Latihan Sistem Basis Data

Ahmad Imanudin

Table of Contents

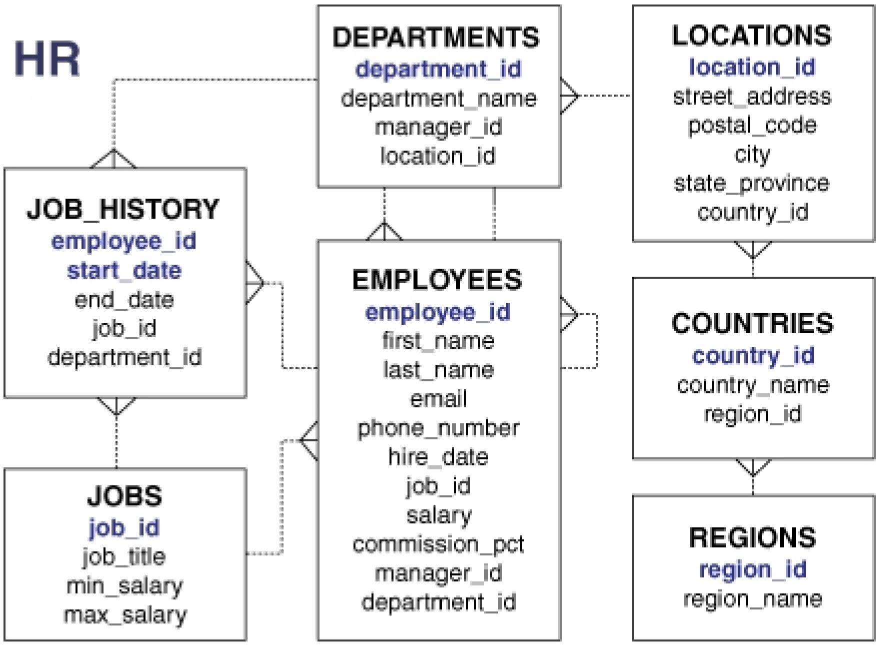

Figure 1. ER Diagram

Basis data yang akan dibuat sesuai dengan ER Diagram diatas

1. Membuat Database

```
mysql -u root -p
create database latihan3;
use latihan3;
```

Contoh tampilan

```
mysql> create database latihan3;
Query OK, 1 row affected (0.00 sec)

mysql> use latihan3;
Database changed
```

2. Membuat Tabel

2.1. Membuat Tabel Regions

```
CREATE TABLE regions(
region_id int(5) not null,
region_name varchar(50),
constraint regions_pk primary key (region_id)
);
```

Contoh tampilan

```
mysql> CREATE TABLE regions(
    -> region_id int(5) not null,
    -> region_name varchar(50),
    -> constraint regions_pk primary key (region_id)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

2.2. Membuat Tabel Countries

```
CREATE TABLE countries(
country_id char(5) not null,
country_name varchar(50),
region_id int(5) not null,
constraint country_pk primary key (country_id),
constraint country_fk foreign key (region_id)
references regions (region_id)
);
```

Contoh tampilan

```
mysql> CREATE TABLE countries(
    -> country_id char(5) not null,
    -> country_name varchar(50),
    -> region_id int(5) not null,
    -> constraint country_pk primary key (country_id),
    -> constraint country_fk foreign key (region_id)
    -> references regions (region_id)
    -> );
Query OK, 0 rows affected (0.07 sec)
```

2.3. Membuat Tabel Locations

```
CREATE TABLE locations(
location_id int(5) not null,
street_address varchar(200),
postal_code varchar(5),
city varchar(25),
state_province varchar(25),
country_id char(5) not null,
constraint location_pk primary key (location_id),
constraint location_fk foreign key (country_id)
references countries (country_id)
);
```

Contoh tampilan

