**IF2240 Basis Data**

**Semester II 2021/2022**

Praktikum 2: Advanced SQL

Dipersiapkan oleh:

Asisten Laboratorium Basis Data

Pelaksanaan:

Waktu Mulai: Jumat, 11 Februari 2022 – 16.05 WIB

Waktu Selesai: Jumat, 11 Februari 2022 – 18.05 WIB

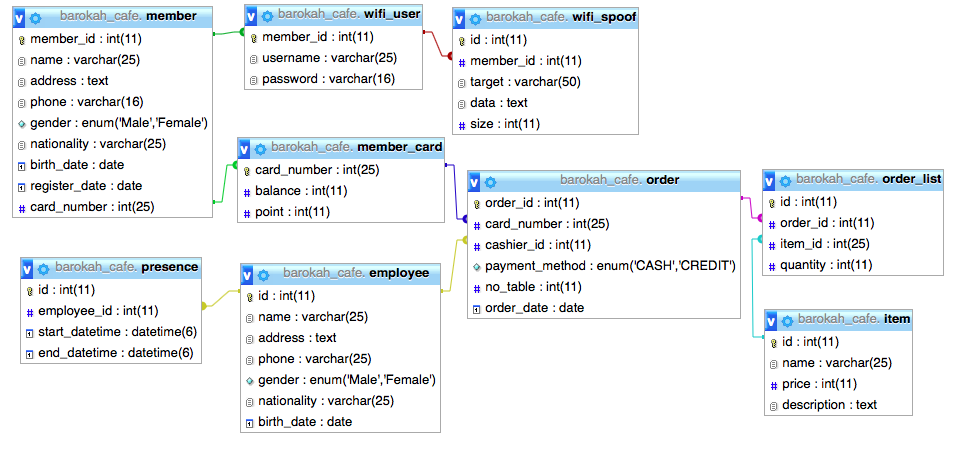
# I. Permasalahan

Buatlah basis data dengan nama “barokah\_cafe”. Gunakan basis data tersebut, kemudian masukkan (import) script SQL barokah\_cafe.sql yang telah diberikan bersama deskripsi tugas ini.

# 

# II. Skema Basis Data

Berikut skema relasional basis data berdasarkan deskripsi sistem yang telah dijabarkan pada bagian sebelumnya.



|  |
| --- |
| member = (member\_id, name, address, phone, gender, nationality, birth\_date, register\_date, card\_number)  wifi\_user = (member\_id, username, password)  wifi\_spoof = (id, member\_id, target, data, size)  member\_card = (card\_number, balance, point)  order = (order\_id, card\_number, cashier\_id, payment\_method, no\_table, order\_date)  order\_list = (id, order\_id, item\_id, quantity)  item = (id, name, price, description)  employee = (id, name, address, phone, gender, nationality, birth\_date)  presence = (id, employee\_id, start\_datetime, end\_datetime) |

# 

# III. Soal

Beberapa hal yang perlu diperhatikan dalam pengerjaan soal:

1. Untuk semua soal **tidak diperbolehkan untuk membuat view/tabel** **baru** terlebih dahulu **kecuali jika diperintahkan pada soal**
2. Untuk **penggabungan dua string**, silakan gunakan fungsi **CONCAT** pada MariaDB
3. Untuk query yang menggunakan **tabel order** perlu menambahkan tanda backtick, contoh: **`order`**

## Buatlah query untuk menampilkan seluruh informasi member dan total pointnya dari member yang mengandung nama "maria" diurutkan dari member terbaru.

**Jawaban query :**

|  |
| --- |
| SELECT m.name, mc.point FROM member m NATURAL JOIN member\_card mc WHERE m.name LIKE "%Maria%" ORDER BY m.register\_date desc; |

**Hasil screenshot :**

|  |
| --- |
|  |

## Buatlah query untuk menampilkan daftar makanan beserta jumlah masing-masing makanan yang dipesan oleh meja nomor 8 pada tanggal 7 April 2016.

