**PERBAIKAN UTS SISTEM MANAJEMEN BASIS DATA**

**Nama : Istiqomah**

**NIM : L200150076**



Gambar-1. Rancangan Tabel dalam database Classicmodels

1. Lihat gambar-1 di halaman-2, gambar tersebut menampilkan semua tabel yang terdapat dalam database ‘classicmodels’. (a) Buatlah semua relasi antar tabel, dan terangkan jenis relasinya (20%). (b) Jelaskan fungsi masing-masing tabel (digunakan untuk menyimpan data apa?) (10%).

Jawaban:

* 1. Relasi dan Kardinalitas:
     + Employees “menempati” Offices

Kardinalitas: many-to-one (n:1)

Alasan: Karena 1 offices bisa ditempati oleh banyak employees.

Primary Key table employees menjadi Foreign Key di dalam table Offices

* + - Employees “melayani” customers

Kardinalitas: one-to-many (1:n)

Alasan: Karena 1 employees dapat melayani banyak customers.

Primary Key table Employees menjadi Foreign Key di dalam table customers.

* + - Customers “melakukan” orders

Kardinalitas: one-to-many (1:n)

Alasan: Karena 1 customers dapat melakukan banyak orders.

Primary Key table customers menjadi Foreign Key di dalam table orders.

* + - Customers “melakukan” payment

Kardinalitas: one-to-many (1:n)

Alasan: Karena 1 customers dapat melakukan payment(pembayaran) dari banyak transaksi (order).

Primary Key table order menjadi Foreign Key table payment.

* + - Employees “melaporkan” ke Employees (pimpinan)

Kardinalitas: many-to-one (n:1)

Alasan: Karena banyak employees akan melaporkan suatu hasil kepada seorang pimpinan.

Terdapat Foreign Key “reportsTo” di dalam table employees.

* + - Orders “memiliki” orderdetails

Kardinalitas: one-to-one (1:1)

Alasan: Karena satu order hanya akan memiliki satu detail order.

Primary Key table orders menjadi Foreign Key table orderdetails.

* + - Orderdetails “memiliki” Products

Kardinalitas: one-to-many (1:n)

Alasan: Karena data dalam orderdetails terdapat data product yang dipesan oleh customers.

Primary Key table Products menjadi Foreign Key table Orderdetails

* + - Products “terklasifikasi” dalam productLines

Kardinalitas: many-to-one (n:1)

Alasan: Karena beberapa product yang dijual dapat diklasifikasikan kedalam satu kategori yang sama.

Primary Key table products menjadi Foreign Key table productLines.

* 1. Fungsi tabel :
     + Customers : untuk menyimpan data pelanggan
     + Products : untuk menyimpan daftar produk/barang
     + Productlines : untuk menyimpan daftar kategori product secara detail
     + Orders : untuk menyimpan data pesanan yang diorder oleh pelanggan
     + Orderdetails : untuk menyimpan detail order
     + Payments : untuk menyimpan data pembayaran(transaksi)
     + Employees : untuk menyimpan informasi tentang pegawai
     + Offices : untuk menyimpan data pegawai kantor

1. Buatlah query untuk menampilkan daftar barang yang terjual pada tanggal tertentu! Kolom output minimal terdiri atas kode, nama barang, dan jumlah barang yang terjual. (20%)

**SELECT**

products.productcode,products.productname,orderdetails.quantityordered,orders.orderdate

**FROM** products,orders,orderdetails

**WHERE** orders.orderdate='2017-11-26' and products.productcode=orderdetails.productcode and orderdetails.ordernumber=orders.ordernumber;

1. Sama dengan soal kedua tetapi dibuat dalam bentuk store procedure dengan tanggal sebagai input, sebagai output adalah nilai uang yang diperoleh di tanggal tersebut, dan tabel daftar barang. (30%)

**DELIMITER //**

**CREATE PROCEDURE** getproduct (IN tanggal VARCHAR(25))

**BEGIN SELECT** products.productcode, products.productname, orderdetails.quantityordered, orders.orderdate

**FROM** products, orders, orderdetails WHERE orders.orderdate = tanggal and products.productcode = orderdetails.productcode and orderdetails.ordernumber = orders.ordernumber ;

**END //**

**DELIMITER**

1. Buatlah store procedure untuk menampilkan daftar customer yang melakukan pembelian (order) selama satu minggu terakhir, sebagai output adalam jumlah pelanggan dan daftar pelanggan. (25%)

**DELIMITER //**

**CREATE PROCEDURE** getcustomer()

**BEGIN SELECT** customers.customernumber, customers.customername,orders.orderdate

**FROM** customers, orders

**WHERE** customers.customernumber = orders.customernumber order by orders.orderdate desc limit 7;

**END**