i2i Systems

innovation to integration

i2i Academy

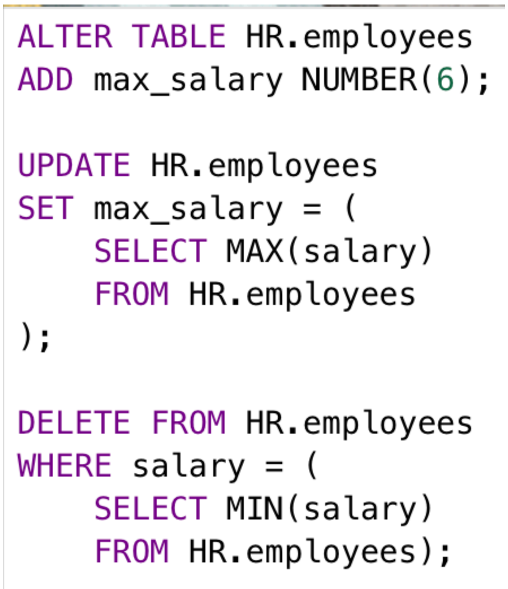
Topic	Oracle SQL Language Fundamentals I
Document Name	SQL03-EX-01-05

Exercise SQL03-EX-01:

Definiton : Write followig SQL queries:

- Add a colum to employees table named MAX_SALARY.
- Update MAX_SALARY with maximum salary amount with subquery.
- Delete employee who have minimum salary using subquery.

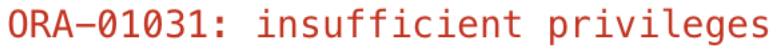
SQL:



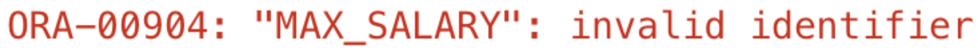
```
ALTER TABLE HR.employees
ADD max_salary NUMBER(6);

UPDATE HR.employees
SET max_salary = (
    SELECT MAX(salary)
    FROM HR.employees
);

DELETE FROM HR.employees
WHERE salary = (
    SELECT MIN(salary)
    FROM HR.employees);
```

Screenshot:

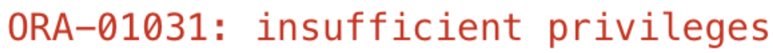
I am using live SQL so, this happens when I try to alter table:



```
ORA-01031: insufficient privileges
```



```
ORA-00904: "MAX_SALARY": invalid identifier
```



```
ORA-01031: insufficient privileges
```

Exercise SQL03-EX-02:

Definiton : Write followig SQL queries:

- Define index (named DPR_NAME_IDX) on DEPARTMENT_NAME column of DEPARTMENTS table.
- Define constraint (named CNSTR_SALARY) on employee salary. (Salary must be between 1000$ and 100.000$)
- Drop defined index.
- Enable, disable, drop defined constraint.

SQL:

```
CREATE INDEX DPR_NAME_IDX
ON Departments.department_name;

ALTER TABLE HR.employees
ADD CONSTRAINT CNSTR_SALARY salary < 100000 AND salary > 1000;

DROP INDEX DPR_NAME_IDX;

ALTER TABLE Departments.department_name
DISABLE CONSTRAINT CNSTR_SALARY;

ALTER TABLE Departments.department_name
ENABLE CONSTRAINT CNSTR_SALARY;

ALTER TABLE Departments.department_name
DROP CONSTRAINT CNSTR_SALARY;
```

Screenshot:

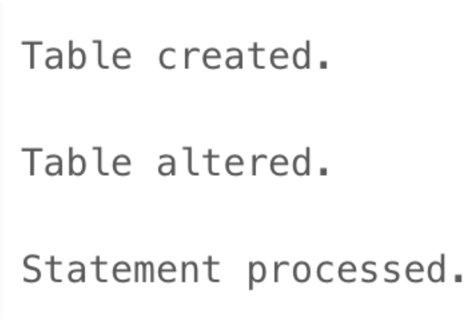
Exercise SQL03-EX-03:

Definiton : Create a table from EMPLOYEES with distinct department_id column. Add department_name to that table. With DEPARTMENTS table, update department_name for included department_ids and insert department_id and department_name values for not included rows. Use MERGE keyword.

SQL:

```sql
CREATE TABLE temp
AS (SELECT DISTINCT department_id
    FROM HR.employees);

ALTER TABLE temp
ADD department_name VARCHAR(20);
MERGE INTO temp t
USING (
    SELECT department_id, department_name
    FROM HR.departments) d
ON (t.department_id = d.department_id)
WHEN MATCHED THEN
    UPDATE SET t.department_name = d.department_name
WHEN NOT MATCHED THEN
    INSERT (t.department_id, t.department_name)
    VALUES (d.department_id, d.department_name);
```

Screenshot:



```
Table created.

Table altered.

Statement processed.
```

Exercise SQL03-EX-04:

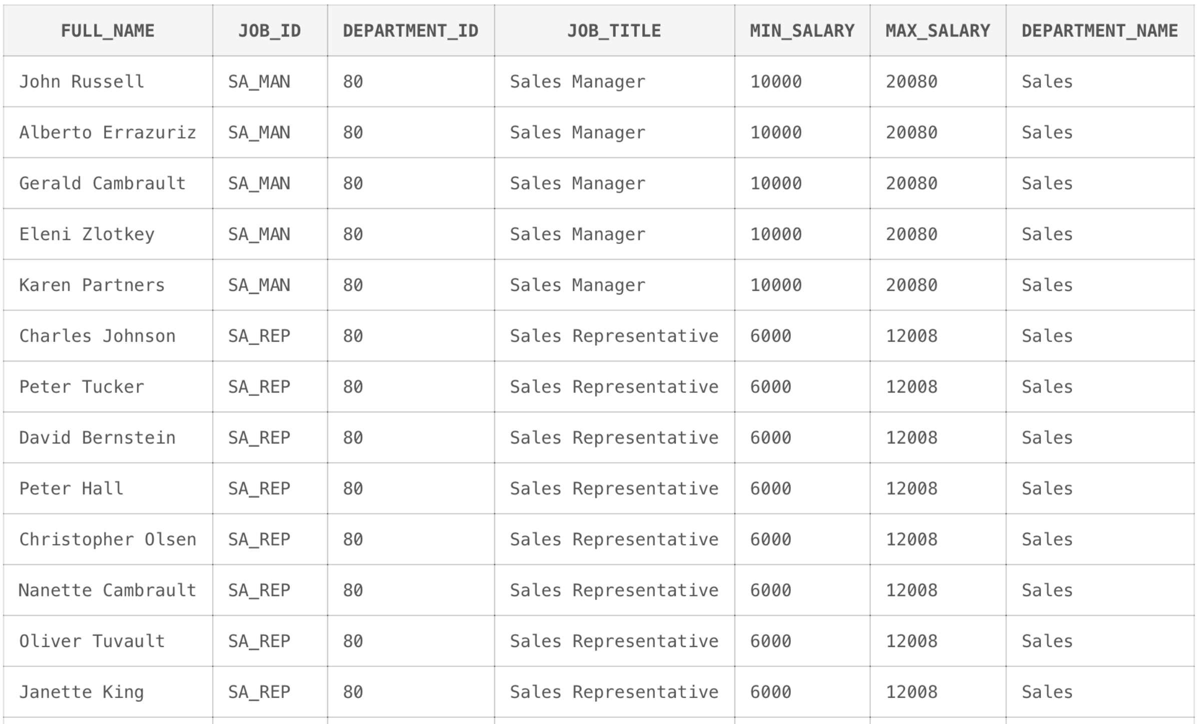
Definiton : Using **WITH** keyword, do following jobs:

- Firstly select first_name, last_name, job_id, department_id from employees table whoes job_id starts with 'S'.
- Additionally select job_title and min-max salary amount.
- Add department_name to that query.
- Lastly concat first_name and last_name with space as full_name alias and list with other selected columns.

SQL:

```
WITH EmpJobStartingWithS AS (
  SELECT first_name, last_name, job_id, department_id
  FROM hr.employees
  WHERE job_id LIKE 'S%'
)
SELECT (a.first_name || ' ' || a.last_name) as full_name, a.job_id, a.department_id,
m.job_title, m.min_salary, m.max_salary, x.department_name
FROM EmpJobStartingWithS a
INNER JOIN hr.jobs m ON a.job_id = m.job_id
INNER JOIN hr.departments x ON a.department_id = x.department_id;
```

Screenshot:



FULL_NAME	JOB_ID	DEPARTMENT_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY	DEPARTMENT_NAME
John Russell	SA_MAN	80	Sales Manager	10000	20080	Sales
Alberto Errazuriz	SA_MAN	80	Sales Manager	10000	20080	Sales
Gerald Cambrault	SA_MAN	80	Sales Manager	10000	20080	Sales
Eleni Zlotkey	SA_MAN	80	Sales Manager	10000	20080	Sales
Karen Partners	SA_MAN	80	Sales Manager	10000	20080	Sales
Charles Johnson	SA_REP	80	Sales Representative	6000	12008	Sales
Peter Tucker	SA_REP	80	Sales Representative	6000	12008	Sales
David Bernstein	SA_REP	80	Sales Representative	6000	12008	Sales
Peter Hall	SA_REP	80	Sales Representative	6000	12008	Sales
Christopher Olsen	SA_REP	80	Sales Representative	6000	12008	Sales
Nanette Cambrault	SA_REP	80	Sales Representative	6000	12008	Sales
Oliver Tuvault	SA_REP	80	Sales Representative	6000	12008	Sales
Janette King	SA_REP	80	Sales Representative	6000	12008	Sales

Exercise SQL03-EX-05:

Definiton : Search for COMMIT and ROLLBACK keywords and explain them.

COMMIT in SQL is a transaction control language that is used to permanently save the changes done in the transaction in tables/databases. The database cannot regain its previous state after its execution of commit.

ROLLBACK in SQL is a transactional control language that is used to undo the transactions that have not been saved in the database. The command is only been used to undo changes since the last COMMIT.