**Jawaban query :**

|  |
| --- |
| SELECT i.name, ol.quantity FROM order\_list ol JOIN item i ON ol.item\_id = i.id WHERE ol.order\_id IN (SELECT order\_id FROM `order` WHERE no\_table=8 AND order\_date="2016-04-07"); |

**Hasil screenshot :**

|  |
| --- |
|  |

## Buatlah sebuah query untuk menampilkan ID dan nama member yang paling sering *numpang* wifi tanpa membeli makanan/minuman di Barokah Cafe.

**Jawaban query :**

|  |
| --- |
| SELECT m.member\_id, m.name FROM member m WHERE m.member\_id IN (SELECT member\_id FROM wifi\_spoof) AND m.card\_number NOT IN (SELECT card\_number FROM `order`); |

**Hasil screenshot :**

|  |
| --- |
|  |

## Buatlah sebuah query untuk membuat view dengan nama "vanilla" untuk pesanan yang terkait dengan item "Vanilla Milkshake" atau "Vanilla Latte", dan pesanan tersebut dilayani oleh kasir yang memiliki jenis kelamin "Female". Dalam view tersebut, data yang disertakan adalah ID orderlist sebagai view\_id, nama member, nomor kartu member, ID order, ID item, nama item, ID kasir, dan nama kasir. Setelah itu, tampilkan data yang ditampung dalam view terurut dari view\_id secara menurun.

## 

**Jawaban query :**

|  |
| --- |
| ALTER TABLE member RENAME column name to member\_name;  ALTER TABLE employee RENAME column name to employee\_name;  ALTER TABLE item RENAME column name to item\_name;  CREATE VIEW vanilla AS (SELECT ol.id AS "view\_id", m.member\_name, o.card\_number, o.order\_id, i.item\_name, o.cashier\_id, e.employee\_name FROM order\_list ol NATURAL JOIN `order` o JOIN item i ON ol.item\_id=i.id, member m NATURAL JOIN member\_card mc, employee e WHERE o.cashier\_id = e.id AND o.card\_number=mc.card\_number AND i.item\_name LIKE "Vanilla %" AND e.gender LIKE "Female");  SELECT \* FROM vanilla ORDER BY view\_id desc; |

**Hasil screenshot :**

|  |
| --- |
|  |

## Buatlah sebuah query untuk membuat view dengan nama "brazilian" untuk member yang berasal dari negara "Brazil" dan bergabung menjadi member pada tahun 2016, serta memiliki point yang lebih besar dari 50. Data yang akan disertakan adalah ID member, nama member sebagai "nome", gender, nomor kartu, dan balance kartu sebagai "saldo". Kemudian, tampilkan data yang ditampung dalam view tersebut.

**Jawaban query :**

|  |
| --- |
| REATE VIEW brazillian AS (SELECT m.member\_id, m.member\_name AS "nome", m.gender, m.card\_number, mc.balance AS "saldo" FROM member m NATURAL JOIN member\_card mc WHERE m.nationality LIKE "Brazil" AND mc.point > 50);  SELECT \* FROM brazillian; |

**Hasil screenshot :**

|  |
| --- |
|  |

## Buatlah query untuk menambahkan sebuah item dengan nama “Matcha Latte” dan harga 23000. Isi kolom deskripsi dibebaskan. ID mengikuti auto increment.

**Jawaban query :**

|  |
| --- |
| INSERT INTO item VALUE(id,"Matcha Latte", 23000,"-"); |

**Hasil screenshots :**

\*screenshot sebelum (tabel awal), saat (query), dan setelah (tabel akhir) melakukan query

|  |
| --- |
| **Sebelum**    **Saat**    **Typo 23000 menjadi 230000, namun sudah dibenarkan untuk soal selanjutnya**  **SetelahD** |

## Buatlah query untuk menambahkan order baru dari member dengan nomor kartu 813187520 yang dilayani oleh cashier dengan id 22, menggunakan metode pembayaran secara cash, dengan no meja 11 dan tanggal order hari ini. ID mengikuti auto increment. Kemudian **tambahkan juga order list baru** dari order tersebut dengan id item 8 dan kuantitas 2.

