**LAPORAN**

**PRAKTEK BASIS DATA**

**INNER JOIN, OUTER JOIN DAN SELF JOIN**

****

**DISUSUN OLEH :**

**SULIS TIYAH**

**2001081002**

**TEKNIK KOMPUTER - 2B**

**SEMESTER 3**

**JURUSAN TEKNOLOGI INFORMASI**

**PROGRAM STUDI TEKNIK KOMPUTER**

**POLITEKNIK NEGERI PADANG**

**2021**

1. **DASAR TEORI**

INNER JOIN

Inner join merupakan jenis join yang paling umum yang dapat digunakan pada semua database. Jenis ini dapat digunakan bila ingin merelasikan dua set data yang ada di tabel, letak relasinya setelah pada perintah ON pada join. • Inner join mengembalikan baris-baris dari dua tabel atau lebih yang memenuhi syarat.

OUTER JOIN

Outer join merupakan jenis join yang sedikit berbeda dengan inner join. Pada MySQL, bentuk perintah untuk menerapkan outer join ada 2 yaitu : 1. Left [Outer] Join 2. Right [Outer]Join 3. Full [Outer] Join

Left [Outer] Join • Left outer join (sering disingkat left join) akan mengembalikan seluruh baris dari tabel disebelah kiri yang dikenai kondisi ON dan hanya baris dari tabel disebelah kanan yang memenuhi kondisi join. • Left join Adalah Relasi Antar Table, biasanya Digunakan untuk menghasilkan baris data dari tabel kiri (nama tabel pertama/ Tabel Utama) yang tidak ada pasangan/Tidak Berelasi datanya pada tabel kanan (nama tabel kedua).

Left [Outer] Join without Intersection • Join ini merupakan variasi dari left outer join. Pada join ini kita hanya akan mengambil data dari tabel sebelah kiri yang dikenai kondisi ON yang juga memenuhi kondisi join tanpa data dari tabel sebelah kanan yang memenuhi kondisi join

Right outer join (sering disingkat right join) akan mengembalikan semua baris dari tabel sebelah kanan yang dikenai kondisi ON dengan data dari tabel sebelah kiri yang memenuhi kondisi join. Teknik ini merupakan kebalikan dari left outer join. • RIGHT JOIN digunakan untuk menghasilkan baris data dari tabel kanan (nama tabel kedua/ Tabel Utama) yang tidak ada pasangan datanya/ Tidak Berelasi pada tabel kiri (nama tabel pertama).

Right[Outer] Join without Intersection • Teknik ini merupakan variasi dari right outer join. Pada join ini kita hanya akan mengambil data dari tabel sebelah kanan yang dikenai kondisi ON yang juga memenuhi kondisi join tanpa data dari tabel sebelah kanan yang memenuhi kondisi join

Full[Outer] Join without Intersection • Variasi lain dari full outer join yang akan mengembalikan seluruh data dari kedua tabel yang dikenai ON tanpa data yang memiliki nilai NULL.

**OUTER JOIN** merupakan tipe join yang mencari referensi data dari suatu table sumber ke table lain dengan tidak menghilangkan data sumber apabila referensi tidak diketemukan.

Untuk menggunakan tipe OUTER JOIN maka perlu memperhatikan beberapa hal berikut :

* perlu dibedakan antara table sumber dan table referensi, ini ditentukan dengan cara menspesifikasikan kedudukan table sumber apakah di kiri (**LEFT**) atau di kanan (**RIGHT**).
* jika tidak ada data dari table referensi yang cocok dengan kondisi join maka hanya data dari table sumber yang ditampilkan tetapi kolom-kolom table referensi akan berisi null.

SELF JOIN

Self join berguna untuk menggabungkan tabel dengan dirinya sendiri. Kami bergabung dengan tabel itu sendiri untuk mengevaluasi baris dengan baris lain di tabel yang sama. Untuk melakukan self-join, kita menggunakan klausa inner join atau left join .

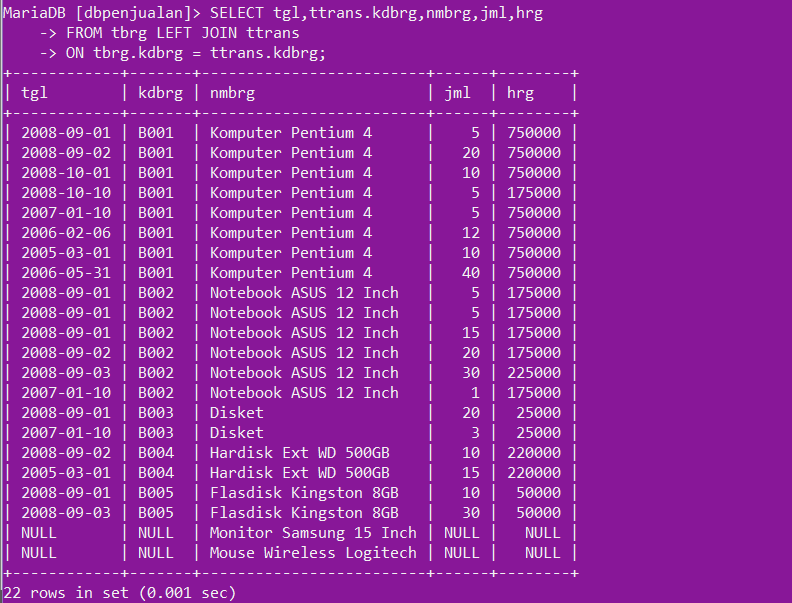
Perintah Join dimana suatu tabel berelasi dengan dirinya sendiri SELECT column\_name(s) FROM table1 T1, table1 T2 WHERE condition;

1. **ALAT DAN BAHAN**
2. PC/Laptop
3. Software XAMPP Control Panel
4. **LANGKAH KERJA**
5. Jalankan XAMPP

