DB Implementation Project

95-703C: Database Management

-----PartII.A-----

---Table Content

--SKILL

Code	Description
0001	Python
0002	R
0003	Power Point
0004	JAVA
0005	Excel
0006	SAS

--DEPARTMENT

Department Code	Department Name	Location	Phone	Manager ID
01	Strategy Development	100 Academy Ave, Pittsburgh	412-888-0274	501
02	Marketing	100 Academy Ave, Pittsburgh	412-888-0394	502
03	IT	100 Academy Ave, Pittsburgh	412-888-0305	503
04	Design	100 Academy Ave, Pittsburgh	412-888-0463	504
05	Administration	100 Academy Ave, Pittsburgh	412-512-0463	504

--EMPLOYEE

Employee#	Last Name	First Name	DOB	Hire Date	Supervisor ID	Department Code
504	Wu	Chien-Shiung	06-AUG-76	13-MAY-17		01
507	Hodgkin	Dorothy	11-DEC-79	04-MAR-13	504	02
511	Meitner	Lise	19-MAY-80	06-MAR-13	504	03
501	Pascal	Blaise	01-OCT-70	05-SEP-15	504	01
514	Bohr	Niels	08-JAN-80	10-MAR-08	504	04
502	Herschel	Caroline	12-MAR-75	23-JAN-09	501	01
503	Halley	Edmond	22-MAY-74	13-MAY-16	501	01
505	Fermi	Enrico	12-APR-80	04-MAR-16	501	01
506	Seydoux	Geraldine	08-OCT-79	04-MAR-17	501	01
508	John	Snow	18-OCT-80	12-AUG-16	501	01
509	Daubechies	Ingrid	18-JAN-82	12-MAR-12	507	02
510	Kepler	Johannes	20-MAR-85	12-MAR-17	507	02
512	Faraday	Michael	17-FEB-80	20-FEB-14	511	03
513	Franklin	Melissa	21-MAY-82	05-MAR-16	511	03
515	Copernicus	Nicolaus	12-FEB-86	14-JAN-06	514	04
516	Amy	Huang	21-MAR-89	20-APR-15	504	05
517	Jin	Hesien	23-FEB-78	21-MAY-16	516	05

--TRAINING

Training#	Skill Code	Employee#	Acquired Date	Training Name	Comments
1001	0001	501	11-JAN-15	Introduction to Python	For 1st Grade
1002	0001	502	01-FEB-16	Upper Level of Python	For 2nd Grade
1003	0002	502	21-MAR-15	Introduction to R	For 1st Grade
1004	0002	502	21-OCT-16	Upper Level of R	For 2nd Grade
1005	0003	505	01-MAY-17	Introduction to Power Point	For 1st Grade
1006	0003	506	08-JUN-17	Upper Level of Power Point	For 2nd Grade
1007	0004	507	23-AUG-15	Introduction to JAVA	For 1st Grade
1008	0004	508	31-AUG-16	Upper Level of JAVA	For 2nd Grade
1009	0005	509	30-JAN-14	Introduction to Excel	For 1st Grade
1010	0005	510	21-OCT-17	Upper Level of Excel	For 2nd Grade
1011	0006	515	06-NOV-16	Introduction to SAS	For 1st Grade
1012	0006	512	23-DEC-15	Upper Level of SAS	For 2nd Grade
1033	0001	513	11-MAY-16	Introduction to Python	For 1st Grade
1034	0006	513	01-JAN-17	Introduction to SAS	For 1st Grade
1035	0004	513	23-OCT-16	Introduction to JAVA	For 1st Grade
1036	0002	508	21-SEP-16	Upper Level of R	For 2nd Grade
1037	0005	510	28-APR-17	Introduction to Excel	For 1st Grade

