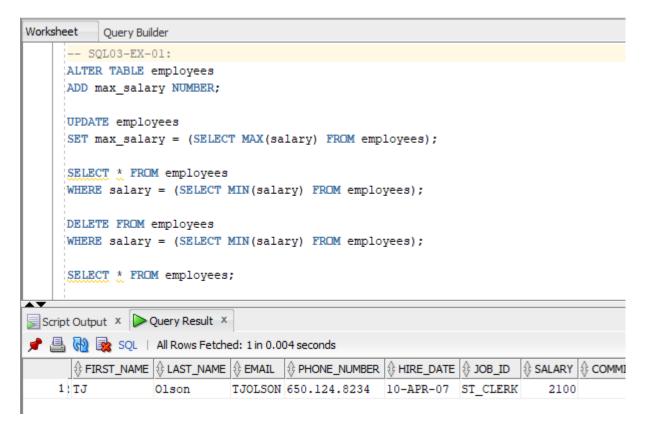
Topic	Oracle SQL Language Fundamentals I
Document Name	SQL03-EX-01-05
1	Enes Tahtacı

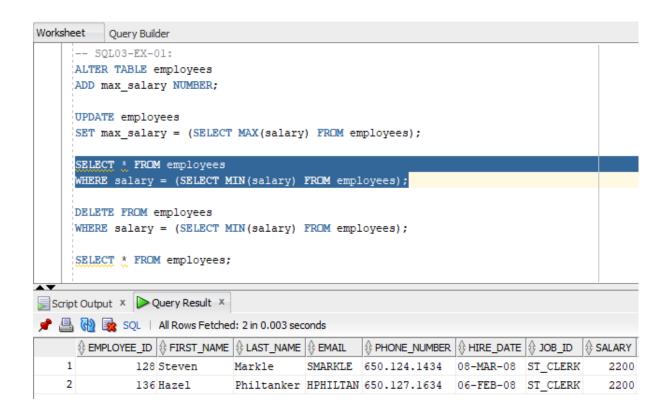
Exercise SQL03-EX-01:

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.

Screenshots:





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.

Screenshot:

```
-- SQL03-EX-02:
CREATE INDEX dpr_name_idx
ON departments (department_name);

ALTER TABLE employees
ADD CONSTRAINT cnstr_salary
CHECK (salary BETWEEN 1000 AND 100000);

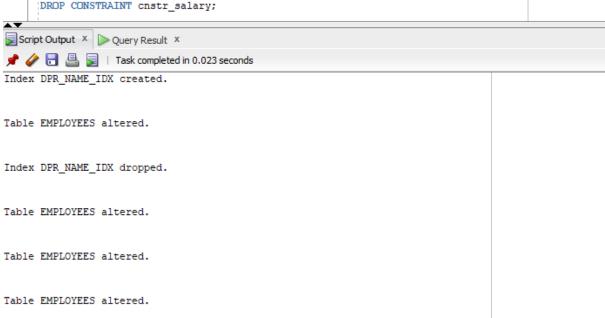
DROP INDEX dpr_name_idx;

ALTER TABLE employees
ENABLE CONSTRAINT cnstr_salary;

ALTER TABLE employees
DISABLE CONSTRAINT cnstr_salary;

ALTER TABLE employees
DISABLE constraint cnstr_salary;

ALTER TABLE employees
DROP CONSTRAINT cnstr_salary;
```



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.

Screenshot:

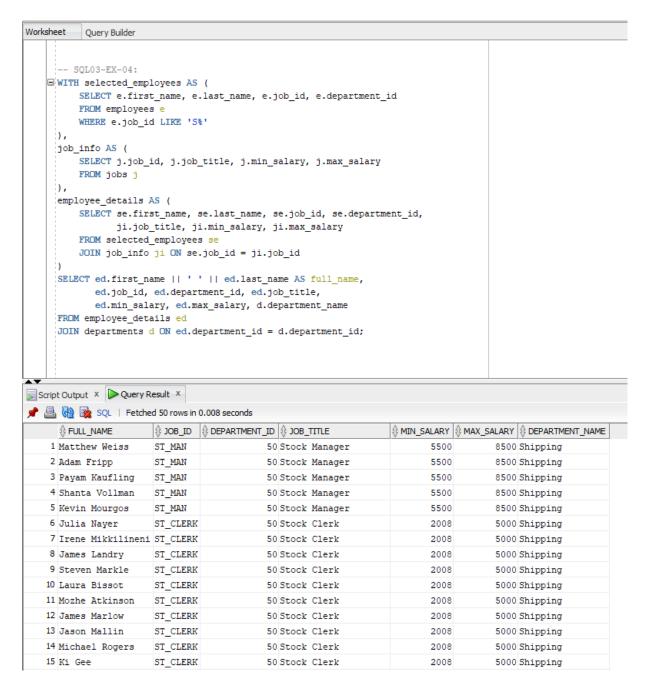
```
Worksheet
         Query Builder
     -- SQL03-EX-03:
     CREATE TABLE emp2 AS
     SELECT DISTINCT department_id
     FROM employees;
     ALTER TABLE emp2 ADD (
         department_name VARCHAR2(30)
     1);
   □ UPDATE emp2
     SET department_name = (SELECT department_name
                            FROM departments
                            WHERE emp2.department_id = departments.department_id)
     WHERE department_id IN (SELECT department_id FROM emp2);
     SELECT department_name,department_id FROM departments;
   ⊞ MERGE INTO emp2 e
     USING departments d
     ON (e.department_id = d.department_id)
     WHEN MATCHED THEN
         UPDATE SET e.department_name = d.department_name
     WHEN NOT MATCHED THEN
         INSERT (e.department_id, e.department_name)
         VALUES (d.department_id, d.department_name);
Script Output × Query Result ×
📌 🚇 🙀 🗽 SQL | All Rows Fetched: 28 in 0.002 seconds
      50 Shipping
                  40 Human Resources
    3
                 110 Accounting
    4
                  90 Executive
    5
                  30 Purchasing
    6
                  70 Public Relations
    7
              (null) (null)
    8
                  10 Administration
    9
                  20 Marketing
   10
                  60 IT
   11
                 100 Finance
   12
                 80 Sales
   13
                 210 IT Support
   14
                 200 Operations
   15
                 270 Payroll
```

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.

Screenshot:



Exercise SQL03-EX-05:

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

1. COMMIT

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.

2. ROLLBACK

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.

	сомміт	ROLLBACK
1.	COMMIT permanently saves the changes made by the current transaction.	ROLLBACK undo the changes made by the current transaction.
2.	The transaction can not undo changes after COMMIT execution.	Transaction reaches its previous state after ROLLBACK.
3.	When the transaction is successful, COMMIT is applied.	When the transaction is aborted, incorrect execution, system failure ROLLBACK occurs.
4.	COMMIT statement permanently save the state, when all the statements are executed successfully without any error.	In ROLLBACK statement if any operations fail during the completion of a transaction, it cannot permanently save the change and we can undo them using this statement.
5.	Syntax of COMMIT statement are:	Syntax of ROLLBACK statement are:
	COMMIT;	ROLLBACK;