | DEPARTMENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | DEPARTMENT_DEPT_CODE_PK | PK | | | |
| | | DEPARTMENT_LOCATION_NN | NN | "LOCATION" IS NOT NULL | | |
| | | DEPARTMENT_MANAGER_ID_CK | CK | Manager_ID > 0 | | |
| | NAME | DEPARTMENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | DEPARTMENT_LOCATION_NN | NN | "LOCATION" IS NOT NULL | | |
| | | DEPARTMENT_MANAGER_ID_CK | CK | Manager_ID > 0 | | |
| | | DEPARTMENT_LOCATION_CK | CK | Location IN ('A100', 'B101', 'B103', 'C303', 'A102','D101') | | |
| | | DEPARTMENT_DEPT_CODE_PK | PK | | | |
| | PHONE | DEPARTMENT_LOCATION_NN | NN | "LOCATION" IS NOT NULL | | |
| | | DEPARTMENT_DEPT_CODE_PK | PK | | | |
| | | DEPARTMENT_MANAGER_ID_CK | CK | Manager_ID > 0 | | |
| | | DEPARTMENT_LOCATION_CK | CK | Location IN ('A100', 'B101', 'B103', 'C303', 'A102','D101') | | |
| | | DEPARTMENT_NAME_NN | NN | "NAME" IS NOT NULL | | |
| EMPLOYEE | DEPT_CODE | EMPLOYEE_SUPER_ID_FK | FK | | EMPLOYEE | EMP_NUM |
| | | EMPLOYEE_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | EMPLOYEE_EMP_NUM_PK | PK | | | |
| | DOB | EMPLOYEE_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | EMPLOYEE_EMP_NUM_PK | PK | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | EMPLOYEE_SUPER_ID_FK | FK | | EMPLOYEE | EMP_NUM |
| | EMP_NUM | EMPLOYEE_SUPER_ID_FK | FK | | EMPLOYEE | EMP_NUM |
| | | EMPLOYEE_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | EMPLOYEE_EMP_NUM_PK | PK | | | |
| | FNAME | EMPLOYEE_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | EMPLOYEE_EMP_NUM_PK | PK | | | |
| | | EMPLOYEE_SUPER_ID_FK | FK | | EMPLOYEE | EMP_NUM |
| | HIRE_DATE | EMPLOYEE_SUPER_ID_FK | FK | | EMPLOYEE | EMP_NUM |
| | | EMPLOYEE_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | EMPLOYEE_EMP_NUM_PK | PK | | | |
| | LNAME | EMPLOYEE_SUPER_ID_FK | FK | | EMPLOYEE | EMP_NUM |
| | | EMPLOYEE_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | EMPLOYEE_EMP_NUM_PK | PK | | | |
| | SUPER_ID | EMPLOYEE_EMP_NUM_PK | PK | | | |
| | | EMPLOYEE_SUPER_ID_FK | FK | | EMPLOYEE | EMP_NUM |
| | | EMPLOYEE_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| PROJECT | CLIENT_ID | PROJECT_TOTAL_COST_CK | CK | TOTAL_COST > 0 | | |
| | | PROJECT_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | PROJECT_CODE_FK | FK | | SKILL | CODE |
| | | PROJECT_CLIENT_ID_FK | FK | | CLIENT | CLIENT_ID |
| | | PROJECT_PROJ_NUMBER_PK | PK | | | |
| | CODE | PROJECT_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | PROJECT_PROJ_NUMBER_PK | PK | | | |
| | | PROJECT_CODE_FK | FK | | SKILL | CODE |
| | | PROJECT_CLIENT_ID_FK | FK | | CLIENT | CLIENT_ID |
| | | PROJECT_TOTAL_COST_CK | CK | TOTAL_COST > 0 | | |
| | DEPT_CODE | PROJECT_TOTAL_COST_CK | CK | TOTAL_COST > 0 | | |
| | | PROJECT_CODE_FK | FK | | SKILL | CODE |
| | | PROJECT_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | PROJECT_CLIENT_ID_FK | FK | | CLIENT | CLIENT_ID |
| | | PROJECT_PROJ_NUMBER_PK | PK | | | |
| | NAME | PROJECT_CLIENT_ID_FK | FK | | CLIENT | CLIENT_ID |
| | | PROJECT_CODE_FK | FK | | SKILL | CODE |
| | | PROJECT_PROJ_NUMBER_PK | PK | | | |
| | | PROJECT_TOTAL_COST_CK | CK | TOTAL_COST > 0 | | |
| | | PROJECT_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | PROJ_NUMBER | PROJECT_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | PROJECT_CLIENT_ID_FK | FK | | CLIENT | CLIENT_ID |