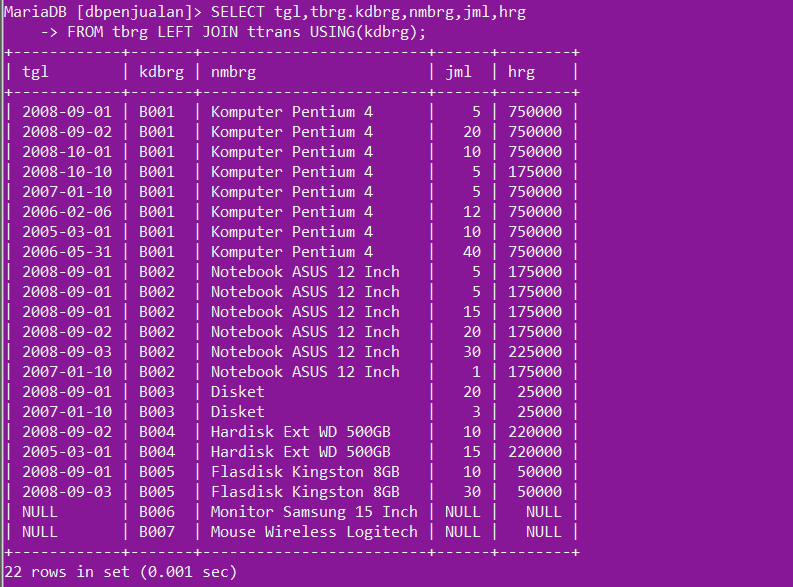
**OUTER JOIN – LEFT OUTER JOIN**

1. Tampilkan data tgl transaksi, namabarang, jumlah, harga untuk seluruh barang baik yang sudah terjual maupun tidak.

* **JOIN ON**



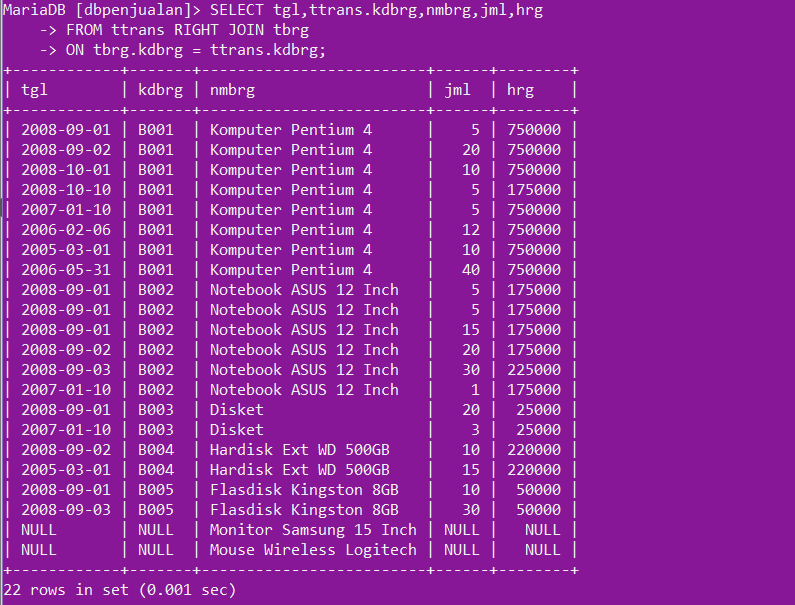
* **USING**



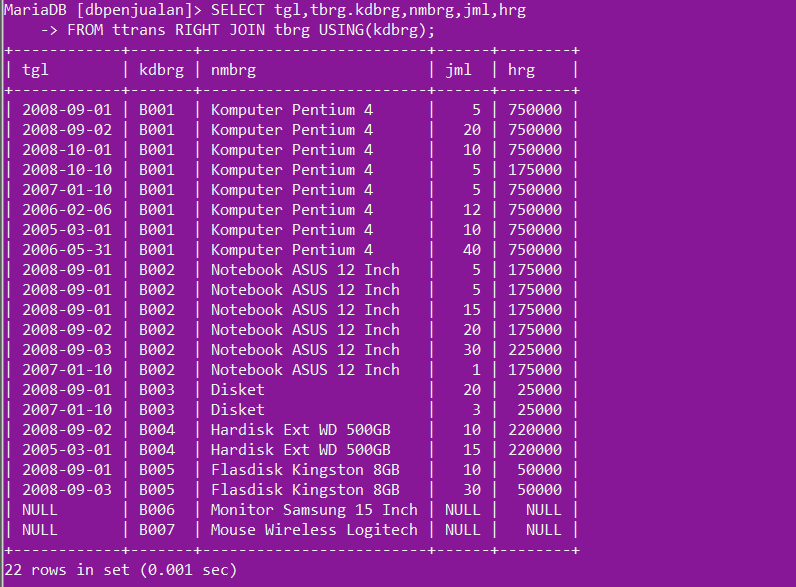
**OUTER JOIN – RIGHT OUTER JOIN**

1. Tampilkan data tgl