--CLIENT

ID	Name	Street	City	State	Zip_Code	Industry	Website	Phone	Contact
C506	Pfizer	212 Bellefonte St.	Pittsburgh	PA	15232	pharmacy	www.pfizer.com	412-334-112	Jim
C502	Merck	2000 Galloping Hill Rd	Kenilworth	NJ	07033	pharmacy	www.merck.com	800-444-2080	Jimmy
C510	Goldman	125 High Street	Boston	MA	02110	Bank	www.goldman.com	617-204-2000	Scott
C512	Evercore	55 East 52nd Street	New York	NY	10055	Bank	www.evercore.com	212-857-3100	Sophia
C530	Twitter	1355 Market Street	San Francisco	CA	94103	Internet	www.twitter.com	415-222-9670	Dorsey
C542	Facebook	1601 Willow Road	Menlo Park	CA	94025	Internet	www.facebook.com	650-543-4800	Mark
C538	Walmart	702 SW 8th Street	Bentonville	AR	72716	Retail	www.walmart.com	479-273-4000	Biggs
C532	JCPenny	6501 Legacy Drive	Plano	TX	75024	Retail	www.jcpenny.com	972-431-1000	Cobe
C591	Kroger	1014 Vine Street	Cincinnati	OH	45202	Retail	www.kroger.com	513-762-4000	Michael
C580	Pepsi	700 Anderson Hill Road	Purchase	NY	10577	Food	www.kroger.com	914-253-2000	Albert

--PROJECT

Project #	Name	Start_Date	Cost	Dept_Code	Client_ID
H1255	Product Launch	01-JUL-16	8000	01	C538
H1240	Organization Design	17-MAY-14	5000	03	C538
H2315	Market Research	25-JAN-09	3000	04	C538
H0988	IPO Evaluation	22-OCT-13	1000	02	C512
H5490	System Implementation	06-SEP-17	NA	01	C542
H8877	Market Research	06-NOV-17	NA	04	C542
H0045	Customer Focus Group	23-JUN-17	3000	03	C542
H1145	Market Research	05-JUL-16	3000	01	C530
H7611	New Service Testing	25-MAR-17	5400	02	C510
H3961	Performance Evaluation	21-NOV-17	6000	01	C510
H0908	New Drug Testing	25-AUG-10	9500	04	C506
H1314	Marketing Strategy	21-SEP-16	5000	02	C506
H1300	New Drug Testing	08-JUN-17	3500	01	C502
H9666	Customer Satisfaction Survey	01-SEP-14	5000	02	C532
H9641	Market Research	21-DEC-16	6700	01	C532
H8166	Competitor Analysis	20-SEP-17	5300	01	C591
H3126	Market Research	01-SEP-13	5300	01	C580

--ASSIGNMENT

Assign#	Project #	Emp_Num	Assigned Date	Ended Date	Used hours
HC1111	H1255	501	01-JUL-16	31-JUL-16	10
HC1112	H1255	501	01-AUG-16	31-AUG-16	15
HC1113	H1255	501	01-SEP-16	13-SEP-16	20
HC1114	H1255	502	01-AUG-16	31-AUG-16	7
HC1115	H1255	502	01-SEP-16	13-SEP-16	8
HC1116	H1255	503	15-AUG-16	31-AUG-16	20
HC1117	H1255	503	01-SEP-16	13-SEP-16	20
HC1118	H1240	511	17-MAY-14	31-MAY-14	3
HC1166	H1240	512	17-MAY-16	31-MAY-16	4
HC1155	H1240	509	17-MAY-16	31-MAY-16	10
HC1119	H2315	514	25-JAN-09	31-JAN-09	15
HC1120	H2315	514	01-FEB-09	28-FEB-09	10
HC1121	H2315	514	01-MAR-09	21-MAR-09	8
HC1124	H0988	509	22-OCT-13	31-OCT-13	20
HC1125	H0988	509	15-DEC-13	31-DEC-13	5
HC1126	H0988	507	22-OCT-13	31-OCT-13	12
HC1128	H5490	506	01-OCT-17	31-OCT-17	5
HC1129	H5490	506	01-DEC-17	NA	NA
HC1130	H5490	504	06-SEP-17	30-SEP-17	10
HC1131	H5490	504	01-OCT-17	31-OCT-17	12
HC1132	H5490	504	01-DEC-17	NA	NA
HC1133	H5490	503	06-SEP-17	30-SEP-17	8