```
mysql> CREATE TABLE locations(
    -> location_id int(5) not null,
    -> street_address varchar(200),
    -> postal_code varchar(5),
    -> city varchar(25),
    -> state_province varchar(25),
    -> country_id char(5) not null,
    -> constraint location_pk primary key (location_id),
    -> constraint location_fk foreign key (country_id)
    -> references countries (country_id)
    -> );
Query OK, 0 rows affected (0.06 sec)
```

2.4. Membuat Tabel Departments

```
CREATE TABLE departments(
department_id int(5) not null,
department_name varchar(25),
manager_id int(5),
location_id int(5) not null,
constraint department_pk primary key (department_id),
constraint department_fk foreign key (location_id)
references locations (location_id)
);
```

Contoh tampilan

```
mysql> CREATE TABLE departments(
    -> department_id int(5) not null,
    -> department_name varchar(25),
    -> manager_id int(5),
    -> location_id int(5) not null,
    -> constraint department_pk primary key (department_id),
    -> constraint department_fk foreign key (location_id)
    -> references locations (location_id)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

2.5. Membuat Tabel Jobs

```
CREATE TABLE jobs(
job_id varchar(10) not null,
job_title varchar(25),
min_salary decimal(10,0),
max_salary decimal(15,0),
constraint job_pk primary key (job_id)
);
```

Contoh tampilan

```
mysql> CREATE TABLE jobs(
    -> job_id varchar(10) not null,
    -> job_title varchar(25),
    -> min_salary decimal(10,0),
    -> max_salary decimal(15,0),
    -> constraint job_pk primary key (job_id)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

2.6. Membuat Tabel Employees

```
CREATE TABLE employees(
employee_id int(5) not null,
first_name varchar(20),
last_name varchar(20),
email varchar(50),
phone_number decimal(15,0),
hire_date date,
job_id varchar(10) not null,
salary int(15),
commission_pct decimal(15,0),
manager_id varchar(10),
department_id int(5) not null,
constraint employee_pk primary key (employee_id)
);
```

Contoh tampilan

```
mysql> CREATE TABLE employees(
    -> employee_id int(5) not null,
    -> first_name varchar(20),
    -> last_name varchar(20),
    -> email varchar(50),
    -> phone_number decimal(15,0),
    -> hire_date date,
    -> job_id varchar(10) not null,
    -> salary int(15),
    -> commission_pct decimal(15,0),
    -> manager_id varchar(10),
    -> department_id int(5) not null,
    -> constraint employee_pk primary key (employee_id)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

2.7. Membuat Tabel Job_History

```
CREATE TABLE job_history(
employee_id int(5) not null,
start_date date,
end_date date not null,
job_id varchar(10) not null,
department_id int(5) not null,
constraint job_history_pk primary key (employee_id, start_date),
constraint job_history_fk1 foreign key (employee_id) references employees
(employee_id),
constraint job_history_fk2 foreign key (job_id) references jobs (job_id),
constraint job_history_fk3 foreign key (department_id) references departments
(department_id)
);
```

Contoh tampilan

```
mysql> CREATE TABLE job_history(
    -> employee_id int(5) not null,
    -> start_date date,
    -> end_date date not null,
    -> job_id varchar(10) not null,
    -> department_id int(5) not null,
    -> constraint job_history_pk primary key (employee_id, start_date),
    -> constraint job_history_fk1 foreign key (employee_id) references employees
(employee_id),
    -> constraint job_history_fk2 foreign key (job_id) references jobs (job_id),
    -> constraint job_history_fk3 foreign key (department_id) references departments
(department_id)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

3. Memasukkan Data Pada Tabel

3.1. Memasukkan Data Pada Tabel Regions

```
INSERT INTO regions value(1001,"Asia");
INSERT INTO regions value(1007,"Amerika");
INSERT INTO regions value(1003,"Eropa");
```

Contoh tampilan

```
mysql> INSERT INTO regions value(1001,"Asia");
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO regions value(1007,"Amerika");
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO regions value(1003,"Eropa");
Query OK, 1 row affected (0.01 sec)
```

3.2. Memasukkan Data Pada Tabel Countries