transaksi, namabarang, jumlah, harga untuk seluruh barang baik yang sudah terjual maupun tidak.

* **JOIN ON**



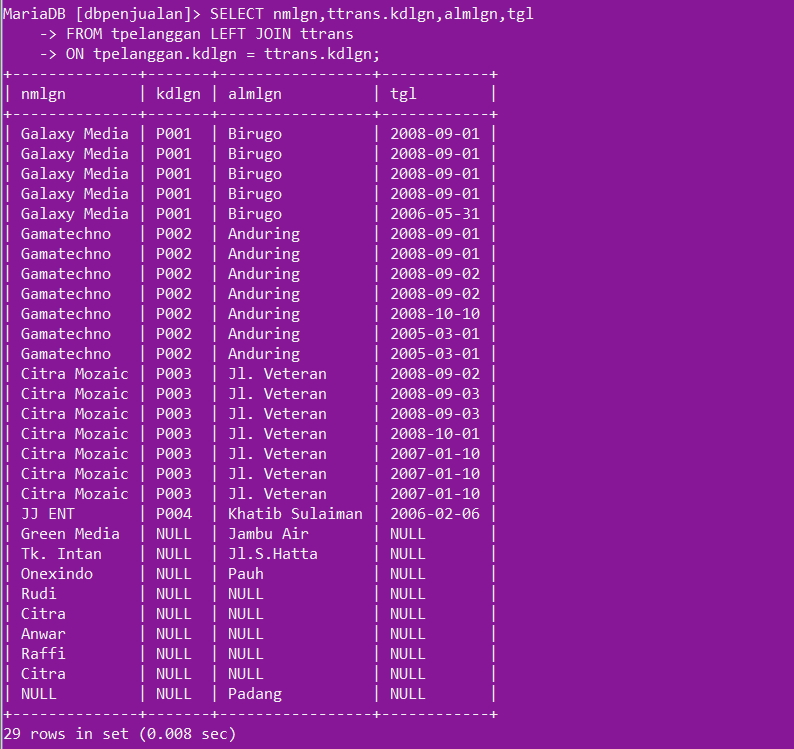
* **USING**



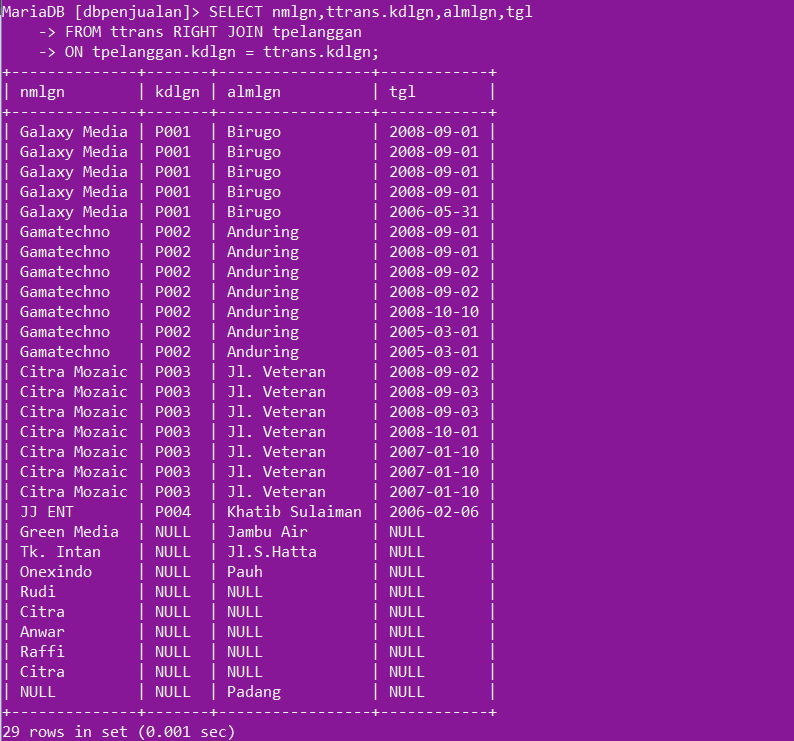
**OUTER JOIN (LEFT, RIGHT) – DBPENJUALAN**

1. Tampilkan seluruh namapelanggan beserta alamat dan tanggal transaksi untuk pelanggan yang pernah maupun belum bertransaksi