HC1134	H5490	503	01-OCT-17	31-OCT-17	15
HC1135	H5490	503	01-DEC-17	NA	NA
HC1170	H5490	501	01-OCT-17	31-OCT-17	3
HC1171	H5490	501	01-DEC-17	NA	NA
HC1138	H0045	513	23-JUN-17	30-JUN-17	8
HC1139	H0045	513	01-JUL-17	31-JUL-17	5
HC1140	H0045	513	01-AUG-17	30-AUG-17	10
HC1141	H0045	513	01-SEP-17	30-SEP-17	7
HC1142	H1145	502	05-JUL-16	31-JUL-16	50
HC1143	H1145	501	01-AUG-16	31-AUG-16	55
HC1144	H7611	507	05-MAR-17	31-MAR-17	13
HC1145	H7611	509	05-MAR-17	31-MAR-17	15
HC1147	H8877	501	21-DEC-17	NA	NA
HC1148	H8877	502	21-DEC-17	NA	NA
HC1149	H8877	503	21-DEC-17	NA	NA
HC1150	H8877	504	21-DEC-17	NA	NA
HC1151	H8877	506	21-DEC-17	NA	NA
HC1152	H0908	514	25-AUG-10	31-AUG-10	5
HC1153	H0908	514	01-NOV-10	30-NOV-10	15
HC1154	H1314	507	21-SEP-16	30-SEP-16	45
HC1188	H1314	507	01-OCT-16	31-OCT-16	10
HC1156	H1314	509	01-OCT-16	31-OCT-16	15
HC1158	H9666	507	01-SEP-14	30-SEP-14	36
HC1159	H9666	507	01-OCT-14	15-OCT-14	20
HC1160	H9666	509	01-OCT-14	15-OCT-14	18

HC1161	H9641	503	21-DEC-16	31-DEC-16	21
HC1162	H8166	504	20-SEP-17	30-SEP-17	28
HC1163	H8166	504	01-OCT-17	31-OCT-17	10
HC1164	H8166	504	01-NOV-17	20-NOV-17	20
HC1165	H8166	506	01-NOV-17	20-NOV-17	20
HC1181	H3126	502	01-SEP-13	30-SEP-13	8
HC1167	H3126	502	01-OCT-13	31-OCT-13	5
HC1168	H3126	502	01-NOV-13	30-NOV-13	10
HC1169	H3126	502	01-DEC-13	25-DEC-13	4
HC1172	H5490	508	06-SEP-17	30-SEP-17	32
HC1173	H5490	508	01-OCT-17	31-OCT-17	10
HC1174	H5490	508	01-NOV-17	30-NOV-17	40
HC1175	H1255	508	15-NOV-16	30-NOV-16	40
HC1176	H1255	508	01-DEC-16	31-DEC-16	21
HC1177	H8877	501	06-NOV-17	30-NOV-17	15
HC1178	H8877	502	06-NOV-17	30-NOV-17	30

-----PartII.B-----

---Q1

The company’s human resources department needs information on the current training levels of the employees. They need information that shows, for each employee, and his/her acquired skills, the number of times he or she received training for that skill, the most recent date of the training, and the number of months (full months only) that have passed since the most recent training for the skill. Include all employees in the output, even if they have not received any training yet.

```
SQL> COLUMN "ID" FORMAT A5
SQL> COLUMN "Employee" FORMAT A20
SQL> COLUMN "Skill Name" FORMAT A11
SQL> COLUMN "# Training" FORMAT A15
SQL> COLUMN "Most Recent" FORMAT A10
SQL>
SQL> BREAK ON "ID" ON"Employee"
SQL> SELECT Emp_Num "ID", Emp_Last||' '||Emp_First "Employee", NVL(Description,'NA') "Skill Name",
  2         NVL(DECODE(COUNT(Train_Num),0,NULL,COUNT(Train_Num)),0) AS "# Training",
  3         NVL(TO_CHAR(MAX(Date_Acquired)),'NA') AS "Most Recent",
  4         NVL(TRUNC(MONTHS_BETWEEN(SYSDATE,MAX(Date_Acquired)),0),0) AS "Months Passed"
  5  FROM Employee LEFT OUTER JOIN Training USING (Emp_Num)
  6                LEFT OUTER JOIN Skill USING (Code)
  7  GROUP BY Emp_Num,Emp_Last,Emp_First,Code,description
  8  ORDER BY 1;
```