```
INSERT INTO countries value("C0001","Indonesia","1001");
INSERT INTO countries value("C0002","Amerika Serikat","1007");
INSERT INTO countries value("C0003","Inggris","1003");
```

Contoh tampilan

```
mysql> INSERT INTO countries value("C0001","Indonesia","1001");
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO countries value("C0002","Amerika Serikat","1007");
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO countries value("C0003","Inggris","1003");
Query OK, 1 row affected (0.01 sec)
```

3.3. Memasukkan Data Pada Tabel Locations

```
INSERT INTO locations VALUES(1000,"Ruko Premier Serenity j 40 Bekasi
Jaya",17112,"Bekasi","Jawa Barat","C0001");
INSERT INTO locations VALUES(1100,"Jabberwocky Rd",26192,"Southlake","Texas","C0002");
INSERT INTO locations VALUES(1200,"9702 Chester
Road",96298,"Stretford","Manchester","C0003");
```

Contoh tampilan

```
mysql> INSERT INTO locations VALUES(1000,"Ruko Premier Serenity j 40 Bekasi
Jaya",17112,"Bekasi","Jawa Barat","C0001");
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO locations VALUES(1100,"Jabberwocky
Rd",26192,"Southlake","Texas","C0002");
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO locations VALUES(1200,"9702 Chester
Road",96298,"Stretford","Manchester","C0003");
Query OK, 1 row affected (0.00 sec)
```

3.4. Memasukkan Data Pada Tabel Departments

```
INSERT INTO departments VALUES(60,'SALES',201,1000);
INSERT INTO departments VALUES(61,'ACCOUNTS',201,1100);
INSERT INTO departments VALUES(80,'FINANCE',211,1200);
```

Contoh tampilan

```
mysql> INSERT INTO departments VALUES(60,'SALES',201,1000);
Query OK, 1 row affected (0.02 sec)

mysql> INSERT INTO departments VALUES(61,'ACCOUNTS',201,1100);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO departments VALUES(80,'FINANCE',211,1200);
Query OK, 1 row affected (0.00 sec)
```

3.5. Memasukkan Data Pada Tabel Jobs

```
INSERT INTO jobs VALUES(101,"Manager",3500000,5000000);
INSERT INTO jobs VALUES(102,"Supervisor",2500000,4000000);
INSERT INTO jobs VALUES(103,"Technical Support",1750000,3500000);
```

Contoh tampilan

```
mysql> INSERT INTO jobs VALUES(101,"Manager",3500000,5000000);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO jobs VALUES(102,"Supervisor",2500000,4000000);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO jobs VALUES(103,"Technical Support",1750000,3500000);
Query OK, 1 row affected (0.00 sec)
```

3.6. Memasukkan Data Pada Tabel Employees

```
INSERT INTO employees
VALUES(001,"Ahmad","Imanudin","ahmad@imanudin.com",08990927097,"2011-10-
01",101,3500000,0,"",60);
INSERT INTO employees
VALUES(002,"Dhenandi","Putra","dhenandi@imanudin.com",08990927098,"2016-10-
10",102,2500000,0,101,61);
INSERT INTO employees
VALUES(003,"Raihan","Utomo","raihan@imanudin.com",08990927099,"2018-04-
01",103,2000000,0,103,80);
```

Contoh tampilan

```
mysql> INSERT INTO employees
VALUES(001,"Ahmad","Imanudin","ahmad@imanudin.com",08990927097,"2011-10-
01",101,3500000,0,"",60);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO employees
VALUES(002,"Dhenandi","Putra","dhenandi@imanudin.com",08990927098,"2016-10-
10",102,2500000,0,101,61);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO employees
VALUES(003,"Raihan","Utomo","raihan@imanudin.com",08990927099,"2018-04-
01",103,2000000,0,103,80);
Query OK, 1 row affected (0.01 sec)
```

3.7. Memasukkan Data Pada Tabel Job_History

