Islamic University of Gaza
Faculty of Engineering
Computer Department

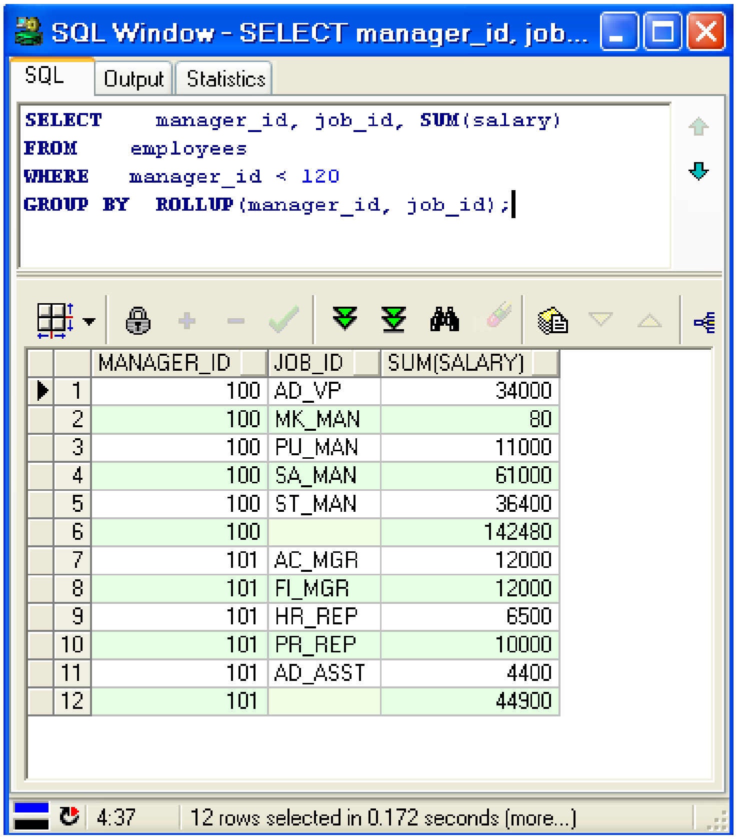
Advanced DataBase ECOM 5054
Instructor: Eng. Husam Alzaq
T.A: Eng. Doaa KH. Abu Jabal

Oracle LAB 8

Exercise 1 :

Write a query to display the following for those employees whose manager ID is less than 120:
- Manager ID
- Job ID and total salary for every job ID for employees who report to the same manager
- Total salary of those managers
- Total salary of those managers, irrespective of the job IDs

The Result:



```
SELECT    manager_id, job_id, SUM(salary)
FROM     employees
WHERE    manager_id < 120
GROUP BY  ROLLUP(manager_id, job_id);
```

	MANAGER_ID	JOB_ID	SUM(SALARY)
1	100	AD_VP	34000
2	100	MK_MAN	80
3	100	PU_MAN	11000
4	100	SA_MAN	61000
5	100	ST_MAN	36400
6	100		142480
7	101	AC_MGR	12000
8	101	FI_MGR	12000
9	101	HR_REP	6500
10	101	PR_REP	10000
11	101	AD_ASST	4400
12	101		44900

4:37   12 rows selected in 0.172 seconds (more...)

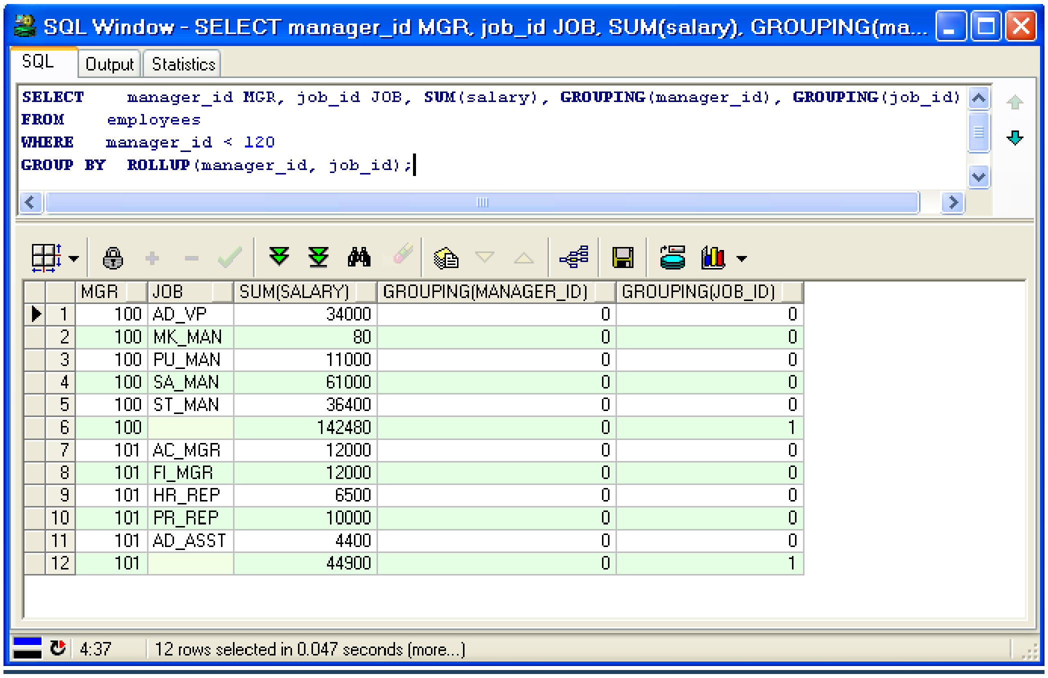
Islamic University of Gaza
Faculty of Engineering
Computer Department