ID	Employee	Skill Name	# Training	Most Recent	Months Passed
501	Pascal Blaise	Python	1	11-JAN-15	34
502	Herschel Caroline	Python	1	01-FEB-16	22
		R	2	21-OCT-16	13
503	Halley Edmond	NA	0	NA	0
504	Wu Chien-Shiung	NA	0	NA	0
505	Fermi Enrico	Power Point	1	01-MAY-17	7
506	Seydoux Geraldine	Power Point	1	08-JUN-17	5
507	Hodgkin Dorothy	JAVA	1	23-AUG-15	27
508	John Snow	R	1	21-SEP-16	14
		JAVA	1	31-AUG-16	15
509	Daubechies Ingrid	Excel	1	30-JAN-14	46
510	Kepler Johannes	Excel	2	21-OCT-17	1
511	Meitner Lise	NA	0	NA	0
512	Faraday Michael	SAS	1	23-DEC-15	23
513	Franklin Melissa	Python	1	11-MAY-16	18
		JAVA	1	23-OCT-16	13
		SAS	1	01-JAN-17	11
514	Bohr Niels	NA	0	NA	0
515	Copernicus Nicolaus	SAS	1	06-NOV-16	13
516	Amy Huang	NA	0	NA	0
517	Jin Hesien	NA	0	NA	0

21 rows selected.

---Q2

Due to a recent re-organization, the company needs a list that shows who supervises who. List all the employees in the company clearly indicating the organizational hierarchy. Include the “level” of the hierarchy each employee is at and the department name of the employee.

```
SQL> COLUMN "Employee" FORMAT A40
SQL> COLUMN "Dept_Name" FORMAT A30
SQL>
SQL> SELECT LEVEL,
  2         LPAD(' ', 3*(LEVEL-1)) || Emp_Num || '   ' || Emp_Last || ' ' || Emp_First "Employee", Name"Dept_Name"
  3  FROM Employee LEFT OUTER JOIN Department USING(Dept_Code)
  4  START WITH Emp_Num = (SELECT Emp_Num
  5                        FROM Employee
  6                        WHERE Super_ID IS NULL)
  7  CONNECT BY PRIOR Emp_Num = Super_ID;
```

LEVEL	Employee	Dept_Name
1	504 Wu Chien-Shiung	Strategy Development
2	501 Pascal Blaise	Strategy Development
3	502 Herschel Caroline	Strategy Development
3	503 Halley Edmond	Strategy Development
3	505 Fermi Enrico	Strategy Development
3	506 Seydoux Geraldine	Strategy Development
3	508 John Snow	Strategy Development
2	507 Hodgkin Dorothy	Marketing
3	509 Daubechies Ingrid	Marketing
3	510 Kepler Johannes	Marketing
2	511 Meitner Lise	IT
3	512 Faraday Michael	IT
3	513 Franklin Melissa	IT
2	514 Bohr Niels	Design
3	515 Copernicus Nicolaus	Design
2	516 Amy Huang	Administration
3	517 Jin Hesien	Administration

17 rows selected.

---Q3

For every ongoing project (i.e., a project with total cost being NULL) provide the project name, project start date. In addition, for each month of the project list the number of employees assigned that month and hours spent on the project that month. At the end provide an overall total of the number of employees assigned to the project and the total hours spent on the project.

```
SQL> COLUMN "Project #" FORMAT A13

SQL> COLUMN "Project Name" FORMAT A30

SQL> COLUMN "# hour" FORMAT A15

SQL> COLUMN "Month" FORMAT A10

SQL>

SQL> BREAK ON "Project #" ON "Project Name" ON "Start Date" ON "Year"

SQL>

SQL> SELECT Proj_number"Project #",Name"Project Name",start_date"Start Date",EXTRACT(YEAR FROM Date_Assigned)"Year",

  2  NVL(TO_CHAR(EXTRACT(MONTH FROM Date_Assigned)),'sub-total')"Month",COUNT(DISTINCT Emp_Num) "#
Employee",NVL(TO_CHAR(SUM(Hours_Used)),'Not completed')"# hour"

  3  FROM Project JOIN Assignment USING (Proj_number)

  4  WHERE Total_Cost IS NULL

  5  GROUP BY GROUPING SETS(

  6  (Proj_number,Name,start_date,EXTRACT(YEAR FROM Date_Assigned),EXTRACT(MONTH FROM Date_Assigned)),

  7  (Proj_number,Name,start_date)

  8  )

  9  ORDER BY 1,4;
```

Project #	Project Name	Start Date	Year	Month	# Employee	# hour
H5490	System Implementation	06-SEP-17	2017	9	3	50
				10	5	45
				11	1	40
				12	4	Not completed
				sub-total	5	135
H8877	Market Research	06-NOV-17	2017	11	2	45
				12	5	Not completed
				sub-total	5	45

8 rows selected.