```
INSERT INTO job_history VALUES(001,"2011-10-01","0000-00-00",101,60);
INSERT INTO job_history VALUES(002,"2016-10-10","0000-00-00","102",61);
INSERT INTO job_history VALUES(003,"2018-04-01","0000-00-00","103",80);
```

Contoh tampilan

```
mysql> INSERT INTO job_history VALUES(001,"2011-10-01","0000-00-00",101,60);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO job_history VALUES(002,"2016-10-10","0000-00-00","102",61);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO job_history VALUES(003,"2018-04-01","0000-00-00","103",80);
Query OK, 1 row affected (0.00 sec)
```

4. Latihan Query

4.1. Menampilkan data dari 2 tabel dengan menggunakan operator perbandingan (>,<,>=,< =,=)

- Operator <

```
SELECT first_name,last_name FROM employees e
INNER JOIN jobs j on e.job_id=j.job_id
where salary < 3000000;
```

Contoh tampilan

```
mysql> SELECT first_name,last_name FROM employees e
    -> INNER JOIN jobs j on e.job_id=j.job_id
    -> where salary < 3000000;
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Dhenandi   | Putra     |
| Raihan     | Utomo     |
+------------+-----------+
2 rows in set (0.00 sec)
```

- Operator >

```
SELECT first_name,last_name FROM employees e
INNER JOIN jobs j on e.job_id=j.job_id
where salary > 3000000;
```

Contoh tampilan

```
mysql> SELECT first_name,last_name FROM employees e
    -> INNER JOIN jobs j on e.job_id=j.job_id
    -> where salary > 3000000;
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Ahmad      | Imanudin  |
+------------+-----------+
1 row in set (0.00 sec)
```

- Operator >=

```
SELECT first_name,last_name FROM employees e
INNER JOIN jobs j on e.job_id=j.job_id
where salary >=2100000;
```

Contoh tampilan

```
mysql> SELECT first_name,last_name FROM employees e
    -> INNER JOIN jobs j on e.job_id=j.job_id
    -> where salary >=2100000;
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Ahmad      | Imanudin  |
| Dhenandi   | Putra     |
+------------+-----------+
2 rows in set (0.00 sec)
```

- Operator < =

```
SELECT first_name,last_name FROM employees e
INNER JOIN jobs j on e.job_id=j.job_id
where salary <=2100000;
```

Contoh tampilan

```
mysql> SELECT first_name,last_name FROM employees e
    -> INNER JOIN jobs j on e.job_id=j.job_id
    -> where salary <=2100000;
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Raihan     | Utomo     |
+------------+-----------+
1 row in set (0.00 sec)
```

- Operator =

```
SELECT first_name,last_name FROM employees e
INNER JOIN jobs j on e.job_id=j.job_id
where salary=2500000;
```

Contoh tampilan

```
mysql> SELECT first_name,last_name FROM employees e
    -> INNER JOIN jobs j on e.job_id=j.job_id
    -> where salary=2500000;
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Dhenandi   | Putra     |
+------------+-----------+
1 row in set (0.01 sec)
```

4.2. Menampilkan data dari 3 tabel dengan menggunakan operator AND, OR

- Operator AND

```
SELECT city FROM regions r
INNER JOIN countries c on r.region_id=c.region_id
INNER JOIN locations l on c.country_id=l.country_id
where location_id < 2000 and location_id > 1000;
```

Contoh tampilan

```
mysql> SELECT city FROM regions r
    -> INNER JOIN countries c on r.region_id=c.region_id
    -> INNER JOIN locations l on c.country_id=l.country_id
    -> where location_id < 2000 and location_id > 1000;
+-----------+
| city      |
+-----------+
| Southlake |
| Stretford |
+-----------+
2 rows in set (0.00 sec)
```

- Operator OR