* **LEFT JOIN**



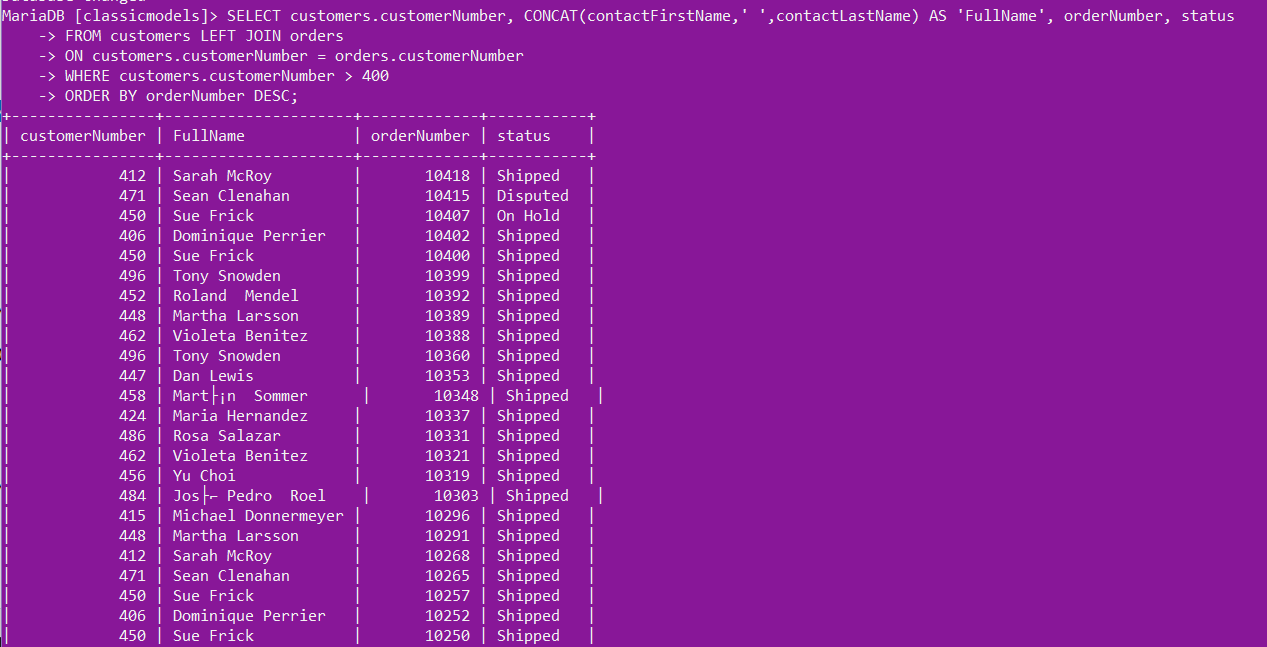
* **RIGHT JOIN**



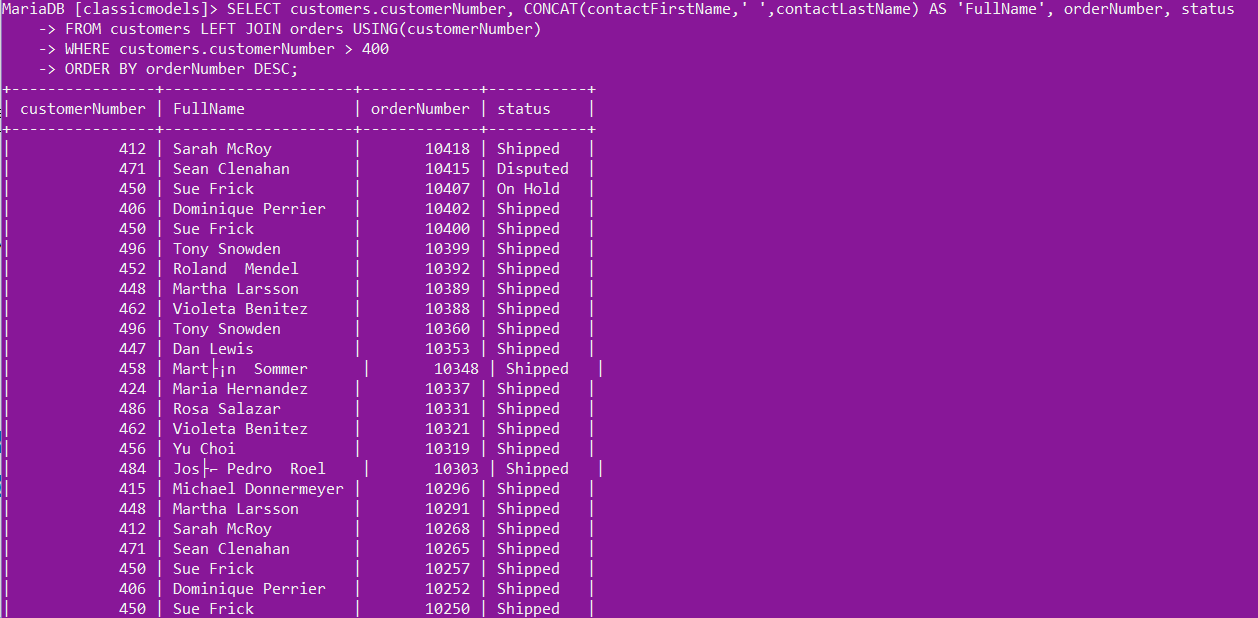
**OUTER JOIN (LEFT, RIGHT) – CLASSICMODELS**

1. Tampilkan customerNumber,namalengkap customer(gabungan contactfirstname, contactlastname), ordernumber dan status untuk semua customer baik yang pernah order ataupun tidak. uutkan dari ordernumber desc.