---Q4

The company has decided to adopt a bonus policy for their employees. Each employee will receive a bonus of $200 for every project started last calendar year, on which they have charged at least 40 hours. As a result, management has asked you to update the employee records to include their current bonus information. In order to accomplish this, you will have to add another column to the EMPLOYEE table (i.e., column BONUS_AMT), before you update the data. After the single Update statement is performed, list the contents of the Employee table.

```
SQL> ALTER TABLE Employee
  2  ADD BONUS_AMT NUMBER(6,2);
Table altered.
SQL>
SQL> UPDATE Employee e
  2  SET BONUS_AMT = (
  3  SELECT NVL(bonus,0)"Bonus"
  4  FROM Employee LEFT OUTER JOIN
  5  (SELECT Emp_Num, 200*COUNT(DISTINCT Proj_Number)bonus
  6   FROM Assignment JOIN Project USING(Proj_Number)
  7   WHERE EXTRACT(YEAR FROM Start_Date) >= EXTRACT(YEAR FROM SYSDATE)-1
  8         AND
  9         (Emp_Num,Proj_Number) IN (SELECT Emp_Num, Proj_Number
 10                                   FROM Assignment JOIN Project USING(Proj_Number)
 11                                   GROUP BY Proj_Number, Emp_Num
 12                                   HAVING SUM(Hours_used) >= 40)
 13   GROUP BY Emp_num
 14   ORDER BY 1,2)
 15  USING(Emp_Num)
 16  WHERE e.Emp_Num = Emp_Num
```

```
 17  );
17 rows updated.
SQL> COLUMN Emp_Num FORMAT A10
SQL> COLUMN Emp_Last FORMAT A10
SQL> COLUMN Emp_First FORMAT A15
SQL> COLUMN Super_ID FORMAT A10
SQL> SELECT *
  2  FROM Employee
  3  ORDER BY Emp_Num;
```

EMP_NUM	EMP_LAST	EMP_FIRST	DOB	HIRE_DATE	SUPER_ID	DEPT_CODE	BONUS_AMT
501	Pascal	Blaise	01-OCT-70	05-SEP-15	504	01	400
502	Herschel	Caroline	12-MAR-75	23-JAN-09	501	01	200
503	Halley	Edmond	22-MAY-74	13-MAY-16	501	01	200
504	Wu	Chien-Shiung	06-AUG-76	13-MAY-17		01	200
505	Fermi	Enrico	12-APR-80	04-MAR-16	501	01	0
506	Seydoux	Geraldine	08-OCT-79	04-MAR-17	501	01	0
507	Hodgkin	Dorothy	11-DEC-79	04-MAR-13	504	02	200
508	John	Snow	18-OCT-80	12-AUG-16	501	01	400
509	Daubechies	Ingrid	18-JAN-82	12-MAR-12	507	02	0
510	Kepler	Johannes	20-MAR-85	12-MAR-17	507	02	0
511	Meitner	Lise	19-MAY-80	06-MAR-13	504	03	0
512	Faraday	Michael	17-FEB-80	20-FEB-14	511	03	0
513	Franklin	Melissa	21-MAY-82	05-MAR-16	511	03	0
514	Bohr	Niels	08-JAN-80	10-MAR-08	504	04	0
515	Copernicus	Nicolaus	12-FEB-86	14-JAN-06	514	04	0
516	Amy	Huang	21-MAR-89	20-APR-15	504	05	0
517	Jin	Hesien	23-FEB-78	21-MAY-16	516	05	0

---Q5

For each employee hired last calendar year, list the name and the hire date of the employee, name of training received (if any), date of the training, and the number of days between the hire date and the training. Also include number of projects that employee worked on so far.

```
SQL> COLUMN "Emp_Num" FORMAT A7

SQL> COLUMN Name FORMAT A40

SQL> COLUMN "Name" FORMAT A20

SQL> COLUMN "Training" FORMAT A30

SQL>

SQL> BREAK ON "Emp_Num" ON "Name" ON "Hired Date" ON "# project"

SQL> SELECT Emp_Num"Emp_Num", Emp_Last||' '||Emp_First"Name",Hire_Date"Hired Date",COUNT(DISTINCT Proj_Number)"#
project",Name"Training",Date_acquired"Date Acquired",

  2  (Date_Acquired-Hire_Date)"# days"

  3  FROM Employee JOIN Training USING(Emp_Num)

  4                LEFT OUTER JOIN Assignment USING(Emp_Num)

  5  WHERE EXTRACT(YEAR FROM Hire_Date) = EXTRACT(YEAR FROM SYSDATE)-1

  6  GROUP BY Emp_Num,Emp_Last,Emp_First,Hire_Date,Name,Date_acquired

  7  ORDER BY 1;

Emp_Num Name                 Hired Date    # project Training                       Date Acquire     # days

------- -------------------- ----------- ---------- ------------------------------ ----------- ----------

505     Fermi Enrico         04-MAR-16             0 Introduction to Power Point    01-MAY-17          423

508     John Snow            12-AUG-16             2 Upper Level of JAVA            31-AUG-16           19

                                                     Upper Level of R                21-SEP-16           40

513     Franklin Melissa     05-MAR-16             1 Introduction to JAVA           23-OCT-16          232

                                                     Introduction to Python          11-MAY-16           67

                                                     Introduction to SAS             01-JAN-17          302
```