```
SELECT city FROM regions r INNER JOIN countries c on r.region_id=c.region_id INNER
JOIN locations l on
c.country_id=l.country_id where location_id=1000 or location_id < 1200;
```

Contoh tampilan

```
mysql> SELECT city FROM regions r INNER JOIN countries c on r.region_id=c.region_id
INNER JOIN locations l on
    -> c.country_id=l.country_id where location_id=1000 or location_id < 1200;
+-----------+
| city      |
+-----------+
| Bekasi    |
| Southlake |
+-----------+
2 rows in set (0.01 sec)
```

4.3. Menampilkan data dari 4 tabel dengan menggunakan operator IN, NOT

- Operator IN

```
SELECT department_id,city,department_name FROM regions r
INNER JOIN countries c on r.region_id=c.region_id
INNER JOIN locations l on c.country_id=l.country_id
INNER JOIN departments d on l.location_id=d.location_id
where department_name IN("SALES");
```

Contoh tampilan

```
mysql> SELECT department_id,city,department_name FROM regions r
    -> INNER JOIN countries c on r.region_id=c.region_id
    -> INNER JOIN locations l on c.country_id=l.country_id
    -> INNER JOIN departments d on l.location_id=d.location_id
    -> where department_name IN("SALES");
+---------------+--------+-----------------+
| department_id | city   | department_name |
+---------------+--------+-----------------+
|            60 | Bekasi | SALES           |
+---------------+--------+-----------------+
1 row in set (0.01 sec)
```

- Operator NOT

```
SELECT department_id,city,department_name FROM regions r
INNER JOIN countries c on r.region_id=c.region_id
INNER JOIN locations l on c.country_id=l.country_id
INNER JOIN departments d on l.location_id=d.location_id
where department_name NOT IN("SALES","FINANCE");
```

Contoh tampilan

```
mysql> SELECT department_id,city,department_name FROM regions r
    -> INNER JOIN countries c on r.region_id=c.region_id
    -> INNER JOIN locations l on c.country_id=l.country_id
    -> INNER JOIN departments d on l.location_id=d.location_id
    -> where department_name NOT IN("SALES","FINANCE");
+---------------+-----------+-----------------+
| department_id | city      | department_name |
+---------------+-----------+-----------------+
|            61 | Southlake | ACCOUNTS        |
+---------------+-----------+-----------------+
1 row in set (0.00 sec)
```

4.4. Menampilkan data dari 5 tabel dengan menggunakan operator LIKE

```
SELECT city,department_name FROM regions r
INNER JOIN countries c on r.region_id=c.region_id
INNER JOIN locations l on c.country_id=l.country_id
INNER JOIN departments d on l.location_id=d.location_id
INNER JOIN employees e on d.department_id=e.department_id where department_name like
"%s";
```

Contoh tampilan

```
mysql> SELECT city,department_name FROM regions r
    -> INNER JOIN countries c on r.region_id=c.region_id
    -> INNER JOIN locations l on c.country_id=l.country_id
    -> INNER JOIN departments d on l.location_id=d.location_id
    -> INNER JOIN employees e on d.department_id=e.department_id where department_name
like "%s";
+-----------+-----------------+
| city      | department_name |
+-----------+-----------------+
| Bekasi    | SALES           |
| Southlake | ACCOUNTS        |
+-----------+-----------------+
2 rows in set (0.00 sec)
```

4.5. Menampilkan data dari 6 tabel dengan menggunakan operator AND, OR dan Like

- Operator AND

```
SELECT start_date,first_name,last_name,email,city,department_name FROM regions r INNER
JOIN countries c on r.region_id=c.region_id INNER JOIN locations l on
c.country_id=l.country_id INNER JOIN departments d on l.location_id=d.location_id
inner join employees e on d.department_id=e.department_id inner join job_history j on
e.employee_id=j.employee_id  where start_date > "2010-01-01" AND start_date < "2018-
01-01";
```

Contoh tampilan

