Nama : Lia Dewi Susanti

NIM : L 200150102

Kelas : B

Tugas 1

Manajemen Sistem Basis Data

Rancangan Database Retail

Pengantar

Database pada sebuah sorum motor,dimana sorum itu juga mempunyai PT yang memproduksi.Rancangan dan detail produk disimpan pada data productilines.Setelah produk siap dipasarkan semua data terkait produk seperti kode produk,nama produuk,jumlah produk,dll disimpan pada entity products. Setelah produk dipesan,data pemesanan disimpan di order seperti nomor order,tanggal,waktu dan status dan detail dari pemesanan seperti nomor pemesanan,kode produk,jumlah pesanan disimpan di orderdetails. Data pelanggan yang melakukan pemesanan disimpan di table costumer termasuk nomor dari karyawan yang melayani. Data dari karyawan yang berada di kantor disimpan di table employees. Data mengenai informasi kantor disimpan di Offices. Data pembayaran yang dilakukan oleh costumer disimpan di Payments.

ER Diagram database retail



Perancangan Database Retail(Classicmodels):

1. Menentukan entitas (object-object dasar) yang perlu ada di database

* Customers : menyimpan data pelanggan
* Employees : menyimpan data karyawan
* Offices : menyimpan data karyawan yang ada di kantor
* Orderdetails : menyimpan detail pesanan
* Orders : meyimpan data pesanan
* Payments : meyimpan data transaksi
* Productlines : menyimpan data detail products
* Product : menyimpan data barang

1. Menentukan attributes(sifat-sifat) masing-masing entity sesuai kebutuhan database

* Customers
* customernumber : menyimpan kode pelanggan (int(11))
* customername : menyimpan nama pelanggan (varchar(50))
* contactlastname : menyimpan nama belakang pelanggan (varchar(50))
* contactfirstname : menyimpan nama depan pelanggan (varchar(50))
* phone : menyimpan nomor telepon pelanggan (varchar(50))
* addressline1 : meyimpan alamat pelanggan (varchar(50))
* addressline2 : menyimpan alamat pelanggan (varchar(50))
* city : menyimpan kota tempat tinggal pelanggan (varchar (50))
* state : menyimpan nama negara bagian tempat pelanggan tinggal (varchar(50))
* postalcode : menyimpan kode pos pelanggan (varchar(50))
* country : menyimpan nama negara tempat pelanggan tinggal (varchar(50))
* salesrepemployeenumber : menyimpan kode karyawan(kasir) yang melayani transaksi (int(11))
* creditlimit : menyimpan nomimal kredit yang dilakukan (decimal(10,2))
* Employees
* employeeNumber : nomor id untuk karyawan (int(11))
* lastname : nama belakang karyawan (varchar(50))
* firstname : nama depan karyawan (varchar(50))
* extension : (varchar(10))
* email : email karyawan (varchar(100))
* officecode : kode kantor (varchar(10))
* reports to : (int(11))
* job title : pekerjaan karyawan (varchar(50))
* Offices
* officecode : menyimpan kode kantor(int(11))
* city : menyimpan kota tempat tinggal pelanggan (varchar (50))
* phone : menyimpan nomor telepon pelanggan (varchar(50))
* addressline1 : meyimpan alamat pelanggan (varchar(50))
* addressline2 : menyimpan alamat pelanggan (varchar(50))
* state : menyimpan nama negara bagian tempat pelanggan tinggal (varchar(50))
* country : menyimpan nama negara tempat pelanggan tinggal (varchar(50))
* postalcode : menyimpan kode pos pelanggan (varchar(50))
* territory : menyimpan alamat daerah kantor(varchar(50))
* Orderdetails
* ordernumber : menyimpan kode detail pesanan (int(11))
* productcode : menyimpan kode produk (varchar(15))
* quantityordered : menyimpan jumlah pesanan(int(15))
* priceeach : menyimpan harga satuan produk(decimal(10,2))
* orderlinenumber : menyimpan nomer urutan pesanan(smallint(6))
* Orders
* ordernumber : menyimpan kode pesanan(int(11))
* orderdate : menyimpan tanggal pemesanan(date)
* requireddate : menyimpan tanggal (date)
* shippeddate : menyimpan tanggal pengiriman(date)
* status : menyimpan status pemesanan(varchar(15))
* comments : menyimpan komentar pemesanan(text)
* customernumber : menyimpan kode pelanggan (int(11))
* Payments
* customernumber : menyimpan kode pelanggan (int(11))
* checknumber : menyimpan nomer check(varchar(50))
* paymentdate : menyimpan tanggal transaksi(date)
* amount : menyimpan jumlah transaksi(decimal(10,2))
* Productlines
* productline : menyimpan urutan produk (varchar(50))
* textdescription : menyimpan deskripsi dari produk (varchar(4000))
* htmldescription : menyimpan link produk(mediumtext)
* image :menyimpan gambar produk (mediumblob)
* Product
* productcode : menyimpan kode produk(varchar(15))
* productname : menyimpan nama produk (varchar(70))
* productline : menyimpan urutan produk (varchar(50))
* productscale : menyimpan skala produk (varchar(10))
* productvendor : menyimpan nama toko dari produk (varchar(50))
* productdescription : menyimpan deskripsi produk (text)
* quantitystock : menyimpan jumlah stok produk(smallint(6))
* buyprice : menyimpan harga beli produk(decimal(10,2))
* MRSP : meyimpan tanggal kadaluarsa produk (decimal(10,2))

1. Menentukan relationship (hubungan) antar entitas

* productilines menjelaskan product
  + - tabel utama:productilines
    - tabel kedua:product
    - relationship:one-to-many (1:n)
    - attribute penghubung: productiline,productCode(FK productiline di product)
* product menerangkan orderdetails
  + - tabel utama: product
    - tabel kedua: orderdetail
    - relationship: one-to-many (1:n)
    - attribute penghubung: productCode,orderNumber(FK productCode di orderdetails
* orderdetails menerangkan order
  + - tabel utama:orderdetails
    - tabel kedua:order
    - relationship: one-to-many(1:n)
    - attribute penghubung: orderNumber,productCode(FK orderNumber,FK productCode di oerderdetails)
* orders dilakukan costumers
  + - tabel utama: orders
    - tabel kedua: costumers
    - relationship: one-to-many(1:n)
    - attribute penghubung:orderNumber,costumerNumber(FK oderNumber di costumers)
* costumers melakukan payment
  + - tabel utama:costumers
    - tabel kedua: payment
    - relationship: one-to-many(1:n)
    - attribute penghubung:costumerNumber,checkNumber(FK costumerNumber di payment)
* employees melayani costumer
  + - tabel utama:employees
    - tabel kedua: costumers
    - relationship: one-to-many(1:n)
    - attribute penghubung: employeeNumber,costumerNumber(FK employeeNumber di costumers)
* employees menempati office
* tabel utama:employees
* tabel kedua:office
* relationship: one-to-many(1:n)
* attribute penghubung:employeeNumber,officeCode(FK officeCode di employees