---Q6

Some projects are discontinued for certain period of time and resumed later. Find out project name and start date of those projects that have discontinuous activity. Indicate whether the “discontinued” projects have been completed (i.e., your output should clearly indicate this by “completed” or “on-going” value in a column).

```
SQL> SELECT proj_number "Project ID", NAME "Project Name", START_DATE "Start Date", COMPLETNESS "Completness"
  2  FROM project join (SELECT proj_number, CASE (EXTRACT (YEAR FROM MAX(NVL(DATE_ENDED, SYSDATE))) - EXTRACT(YEAR FROM START_DATE))
  3                                          WHEN 0 THEN (extract (MONTH FROM MAX(NVL(DATE_ENDED, SYSDATE))) -  extract (MONTH FROM START_DATE) + 1)
  4                                          ELSE (extract (MONTH FROM MAX(NVL(DATE_ENDED, SYSDATE))) -  extract (MONTH FROM START_DATE) + 12*(EXTRACT (YEAR
FROM MAX(NVL(DATE_ENDED, SYSDATE))) - EXTRACT(YEAR FROM START_DATE)))
  5                                          END AS "PROJECT_LENGTH",
  6                                       CASE (EXTRACT (YEAR FROM MAX(NVL(DATE_ENDED, SYSDATE))) - EXTRACT(YEAR FROM START_DATE))
  7                                            WHEN 0 THEN COUNT(DISTINCT (EXTRACT(MONTH FROM NVL(DATE_ENDED, SYSDATE))))
  8                                            ELSE COUNT(DISTINCT EXTRACT(MONTH FROM NVL(DATE_ENDED, SYSDATE)) || EXTRACT(YEAR FROM NVL(DATE_ENDED, SYSDATE)))
  9                                            END AS "ASSIGNMENT_LENGTH",
 10                                       CASE MAX(NVL(DATE_ENDED, SYSDATE))
 11                                          WHEN SYSDATE THEN 'ON-GOING'
 12                                          ELSE 'COMPLETED' END AS "COMPLETNESS"
 13              FROM project join assignment using (proj_number)
 14              GROUP BY proj_number, START_DATE)
 15       using (proj_number)
 16  WHERE PROJECT_LENGTH != ASSIGNMENT_LENGTH;

Project ID Project Name                   Start Date           Completness
---------- ------------------------------ -------------------- ---------------
H0908      New Drug Testing               25-AUG-10            COMPLETED
H0988      IPO Evaluation                 22-OCT-13            COMPLETED
H1240      Organization Design            17-MAY-14            COMPLETED
H1255      Product Launch                 01-JUL-16            COMPLETED


4 rows selected.
```