```
mysql> SELECT start_date,first_name,last_name,email,city,department_name FROM regions
r INNER JOIN countries c on r.region_id=c.region_id INNER JOIN locations l on
c.country_id=l.country_id INNER JOIN departments d on l.location_id=d.location_id
inner join employees e on d.department_id=e.department_id inner join job_history j on
e.employee_id=j.employee_id  where start_date > "2010-01-01" AND start_date < "2018-
01-01";
+------------+------------+-----------+-----------------------+-----------
+-----------------+
| start_date | first_name | last_name | email                 | city      |
department_name |
+------------+------------+-----------+-----------------------+-----------
+-----------------+
| 2011-10-01 | Ahmad      | Imanudin  | ahmad@imanudin.com    | Bekasi    | SALES
|
| 2016-10-10 | Dhenandi   | Putra     | dhenandi@imanudin.com | Southlake | ACCOUNTS
|
+------------+------------+-----------+-----------------------+-----------
+-----------------+
2 rows in set (0.00 sec)
```

- Operator OR

```
SELECT start_date,first_name,last_name,email,city,department_name FROM regions r
INNER JOIN countries c on r.region_id=c.region_id
INNER JOIN locations l on c.country_id=l.country_id
INNER JOIN departments d on l.location_id=d.location_id
INNER JOIN employees e on d.department_id=e.department_id
INNER JOIN job_history j on e.employee_id=j.employee_id
where start_date > "2010-01-01" AND start_date < "2018-01-01"
or department_name not in("FINANCE");
```

Contoh tampilan

```
mysql> SELECT start_date,first_name,last_name,email,city,department_name FROM regions
r
    -> INNER JOIN countries c on r.region_id=c.region_id
    -> INNER JOIN locations l on c.country_id=l.country_id
    -> INNER JOIN departments d on l.location_id=d.location_id
    -> INNER JOIN employees e on d.department_id=e.department_id
    -> INNER JOIN job_history j on e.employee_id=j.employee_id
    -> where start_date > "2010-01-01" AND start_date < "2018-01-01"
    -> or department_name not in("FINANCE");
+------------+------------+-----------+-----------------------+-----------
+-----------------+
| start_date | first_name | last_name | email                 | city      |
department_name |
+------------+------------+-----------+-----------------------+-----------
+-----------------+
| 2011-10-01 | Ahmad      | Imanudin  | ahmad@imanudin.com    | Bekasi    | SALES
|
| 2016-10-10 | Dhenandi   | Putra     | dhenandi@imanudin.com | Southlake | ACCOUNTS
|
+------------+------------+-----------+-----------------------+-----------
+-----------------+
2 rows in set (0.00 sec)
```

- Operator Like

```
SELECT start_date,first_name,last_name,email,city,department_name FROM regions r
INNER JOIN countries c on r.region_id=c.region_id
INNER JOIN locations l on c.country_id=l.country_id
INNER JOIN departments d on l.location_id=d.location_id
INNER JOIN employees e on d.department_id=e.department_id INNER JOIN job_history j on
e.employee_id=j.employee_id
where city like "b%";
```

Contoh tampilan

```
mysql> SELECT start_date,first_name,last_name,email,city,department_name FROM regions
r
    -> INNER JOIN countries c on r.region_id=c.region_id
    -> INNER JOIN locations l on c.country_id=l.country_id
    -> INNER JOIN departments d on l.location_id=d.location_id
    -> INNER JOIN employees e on d.department_id=e.department_id INNER JOIN
job_history j on e.employee_id=j.employee_id
    -> where city like "b%";
+------------+------------+-----------+-------------------+--------+-----------------
+
| start_date | first_name | last_name | email             | city   | department_name
|
+------------+------------+-----------+-------------------+--------+-----------------
+
| 2011-10-01 | Ahmad      | Imanudin  | ahmad@imanudin.com | Bekasi | SALES
|
+------------+------------+-----------+-------------------+--------+-----------------
+
1 row in set (0.00 sec)
```