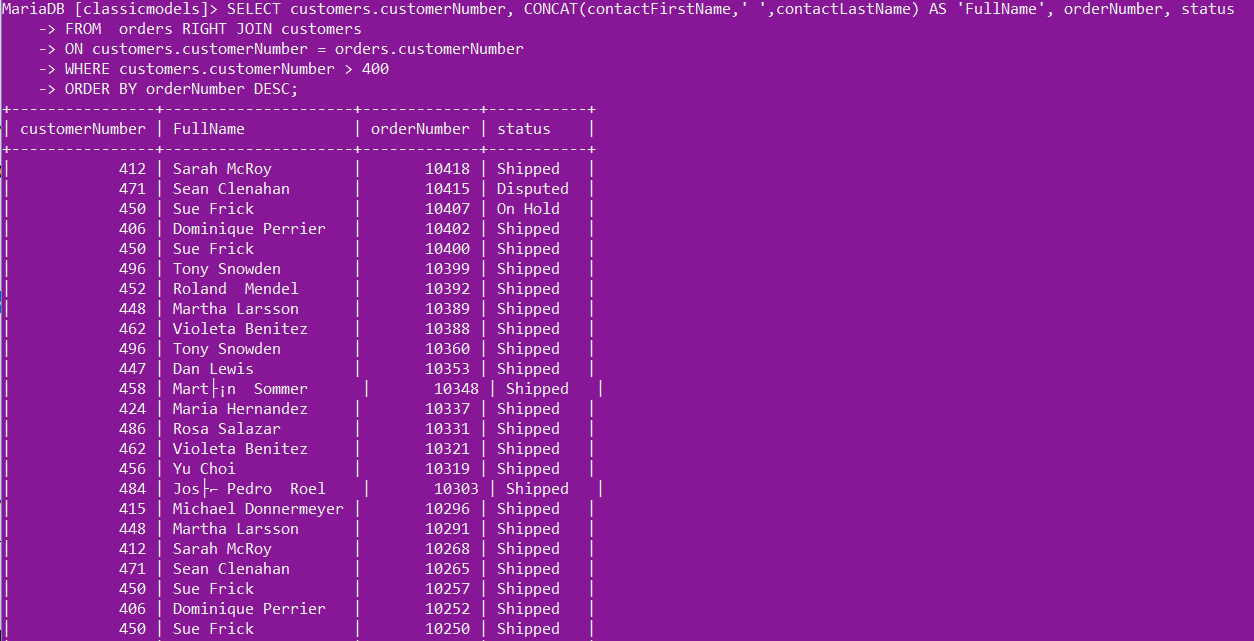
* **LEFT JOIN ON**



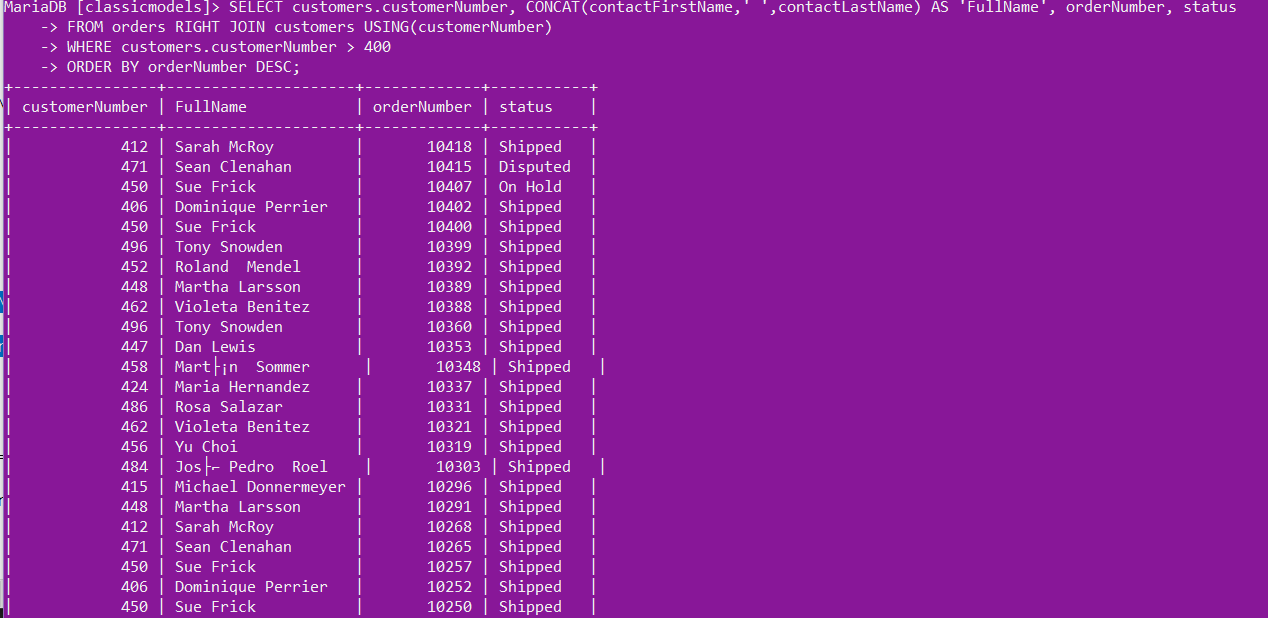
* **LEFT JOIN USING**



* **RIGHT JOIN ON**



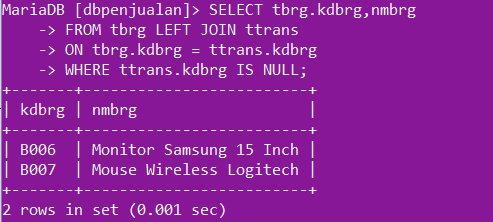
* **RIGHT JOIN USING**



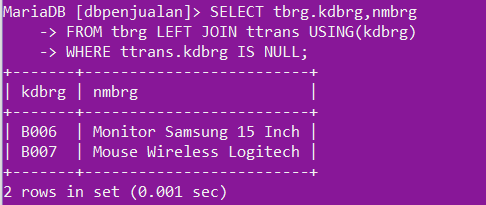
**OUTER JOIN – LEFT/RIGHT JOIN WITHOUT INTERSECTION**

1. Tampilkan data barang (kode barang , nama barang, nonota) yang tidak laku

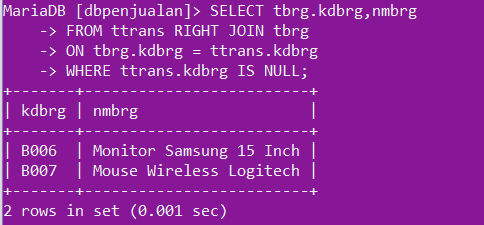
* **LEFT JOIN ON**



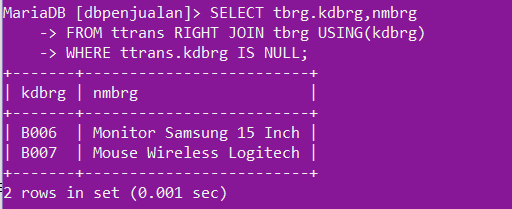
* **LEFT USING**



* **RIGHT JOIN ON**

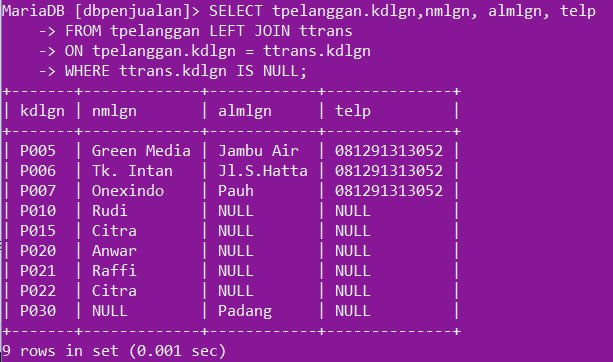


