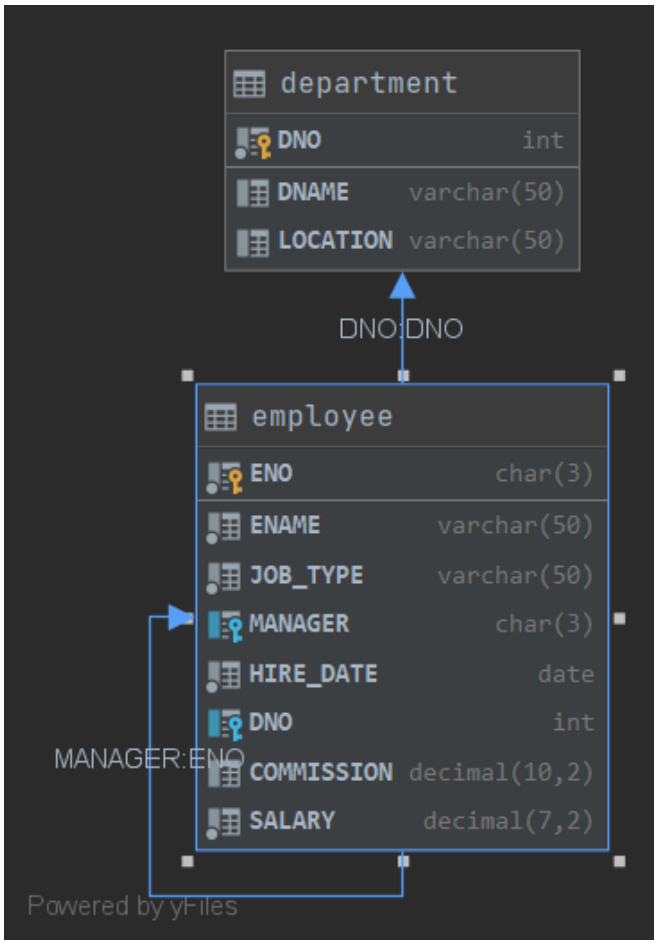
Creating Schema – **COMPANYDB**



creatingSchemas.sql

```
CREATE SCHEMA COMPANYDB;
USE COMPANYDB;

-- EMPLOYEE TABLE
CREATE TABLE EMPLOYEE(
    ENO CHAR(3),
    ENAME VARCHAR(50) NOT NULL,
    JOB_TYPE VARCHAR(50) NOT NULL,
    MANAGER CHAR(3),
    HIRE_DATE DATE NOT NULL,
```

```
    DNO INT,
    COMMISSION DECIMAL(10,2),
    SALARY DECIMAL(7,2) NOT NULL,
    PRIMARY KEY(ENO)
);

-- DEPARTMENT TABLE
CREATE TABLE DEPARTMENT(
    DNO INT NOT NULL,
    DNAME VARCHAR(50),
    LOCATION VARCHAR(50) DEFAULT 'New Delhi',
    PRIMARY KEY(DNO)
);


-- POPULATING DEPARTMENT TABLE
INSERT INTO DEPARTMENT(DNO,DNAME,LOCATION) VALUES
    (1,'Research','Chennai'),
    (2,'Adminstration','New Delhi'),
    (7,'Headquaters','New Delhi');

-- POPULATING EMPLOYEE TABLE
INSERT INTO EMPLOYEE VALUES
    (001, 'Georgi Facello', 'Senior Engineer', NULL, '1986-06-
26', 1, 580, 60117),
    (911, 'Shay Casley', 'Senior Engineer', NULL, '1988-06-
25', 1, 555, 66074),
    (667, 'Makato Cusworth', 'Senior Staff', NULL, '1990-06-
25', 2, 736, 66961),
    (004, 'Chirstian Koblick', 'Engineer', NULL, '1986-12-
01', 1, NULL, 40054),
    (339, 'Tenius Liedekerke', 'Engineer', NULL, '1989-11-
30', 1, 200, 46065),
    (005, 'Kyoichi Maliniak', 'Staff', NULL, '1989-09-
12', 2, NULL, 58326),
    (882, 'Fun Heuser', 'CEO', NULL, '1994-11-29' ,7 ,1500, 78228);


-- ASSIGNING MANAGER
```

```
UPDATE EMPLOYEE SET MANAGER=667 WHERE JOB_TYPE = 'Staff';
UPDATE EMPLOYEE SET MANAGER=001 WHERE JOB_TYPE = 'Engineer';
UPDATE EMPLOYEE SET MANAGER=882 WHERE JOB_TYPE = 'Senior Engineer';
UPDATE EMPLOYEE SET MANAGER=882 WHERE JOB_TYPE = 'Senior Staff';

-- ALTERING THE CREATED TABLE TO ADD FOREIGN KEYS
ALTER TABLE EMPLOYEE ADD FOREIGN KEY(DNO) REFERENCES DEPARTMENT(DNO);
ALTER TABLE EMPLOYEE ADD FOREIGN KEY(MANAGER) REFERENCES EMPLOYEE(ENO)
;
```