---Q7

We need a summary of project information for each quarter of last year. For each quarter include: number of projects started in the quarter, number of employees working on those projects in the quarter, number of hours spent on average per project.

```
SQL> SELECT START_QUARTER "Quarter", count(DISTINCT T2.proj_number) "# of Projects", NVL(SUM(num_employee), 0) "# of Employee",
  2        NVL(AVG(TOTAL), 0) "Average Hours"
  3  FROM (SELECT proj_number, CASE WHEN EXTRACT(MONTH FROM START_DATE) < 4 THEN '1st quarter'
  4                                 WHEN (6 < EXTRACT(MONTH FROM START_DATE) and EXTRACT(MONTH FROM START_DATE) < 10)
  5                                 THEN '3rd quarter'
  6                                 WHEN 9 < EXTRACT(MONTH FROM START_DATE) THEN '4th quarter'
  7                                 else '2nd quarter'
  8                                 END AS "START_QUARTER"
  9                        FROM project JOIN ASSIGNMENT USING (PROJ_NUMBER)
 10                        GROUP BY PROJ_NUMBER, START_DATE) T1
 11                        LEFT OUTER JOIN (SELECT proj_number, CASE WHEN EXTRACT(MONTH FROM START_DATE) < 4 THEN '1st quarter'
 12                                 WHEN (6 < EXTRACT(MONTH FROM START_DATE) and EXTRACT(MONTH FROM START_DATE) < 10)
 13                                 THEN '3rd quarter'
 14                                 WHEN 9 < EXTRACT(MONTH FROM START_DATE) THEN '4th quarter'
 15                                 else '2nd quarter'
 16                                 END AS "START_QUARTER",
 17                              count(DISTINCT EMP_num) num_employee,
 18                              SUM(HOURS_USED) TOTAL
 19                        FROM project JOIN ASSIGNMENT USING (PROJ_NUMBER)
 20                        WHERE EXTRACT(YEAR FROM START_DATE) = (EXTRACT(YEAR FROM SYSDATE) - 1)
 21                        GROUP BY PROJ_NUMBER, START_DATE) T2 USING (START_QUARTER)
 22  GROUP BY START_QUARTER
 23  ORDER BY START_QUARTER;
```

Quarter	# of Projects	# of Employee	Average Hours
1st quarter	0	0	0
2nd quarter	0	0	0
3rd quarter	3	64	112
4th quarter	1	3	21

4 rows selected.

---Q8

For each employee and each skill, list the number of times the employee had training and the latest date (most recent) when the employee acquired that particular skill. Also, provide the number of trainings provided/attended for each skill, and the number of skills acquired by each employee. The output should look like this:

```
SQL> COLUMN "Employee Name" FORMAT A20
SQL> COLUMN "java" FORMAT 99
SQL> COLUMN "R" FORMAT 99
SQL> COLUMN "PTN" FORMAT 99
SQL> COLUMN "EXL" FORMAT 99
SQL> COLUMN "SAS" FORMAT 99
SQL> COLUMN "PPT" FORMAT 99
SQL> COLUMN "Latest Acquired Date" HEADING "Latest |Acquired |Date" FORMAT A9
SQL>
SQL> select ID, "EMPLOYEE NAME", PTN, PYTHON_DATE "Latest Acquired Date", R, r_date "Latest Acquired Date",
  2         "PPT", pp_date "Latest Acquired Date", java, JAVA_date "Latest Acquired Date",
  3         "EXL", EXCEL_date "Latest Acquired Date",
  4          "SAS" , SAS_date "Latest Acquired Date", "# of Skills"
  5  from (SELECT DECODE(emp_num, NULL, '       ', emp_num) "ID",
  6         DECODE("Employee Name", NULL, 'Number of Trainings:', "Employee Name") "EMPLOYEE NAME",
  7         SUM(DECODE (CODE, 0001, 1, 0)) "PTN",
  8         SUM(DECODE (CODE, 0002, 1, 0)) "R",
  9         SUM(DECODE (CODE, 0003, 1, 0)) "PPT",
 10         SUM(DECODE (CODE, 0004, 1, 0)) "JAVA", SUM(DECODE (CODE, 0005, 1, 0)) "EXL",
 11         SUM(DECODE (CODE, 0006, 1, 0)) "SAS",
```

```
12         COUNT(DISTINCT CODE) "# of Skills"
13  FROM (SELECT EMP_FIRST ||' '|| EMP_LAST "Employee Name", EMP_NUM
14       FROM EMPLOYEE) join training using (emp_num) join skill using (code)
15  GROUP BY GROUPING SETS (("Employee Name", EMP_NUM), ())) t1
16  LEFT OUTER join (select emp_num, MAX(DATE_ACQUIRED) python_date
17       from training
18       where code = 0001
19       group by emp_num)t2 on (t2.emp_num = t1.ID)
20  LEFT OUTER join (select emp_num, MAX(DATE_ACQUIRED) r_date
21       from training
22       where code = 0002
23       group by emp_num)t3 on (t3.emp_num = t1.ID)
24  LEFT OUTER join (select emp_num, MAX(DATE_ACQUIRED) pp_date
25       from training
26       where code = 0003
27       group by emp_num)t4 on (t4.emp_num = t1.ID)
28  LEFT OUTER join (select emp_num, MAX(DATE_ACQUIRED) JAVA_date
29       from training
30       where code = 0004
31       group by emp_num)t5 on (t5.emp_num = t1.ID)
32  LEFT OUTER join (select emp_num, MAX(DATE_ACQUIRED) EXCEL_date
33       from training
34       where code = 0005
35       group by emp_num)t6 on (t6.emp_num = t1.ID)
36  LEFT OUTER join (select emp_num, MAX(DATE_ACQUIRED) SAS_date
```

```
 37       from training
 38       where code = 0006
 39       group by emp_num)t7 on (t7.emp_num = t1.ID)
 40  ORDER BY ID DESC NULLS LAST;
```