Advanced DataBase ECOM 5054
Instructor: Eng. Husam Alzaq
T.A: Eng. Doaa KH. Abu Jabal

Exercise 2 :

Observe the output from question 1. Write a query using the `GROUPING` function to determine whether the `NULL` values in the columns corresponding to the `GROUP BY` expressions are caused by the `ROLLUP` operation.

The Result:


SQL Window - SELECT manager_id MGR, job_id JOB, SUM(salary), GROUPING(ma...
SQL
Output
Statistics

```
SELECT    manager_id MGR, job_id JOB, SUM(salary), GROUPING(manager_id), GROUPING(job_id)
FROM      employees
WHERE     manager_id < 120
GROUP BY  ROLLUP(manager_id, job_id);
```

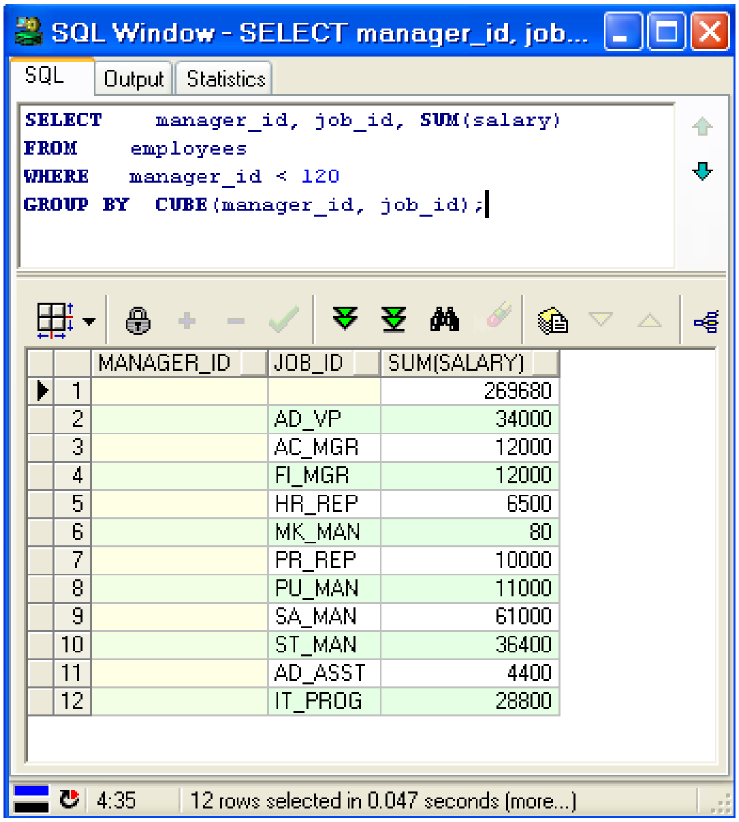
	MGR	JOB	SUM(SALARY)	GROUPING(MANAGER_ID)	GROUPING(JOB_ID)
1	100	AD_VP	34000	0	0
2	100	MK_MAN	80	0	0
3	100	PU_MAN	11000	0	0
4	100	SA_MAN	61000	0	0
5	100	ST_MAN	36400	0	0
6	100		142480	0	1
7	101	AC_MGR	12000	0	0
8	101	FI_MGR	12000	0	0
9	101	HR_REP	6500	0	0
10	101	PR_REP	10000	0	0
11	101	AD_ASST	4400	0	0
12	101		44900	0	1

4:37   12 rows selected in 0.047 seconds (more...)