* **RIGHT USING**

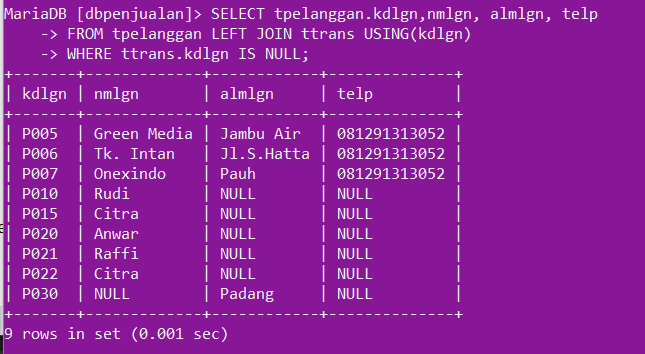


1. Tampilkan data pelanggan yang tidak pernah melakukan pembelian (nama, alamat, dan no tlp)

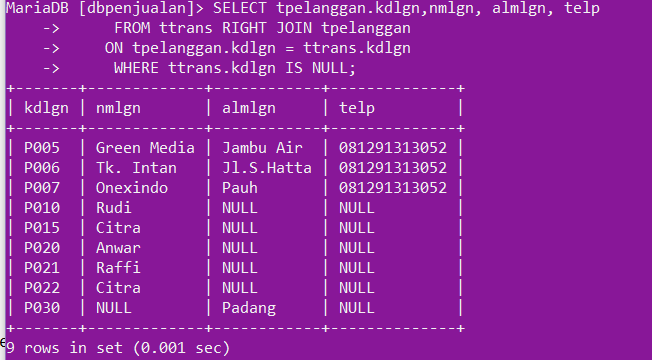
* **LEFT JOIN ON**



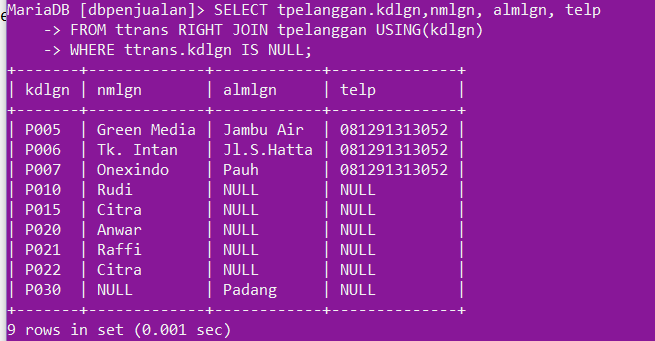
* **LEFT USING**



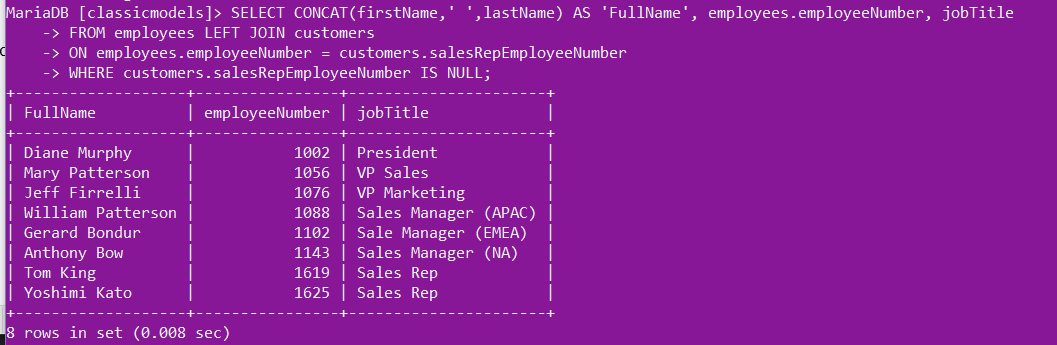
* **RIGHT JOIN ON**

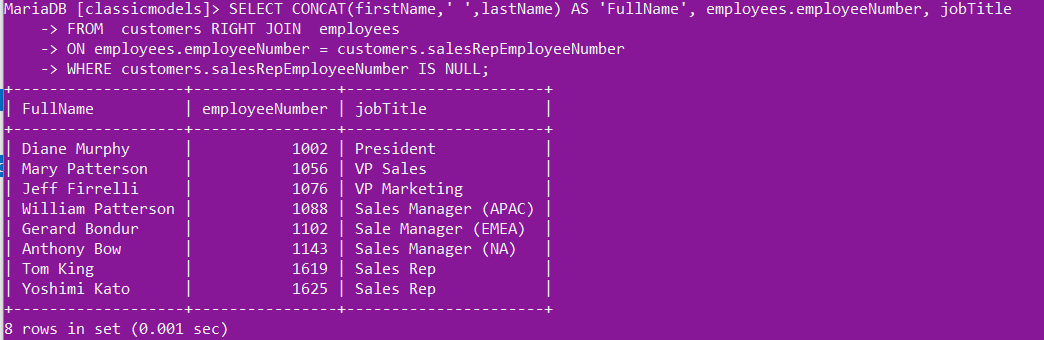


* **RIGHT USING**



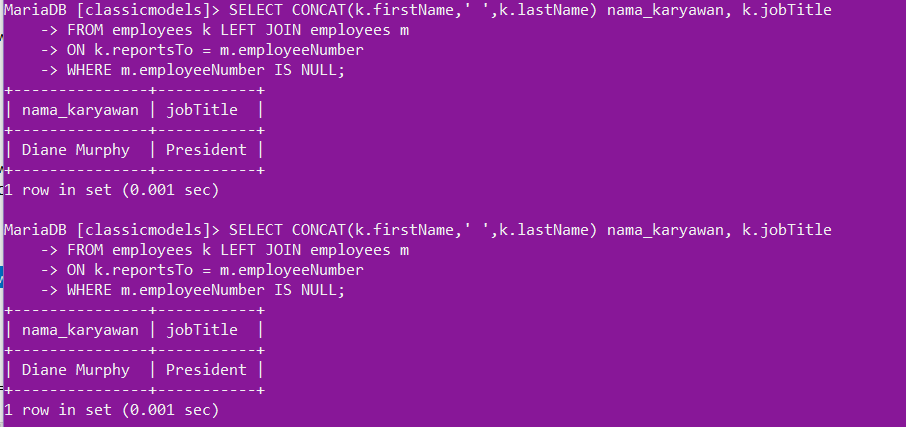
1. Tampilkan nama lengkap employee beserta employeenumber, jobtitle yang tidak pernah melayani transaksi penjualan customers