Resulting Tables:

```
mysql> USE COMPANYDB;
Database changed
mysql> SHOW COLUMNS FROM EMPLOYEE;
+------------+---------------+------+-----+---------+-------+
| Field      | Type          | Null | Key | Default | Extra |
+------------+---------------+------+-----+---------+-------+
| ENO        | char(3)       | NO   | PRI | NULL    |       |
| ENAME      | varchar(50)   | NO   |     | NULL    |       |
| JOB_TYPE   | varchar(50)   | NO   |     | NULL    |       |
| MANAGER    | char(3)       | YES  | MUL | NULL    |       |
| HIRE_DATE  | date          | NO   |     | NULL    |       |
| DNO        | int           | YES  | MUL | NULL    |       |
| COMMISSION | decimal(10,2) | YES  |     | NULL    |       |
| SALARY     | decimal(7,2)  | NO   |     | NULL    |       |
+------------+---------------+------+-----+---------+-------+
8 rows in set (0.12 sec)

mysql> SHOW COLUMNS FROM DEPARTMENT;
+----------+-------------+------+-----+-----------+-------+
| Field    | Type        | Null | Key | Default   | Extra |
+----------+-------------+------+-----+-----------+-------+
| DNO      | int         | NO   | PRI | NULL      |       |
| DNAME    | varchar(50) | YES  |     | NULL      |       |
| LOCATION | varchar(50) | YES  |     | New Delhi |       |
+----------+-------------+------+-----+-----------+-------+
3 rows in set (0.00 sec)

mysql>
```

Queries:

21. Query to display Name, Hire Date and Day of the week on which the employee started.

Query

```
SELECT ENAME,HIRE_DATE,DAYNAME(HIRE_DATE) AS DAY
FROM EMPLOYEE;
```

Output

```
mysql> SELECT ENAME,HIRE_DATE,DAYNAME(HIRE_DATE) AS DAY
    -> FROM EMPLOYEE;
+-------------------+------------+-----------+
| ENAME             | HIRE_DATE  | DAY       |
+-------------------+------------+-----------+
| Georgi Facello    | 1986-06-26 | Thursday  |
| Tenius Liedekerke | 1989-11-30 | Thursday  |
| Chirstian Koblick | 1986-12-01 | Monday    |
| Kyoichi Maliniak  | 1989-09-12 | Tuesday   |
| Makato Cusworth   | 1990-06-25 | Monday    |
| Fun Heuser        | 1994-11-29 | Tuesday   |
| Shay Casley       | 1988-06-25 | Saturday  |
+-------------------+------------+-----------+
7 rows in set (0.00 sec)

mysql>
```

22. Query to display Name, Department Name and Department No for all the employees.

Query

```
SELECT E.ENAME,D.DNAME,D.DNO
FROM EMPLOYEE AS E JOIN DEPARTMENT AS D
ON E.DNO = D.DNO;
```

Output

```
mysql> SELECT E.ENAME,D.DNAME,D.DNO
    -> FROM EMPLOYEE AS E JOIN DEPARTMENT AS D
    -> ON E.DNO = D.DNO;
+-------------------+---------------+-----+
| ENAME             | DNAME         | DNO |
+-------------------+---------------+-----+
| Georgi Facello    | Research      |   1 |
| Tenius Liedekerke | Research      |   1 |
| Chirstian Koblick | Research      |   1 |
| Shay Casley       | Research      |   1 |
| Kyoichi Maliniak  | Adminstration |   2 |
| Makato Cusworth   | Adminstration |   2 |
| Fun Heuser        | Headquaters   |   7 |
+-------------------+---------------+-----+
7 rows in set (0.07 sec)

mysql>
```