**Jawaban query :**

|  |
| --- |
| INSERT INTO `order` value(order\_id, 813187520, 22, 'CASH', 11, CURRENT\_DATE);  INSERT INTO order\_list value(id,(SELECT order\_id FROM `order` WHERE card\_number = 813187520 AND cashier\_id = 22 AND no\_table=11 AND order\_date = CURRENT\_DATE), 8, 2); |

**Hasil screenshots :**

\*screenshot sebelum (tabel awal), saat (query), dan setelah (tabel akhir) melakukan query

|  |
| --- |
| **Sebelum**          **Saat**      **Sesudah**    **catatan: id=102 karena 1x melakukan rollback (undo)** |

## Di masa pandemi kali ini, pemilik barokah cafe mempunyai ide untuk menaikkan harga sebagian item yang dimilikinya sehingga mendapatkan keuntungan yang lebih besar. Adapun ketentuan kenaikan harga item barokah cafe sebagai berikut:

## Untuk harga diatas 50000, harga item dinaikkan sebesar 10%

## Untuk harga dibawah 30000, harga item dinaikkan sebesar 20%

## Item sisanya akan memiliki harga tetap

## 

## Buktikan perubahan dengan menampilkan nama item dan harga yang baru.

## Constraint: hanya boleh menggunakan satu query update.

**Jawaban query :**

|  |
| --- |
| UPDATE item SET price = IF(price>50000, price\*1.1, IF(price<30000,price\*1.2,price)); |

**Hasil screenshot :**

|  |
| --- |
| **SEBELUM**    **SAAT**    **SETELAH** |

## Untuk merayakan *3.3* *flash sale* barokah cafe ingin memberikan uang untuk negara philippines.

## Buatlah query untuk melakukan update balance menjadi 100.000.000 pada tabel member card yang merupakan member yang berasal dari negara philippines. Lalu buktikan perubahan tersebut dengan tampilkan nama member, nationality, balance, dan card\_number member yang terkena update.

**Jawaban query :**

|  |
| --- |
| UPDATE member\_card mc SET mc.balance = 1000000000 WHERE mc.card\_number IN (SELECT card\_number FROM member WHERE nationality LIKE "Philippines"); |

**Hasil screenshot :**

|  |
| --- |
| **Sebelum**    **Saat**    **Setelah** |

## Hapus seluruh order\_list yang memiliki item dengan harga lebih dari 30.000. Buktikan dengan menampilkan seluruh atribut dari order\_list dan diurutkan secara menaik berdasarkan id item.

**Jawaban query :**

|  |
| --- |
| DELETE FROM order\_list WHERE item\_id IN (SELECT id FROM item WHERE price > 30000); |

**Hasil screenshot :**

|  |
| --- |
| **Sebelum**      **Saat**    **Sesudah** |

## Hapus seluruh row pada wifi\_spoof yang memiliki member dengan point pada member card kurang dari 10. Buktikan dengan menampilkan seluruh atribut pada wifi\_spoof.

**Jawaban query :**

|  |
| --- |
| DELETE FROM wifi\_spoof WHERE member\_id IN (SELECT m.member\_id FROM member m NATURAL JOIN member\_card mc WHERE mc.point < 10); |

**Hasil screenshot :**

|  |
| --- |
| **Sebelum**    **Saat**    **Sesudah** |

## Buatlah tabel baru bernama employee\_salary yang berisi employee\_id yang mengacu kepada employee(id), salary bertipe INTEGER, dan start\_date bertipe DATE. Ketiga atribut tidak boleh bernilai null. Apabila suatu record A pada tabel employee dihapus/diupdate, maka record B pada employee\_salary yang berisi foreign key record A juga ikut terhapus/terupdate. Buktikan menggunakan describe.

**Jawaban query :**

|  |
| --- |
| CREATE TABLE IF NOT EXISTS employee\_salary(employee\_id INT(11) UNIQUE NOT NULL, salary INT NOT NULL, start\_date DATE NOT NULL, FOREIGN KEY(employee\_id) REFERENCES employee(id) ON DELETE CASCADE); |

**Hasil screenshot :**

|  |
| --- |
|  |