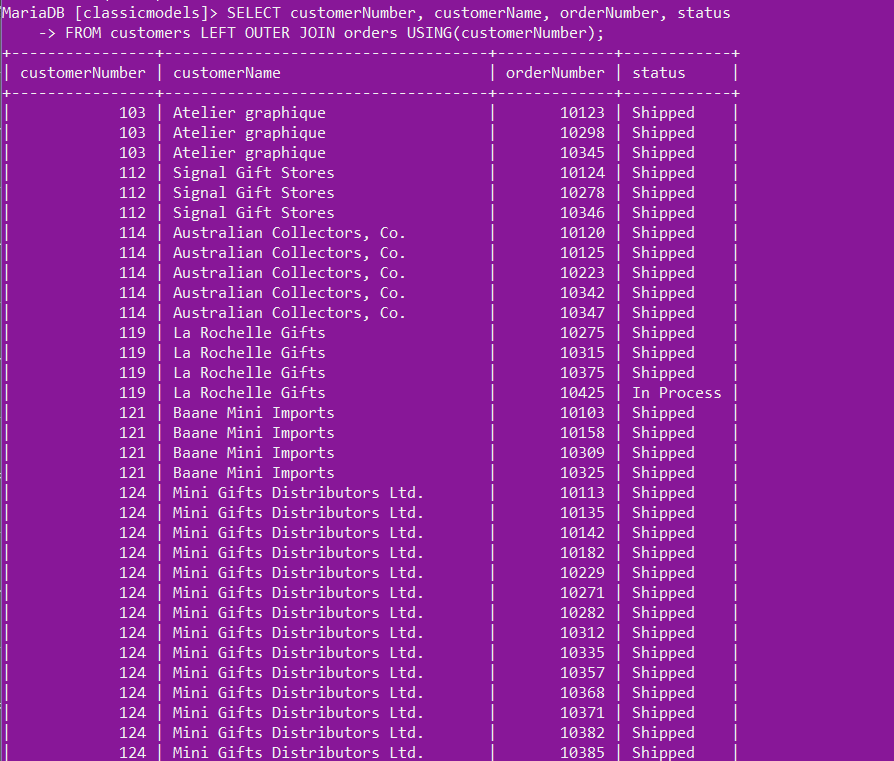


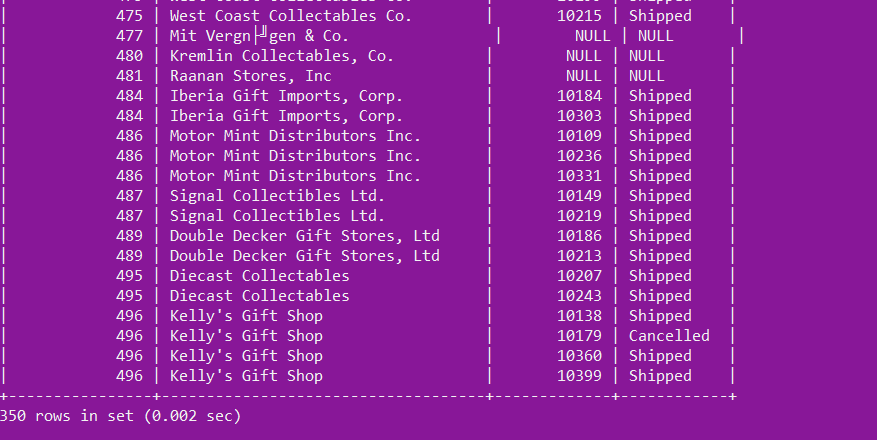
**SELF JOIN**

1. Tampilkan nama karyawan dan jontitle yang tidak memiliki manager(tidak memberikan laporan kesiapa-siapa)

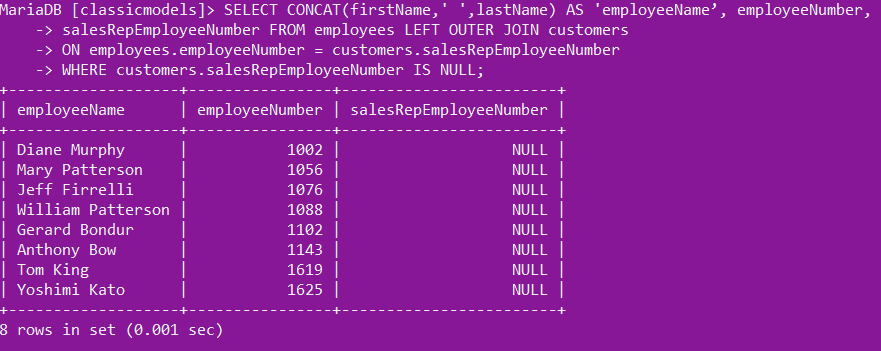


1. **SOAL DAN PEMBAHASAN**
2. Tampilkan customer number, customer name, order number, status untuk semua customer baik yang pernah melakukan transaksi maupun yang belum pernah.

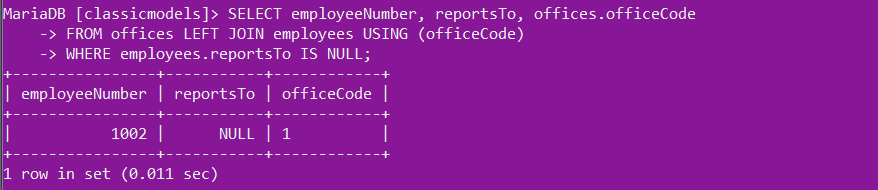




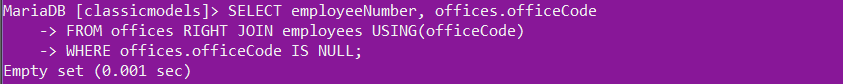
1. Tampilkan nama lengkap employee, employeesnumber, salesrepemployeenumber untuk employee yang tidak pernah melakukan proses sales terhadap customer.



1. Adakah offices yang tidak memiliki employee? Ada atau tidaknya buktikan dengan query.



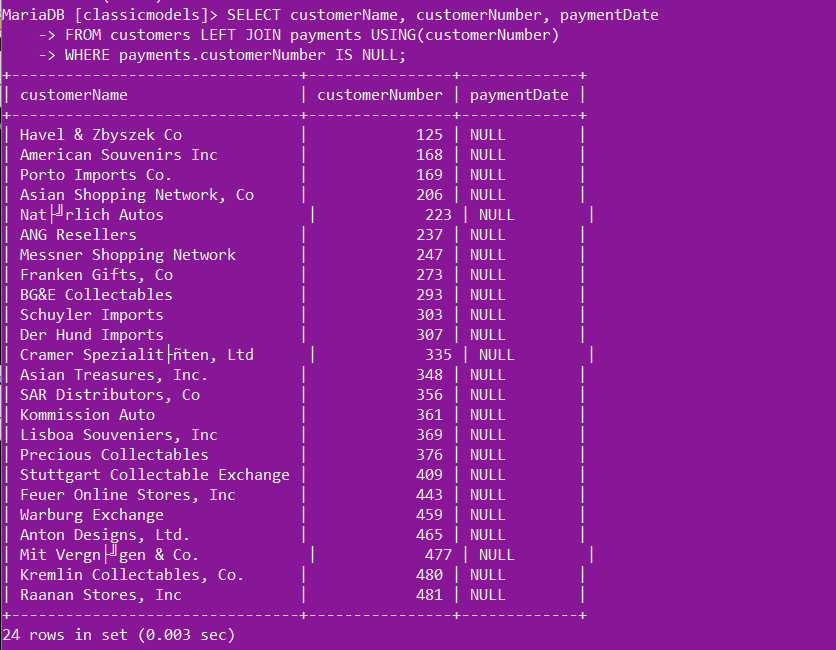
4. Adakah employee yang tidak bekerja di office manapun? Ada atau tidaknya buktikan dengan query.



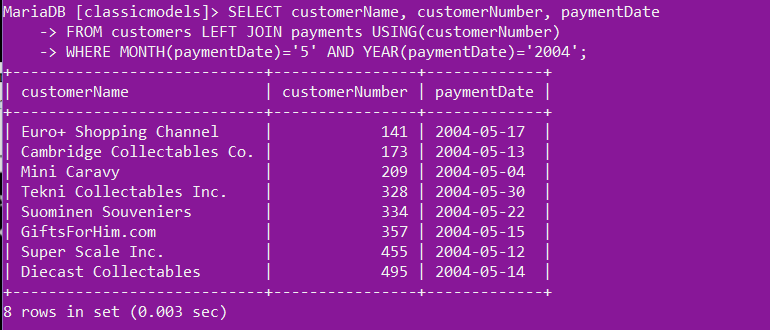
5. Tampilkan nama lengkap karyawan, job title karyawan dan nama manajer serta kode dan kota offices untuk karyawan dengan job title : Sales Rep yang kantornya di Kota Paris, Sidney dan London