Exercise 3:

Write a query to display the following for those employees whose manager ID is less than 120:
- Manager ID
- Job and total salaries for every job for employees who report to the same manager
- Total salary of those managers
- Cross-tabulation values to display the total salary for every job, irrespective of the manager
- Total salary irrespective of all job titles

The Result:



```
SELECT     manager_id, job_id, SUM(salary)
FROM     employees
WHERE    manager_id < 120
GROUP BY   CUBE(manager_id, job_id);
```

	MANAGER_ID	JOB_ID	SUM(SALARY)
1			269680
2		AD_VP	34000
3		AC_MGR	12000
4		FI_MGR	12000
5		HR_REP	6500
6		MK_MAN	80
7		PR_REP	10000
8		PU_MAN	11000
9		SA_MAN	61000
10		ST_MAN	36400
11		AD_ASST	4400
12		IT_PROG	28800

4:35 12 rows selected in 0.047 seconds (more...)

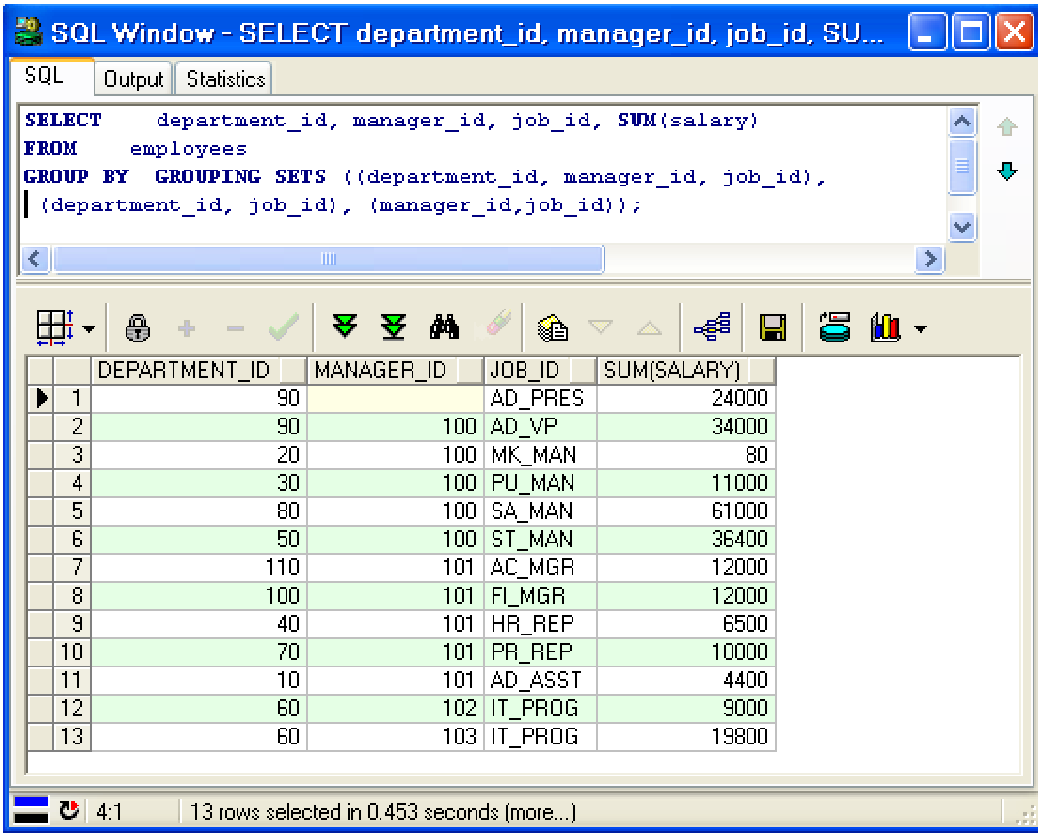
Exercise 4:

Using `GROUPING SETS`, write a query to display the following groupings:

- `department_id, manager_id, job_id`
- `department_id, job_id`
- `manager_id, job_id`

The query should calculate the sum of the salaries for each of these groups.

The Result:


SQL Window - SELECT department_id, manager_id, job_id, SU...
SQL
Output
Statistics

```
SELECT     department_id, manager_id, job_id, SUM(salary)
FROM     employees
GROUP BY   GROUPING SETS ((department_id, manager_id, job_id),
 (department_id, job_id), (manager_id,job_id));
```

	DEPARTMENT_ID	MANAGER_ID	JOB_ID	SUM(SALARY)
1	90		AD_PRES	24000
2	90	100	AD_VP	34000
3	20	100	MK_MAN	80
4	30	100	PU_MAN	11000
5	80	100	SA_MAN	61000
6	50	100	ST_MAN	36400
7	110	101	AC_MGR	12000
8	100	101	FI_MGR	12000
9	40	101	HR_REP	6500
10	70	101	PR_REP	10000
11	10	101	AD_ASST	4400
12	60	102	IT_PROG	9000
13	60	103	IT_PROG	19800

4:1 13 rows selected in 0.453 seconds (more...)