23. Query to display Unique Listing of all Jobs that are in Department #1.

Query

```
SELECT DISTINCT E.JOB_TYPE
FROM EMPLOYEE AS E
WHERE E.DNO = 1;
```

Output

```
mysql> SELECT DISTINCT E.JOB_TYPE
    -> FROM EMPLOYEE AS E
    -> WHERE E.DNO = 1;
+-----------------+
| JOB_TYPE        |
+-----------------+
| Senior Engineer |
| Engineer        |
+-----------------+
2 rows in set (0.00 sec)

mysql>
```

24. Query to display Name, Dept Name of all employees who have an ‘A’ in their name.

Query

```
SELECT E.ENAME,D.DNAME
FROM EMPLOYEE AS E JOIN DEPARTMENT AS D ON E.DNO = D.DNO
WHERE E.ENAME LIKE '%A%';
```

Output

```
mysql> SELECT E.ENAME,D.DNAME
    -> FROM EMPLOYEE AS E JOIN DEPARTMENT AS D ON E.DNO = D.DNO
    -> WHERE E.ENAME LIKE '%A%';
+-------------------+---------------+
| ENAME             | DNAME         |
+-------------------+---------------+
| Georgi Facello    | Research      |
| Chirstian Koblick | Research      |
| Kyoichi Maliniak  | Adminstration |
| Makato Cusworth   | Adminstration |
| Shay Casley       | Research      |
+-------------------+---------------+
5 rows in set (0.00 sec)

mysql>
```

25. Query to display Name, Job, Department No. And Department Name for all the employees working at the New Delhi location

Query

```
SELECT E.ENO,E.ENAME,E.JOB_TYPE,E.DNO,D.DNAME
FROM EMPLOYEE AS E JOIN DEPARTMENT AS D ON E.DNO = D.DNO
WHERE D.LOCATION = 'New Delhi';
```

Output

```
mysql> SELECT E.ENO,E.ENAME,E.JOB_TYPE,E.DNO,D.DNAME
    -> FROM EMPLOYEE AS E JOIN DEPARTMENT AS D ON E.DNO = D.DNO
    -> WHERE D.LOCATION = 'New Delhi';
+-----+------------------+--------------+------+---------------+
| ENO | ENAME            | JOB_TYPE     | DNO  | DNAME         |
+-----+------------------+--------------+------+---------------+
| 5   | Kyoichi Maliniak | Staff        |    2 | Adminstration |
| 667 | Makato Cusworth  | Senior Staff |    2 | Adminstration |
| 882 | Fun Heuser       | CEO          |    7 | Headquaters   |
+-----+------------------+--------------+------+---------------+
3 rows in set (0.00 sec)
```

26. Query to display Name and Employee no. Along with their Manger's Name and the Manager's employee no; along with the Employees' Name who do not have a Manager.

Query

```
SELECT E.ENO,E.ENAME,S.ENO,S.ENAME
FROM EMPLOYEE AS E LEFT JOIN EMPLOYEE AS S on E.MANAGER = S.ENO;
```

Output

```
mysql> SELECT E.ENO,E.ENAME,S.ENO,S.ENAME
    -> FROM EMPLOYEE AS E LEFT JOIN EMPLOYEE AS S on E.MANAGER = S.ENO;
+-----+-------------------+------+-----------------+
| ENO | ENAME             | ENO  | ENAME           |
+-----+-------------------+------+-----------------+
| 1   | Georgi Facello    | 882  | Fun Heuser      |
| 339 | Tenius Liedekerke | 1    | Georgi Facello  |
| 4   | Chirstian Koblick | 1    | Georgi Facello  |
| 5   | Kyoichi Maliniak  | 667  | Makato Cusworth |
| 667 | Makato Cusworth   | 882  | Fun Heuser      |
| 882 | Fun Heuser        | NULL | NULL            |
| 911 | Shay Casley       | 882  | Fun Heuser      |
+-----+-------------------+------+-----------------+
7 rows in set (0.00 sec)

mysql>
```

27. Query to display Name, Dept No. And Salary of any employee whose department No. and salary matches both the department no. And the salary of any employee who earns a commission.