| | | PROJECT_CODE_FK | FK | | SKILL | CODE |
| | | PROJECT_TOTAL_COST_CK | CK | TOTAL_COST > 0 | | |
| | | PROJECT_PROJ_NUMBER_PK | PK | | | |
| | START_DATE | PROJECT_TOTAL_COST_CK | CK | TOTAL_COST > 0 | | |
| | | PROJECT_CODE_FK | FK | | SKILL | CODE |
| | | PROJECT_CLIENT_ID_FK | FK | | CLIENT | CLIENT_ID |
| | | PROJECT_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | PROJECT_PROJ_NUMBER_PK | PK | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | TOTAL_COST | PROJECT_CODE_FK | FK | | SKILL | CODE |
| | | PROJECT_DEPT_CODE_FK | FK | | DEPARTMENT | DEPT_CODE |
| | | PROJECT_CLIENT_ID_FK | FK | | CLIENT | CLIENT_ID |
| | | PROJECT_TOTAL_COST_CK | CK | TOTAL_COST > 0 | | |
| | | PROJECT_PROJ_NUMBER_PK | PK | | | |
| SKILL | CATEGORY | SKILL_CATEGORY_CK | CK | Category IN ('Communication','Programming', 'Management') | | |
| | | SKILL_CODE_PK | PK | | | |
| | | SKILL_NAME_CK | CK | Name IN ('Speaking','Java','C++', 'Python', 'Leadership','Writing','Decisio n Making') | | |
| | | SKILL_CODE_CK | CK | Code >= 0 | | |
| | | SKILL_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | CODE | SKILL_CATEGORY_CK | CK | Category IN ('Communication','Programming', 'Management') | | |
| | | SKILL_CODE_CK | CK | Code >= 0 | | |
| | | SKILL_CODE_PK | PK | | | |
| | | SKILL_NAME_CK | CK | Name IN ('Speaking','Java','C++', 'Python', 'Leadership','Writing','Decisio n Making') | | |
| | | SKILL_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | NAME | SKILL_CODE_PK | PK | | | |
| | | SKILL_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | SKILL_CODE_CK | CK | Code >= 0 | | |
| | | SKILL_CATEGORY_CK | CK | Category IN ('Communication','Programming', 'Management') | | |
| | | SKILL_NAME_CK | CK | Name IN ('Speaking','Java','C++', 'Python', 'Leadership','Writing','Decisio n Making') | | |
| TRAINING | CODE | TRAINING_CODE_FK | FK | | SKILL | CODE |
| | | TRAINING_TRAIN_NUM_PK | PK | | | |
| | | TRAINING_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | TRAINING_EMP_NUM_FK | FK | | EMPLOYEE | EMP_NUM |
| | COMMENTS | TRAINING_TRAIN_NUM_PK | PK | | | |
| | | TRAINING_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | TRAINING_CODE_FK | FK | | SKILL | CODE |
| | | TRAINING_EMP_NUM_FK | FK | | EMPLOYEE | EMP_NUM |
| | DATE_ACQUIRED | TRAINING_CODE_FK | FK | | SKILL | CODE |
| | | TRAINING_EMP_NUM_FK | FK | | EMPLOYEE | EMP_NUM |
| | | TRAINING_TRAIN_NUM_PK | PK | | | |
| | | TRAINING_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | EMP_NUM | TRAINING_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | TRAINING_EMP_NUM_FK | FK | | EMPLOYEE | EMP_NUM |
| | | TRAINING_CODE_FK | FK | | SKILL | CODE |
| | | TRAINING_TRAIN_NUM_PK | PK | | | |
| | NAME | TRAINING_CODE_FK | FK | | SKILL | CODE |
| | | TRAINING_NAME_NN | NN | "NAME" IS NOT NULL | | |
| | | TRAINING_TRAIN_NUM_PK | PK | | | |
| | | TRAINING_EMP_NUM_FK | FK | | EMPLOYEE | EMP_NUM |

```
          TRAIN_NUM     TRAINING_CODE_FK            FK                                                                  SKILL        CODE
                        TRAINING_TRAIN_NUM_PK       PK
                        TRAINING_EMP_NUM_FK         FK                                                                  EMPLOYEE     EMP_NUM
                        TRAINING_NAME_NN            NN   "NAME" IS NOT NULL

205 rows selected.
```