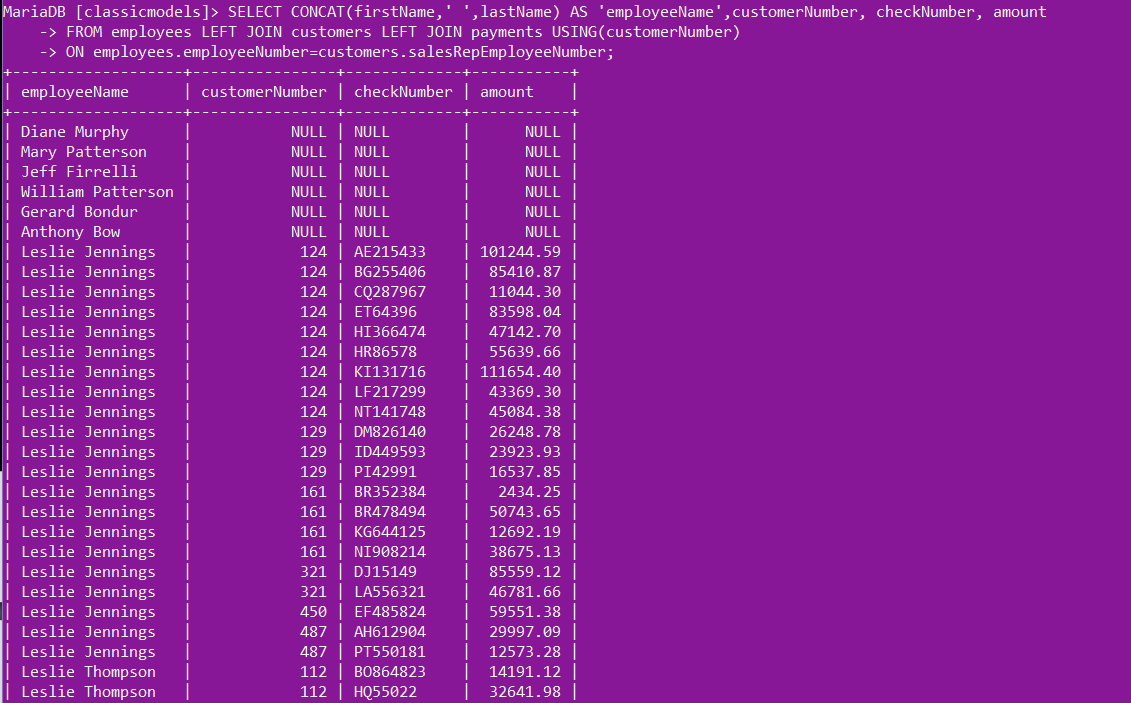
6. Tampilkan nama customer, customer number, paymentdate untuk customer yang belum melakukan payments.

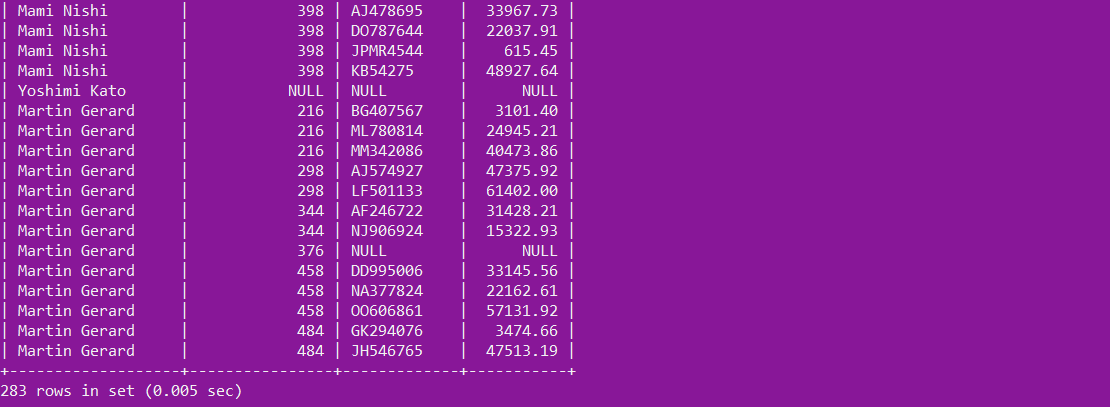


7. Tampilkan nama customer, customernumber dan paymentdate untuk customer yang sudah pernah melakukan order pada bulan mei 2004.

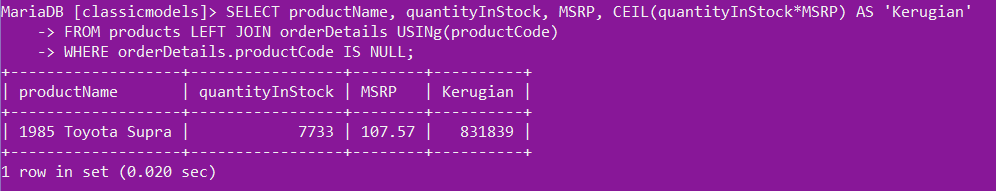


8. Tampilkan nama lengkap employee, customernumber,checknumber dan amount untuk semua employee yang telah/belum melakukan proses sales terhadap customer dan untuk semua customer baik yang sudah melakukan payments ataupun belum.





9. Tampilkan productname,quantityinstock, MSRP dan perkiraan kerugian(quantityinstock \* MSRP) untuk product yang belum pernah dibeli. Lakukan pembulatan keatas untuk perkiraan kerugian.



1. **REFERENSI**

<https://www.google.com/search?q=outer+join+mysql&oq=outer+join+&aqs=chrome.2.69i57j0i512l4j69i61j69i60j69i61.10340j0j7&sourceid=chrome&ie=UTF-8>

<http://mysql.phi-integration.com/sql/tipe-tipe-join-pada-mysql-5#:~:text=OUTER%20JOIN%20merupakan%20tipe%20join,sumber%20apabila%20referensi%20tidak%20diketemukan>

<http://fatihfiles.blogspot.com/2010/10/join-dalam-mysql.html>

<https://dosenit.com/sql/self-join-sql-syntax-dan-contoh-penggunaannya#:~:text=Self%20JOIN%20adalah%20gabungan%20reguler%2C%20tetapi%20tabel%20bergabung%20dengan%20dirinya%20sendiri>