Query

```
SELECT ENAME,DNO,SALARY
FROM EMPLOYEE E
WHERE COMMISSION IN (SELECT COMMISSION FROM EMPLOYEE
                     WHERE E.DNO = DNO AND E.SALARY = SALARY);
```

Output

```
mysql> SELECT ENAME,DNO,SALARY
    -> FROM EMPLOYEE E
    -> WHERE COMMISSION IN (SELECT COMMISSION FROM EMPLOYEE
    ->                      WHERE E.DNO = DNO AND E.SALARY = SALARY);
+-------------------+------+----------+
| ENAME             | DNO  | SALARY   |
+-------------------+------+----------+
| Georgi Facello    |    1 | 60117.00 |
| Tenius Liedekerke |    1 | 46065.00 |
| Makato Cusworth   |    2 | 66961.00 |
| Fun Heuser        |    7 | 78228.00 |
| Shay Casley       |    1 | 66074.00 |
+-------------------+------+----------+
5 rows in set (0.00 sec)

mysql>
```

28. Query to display Name and Salaries represented by asterisks, where each asterisk (*) signifies $100

Query

```
SELECT ENAME,CONCAT(CONVERT(SALARY/100,UNSIGNED INTEGER),'*')
AS SALARY
FROM EMPLOYEE;
```

Output

```
mysql> SELECT ENAME,CONCAT(CONVERT(SALARY/100,UNSIGNED INTEGER),'*')
    -> AS SALARY
    -> FROM EMPLOYEE;
+-------------------+--------+
| ENAME             | SALARY |
+-------------------+--------+
| Georgi Facello    | 601*   |
| Tenius Liedekerke | 461*   |
| Chirstian Koblick | 401*   |
| Kyoichi Maliniak  | 583*   |
| Makato Cusworth   | 670*   |
| Fun Heuser        | 782*   |
| Shay Casley       | 661*   |
+-------------------+--------+
7 rows in set (0.00 sec)

mysql>
```

29. Query to display the Highest, Lowest, Sum and Average Salaries of all the employees.

Query

```
SELECT
MAX(SALARY) AS HIGHEST,
MIN(SALARY) AS LOWEST,
AVG(SALARY) AS AVERAGE,
SUM(SALARY) AS SUM
FROM EMPLOYEE;
```

Output

```
mysql> SELECT
    -> MAX(SALARY) AS HIGHEST,
    -> MIN(SALARY) AS LOWEST,
    -> AVG(SALARY) AS AVERAGE,
    -> SUM(SALARY) AS SUM
    -> FROM EMPLOYEE;
+----------+----------+--------------+-----------+
| HIGHEST  | LOWEST   | AVERAGE      | SUM       |
+----------+----------+--------------+-----------+
| 78228.00 | 40054.00 | 59403.571429 | 415825.00 |
+----------+----------+--------------+-----------+
1 row in set (0.00 sec)

mysql>
```

30. Query to display the number of employees performing the same Job type functions.

Query

```
SELECT JOB_TYPE,COUNT(*) AS NUMBER_OF_EMPLOYEE
FROM EMPLOYEE
GROUP BY JOB_TYPE
```

Output

```
mysql> SELECT JOB_TYPE,COUNT(*) AS NUMBER_OF_EMPLOYEE
    -> FROM EMPLOYEE
    -> GROUP BY JOB_TYPE;
+-----------------+--------------------+
| JOB_TYPE        | NUMBER_OF_EMPLOYEE |
+-----------------+--------------------+
| Senior Engineer |                  2 |
| Engineer        |                  2 |
| Staff           |                  1 |
| Senior Staff    |                  1 |
| CEO             |                  1 |
+-----------------+--------------------+
5 rows in set (0.00 sec)

mysql>
```