ID	EMPLOYEE NAME	PTN	Latest Acquired Date	R	Latest Acquired Date	PPT	Latest Acquired Date	JAVA	Latest Acquired Date	EXL	Latest Acquired Date	SAS	Latest Acquired Date	# of Skills
515	Nicolaus Copernicus	0		0		0		0		0		1	06-NOV-16	1
513	Melissa Franklin	1	11-MAY-16	0		0		1	23-OCT-16	0		1	01-JAN-17	3
512	Michael Faraday	0		0		0		0		0		1	23-DEC-15	1
510	Johannes Kepler	0		0		0		0		2	21-OCT-17	0		1
509	Ingrid Daubechies	0		0		0		0		1	30-JAN-14	0		1
508	Snow John	0		1	21-SEP-16	0		1	31-AUG-16	0		0		2
507	Dorothy Hodgkin	0		0		0		1	23-AUG-15	0		0		1
506	Geraldine Seydoux	0		0		1	08-JUN-17	0		0		0		1
505	Enrico Fermi	0		0		1	01-MAY-17	0		0		0		1
502	Caroline Herschel	1	01-FEB-16	2	21-OCT-16	0		0		0		0		2
501	Blaise Pascal	1	11-JAN-15	0		0		0		0		0		1
	Number of Trainings:	3		3		2		3		3		3		6

12 rows selected.

---Q9

For each department and each skill combination, list the number of trainings completed within the department that was associated with the particular skill. Also provide a rank of each skill within each department. The rank should be based on the number of trainings completed for that skill. Same rank should be given when the number of trainings is the same.

```
SQL> COLUMN "Department" FORMAT A20

SQL> COLUMN "SKILL" FORMAT A15

SQL> COLUMN "Rank" FORMAT 99

SQL> BREAK ON "Department"

SQL> SELECT  NM2 "Department", AA "Skill", NVL(TIMES, 0) "Training Times",
  2          RANK () OVER (partition by NM2 ORDER BY TIMES DESC NULLS LAST) AS "Rank"
  3  FROM (SELECT T1.NAME NM1, T1.DEPT_CODE, T5.DESCRIPTION TT, COUNT (DISTINCT TRAIN_NUM)  Times
  4        FROM DEPARTMENT T1  JOIN  EMPLOYEE T3 ON (T1.DEPT_CODE  =T3.DEPT_CODE)
  5             JOIN Training T4  ON (T4.Emp_Num=T3.Emp_Num) join skill t5 ON (T4.CODE=T5.CODE)
  6        GROUP BY T1.NAME,T5. DESCRIPTION, T1.DEPT_CODE) RIGHT OUTER JOIN
  7       (SELECT T6.NAME NM2, T7.DESCRIPTION AA FROM DEPARTMENT T6 CROSS JOIN SKILL T7  ) ON NM2 = NM1 AND AA = TT
  8  ORDER BY 1, 3 desc;
```

Department	Skill	Training Times	Rank
Administration	Python	0	1
	R	0	1
	JAVA	0	1
	Excel	0	1
	Power Point	0	1
	SAS	0	1
Design	SAS	1	1
	R	0	2
	Excel	0	2
	JAVA	0	2
	Python	0	2
	Power Point	0	2
IT	SAS	2	1
	JAVA	1	2
	Python	1	2
	Power Point	0	4
	Excel	0	4
	R	0	4
Marketing	Excel	3	1
	JAVA	1	2
	R	0	3
	SAS	0	3
	Power Point	0	3
	Python	0	3
Strategy Development	R	3	1

```
Power Point                2    2

Python                     2    2

JAVA                       1    4

SAS                        0    5

Excel                      0    5          30 rows selected.
```