**NAMA : FITRIA WIDYA ASTARI**

**NIM : L200150027**

**KELAS : A**

**TUGAS 1**

**SISTEM MANAJEMEN BASIS DATA**

1. **Pengantar**

Basis data (database) adalah kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan pada data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi karena berfungsi sebagai gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghidari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit.

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (database management system | DBMS). DBMS merupakan sistem perangkat lunak yang memungkinkan pengguna basis data (database user) untuk memelihara, mengontrol dan mengakses data secara praktis dan efisien. Dengan kata lain, semua akses ke basis data akan ditangani oleh DBMS. DBMS ini menjadi lapisan yang menghubungkan basis data dengan program aplikasi untuk memastikan bahwa basis data tetap terorganisasi secara konsisten dan dapat diakses dengan mudah.

1. **ER Diagram**

Dari entitas dan atribut pada database classicmodels maka dapat ditentukan relasi sehingga dapat dibuat ER Diagram seperti dibawah:

1. **Keterangan**

ENTITAS & ATRIBUT

Pada ER Diagram database classicmodels memiliki 8 entitas berupa Offices, Customers, Employees, Payments, Orders, OrderDetils, Products, ProductLines.

Penjelasan dari masing-masing entitas;

1. Offices 2. Product

Memiliki atribut berupa: Memiliki atribut berupa:

* officesCode merupakan PRIMARY KEY - productCode merupakan PRIMARY KEY
* city - poductName
* phone - productLine merupakan FOREIGN KEY
* addressLine1 - productScale
* addressLine2 - productVendor
* state - productDescription
* country - buyPrice
* postalCode - MSRP
* territory - quantityInStock

1. Orders 4. OrderDetails

Memiliki atribut berupa: Memiliki atribut berupa:

* orderNumber merupakan PRIMARY KEY - orderNumber merupakan PRIMARY KEY
* orderDate - productCode
* comments - priceEach
* customerNumber merupakan FOREIGN KEY - orderLineNumber
* shippedDate - quantityOrdered
* status
* requiredDate

1. Customers 6. Employees

Memiliki atribut berupa: Memiliki atribut berupa:

* customerNumber merupakan PRIMARY KEY - employeeNumber merupakan PRIMARY KEY
* country - lastName
* city - firstName
* phone - officeCode merupakan FOREIGN KEY
* addressLine1 - reportsTo
* addressLine2 - jobTitle
* salesRepEmployeeNumber merupakan FOREIGN KEY - email
* customerName - extension
* contactFirstName
* contactLastName
* state
* postalCode
* creditLimit

1. Payments 8. ProductLines

Memiliki atribut berupa: Memiliki atribut beruoa:

* checkNumber merupakan PRIMARY KEY - productLine merupakvn PRIMARY KEY
* amount - image
* paymentDate - htmlDescription
* customerNumber merupakan FOREIGN KEY - textDesription

RELASI

Dari entitas dan atribut diatas dapat ditentukan relasi.  Relasi yaitu hubungan antara tabel yang mempresentasikan hubungan antar objek di dunia nyata. Relasi merupakan hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur mengatur operasi suatu database. Relasi dapat terjadi karena primary key pada suatu atribut berada/masuk pada atribut lain. Primary key yang berada/masuk pada atribut lain biasa juga disebut *foreign key*. Relasi yang tidak melibatkan *primary key* terdapat pada relasi 1:1(one-to-one) dimanarelasi tersebut mempresentasikan hubungan antar objek.

1. Offices - Employees

Diibaratkan seperti satu kantor memiliki(*have*) banyak pegawai. Sesuai dengan atribut berupa primary key pada Offices berada pada entitas Employees. Yang menunjukkan bahwa Employees merupakan *many*(**n**) dan Offices merupakan one(**1**). Sehingga relasi antara Offices dan Employees adalah 1:n (one-to-many).

1. Employees - Customers

Diibaratkan seperti satu pekerja melayani(*service*) banyak pelanggan. Sesuai dengan atribut berupa primary key pada Employees berada pada entitas Customers. Yang menunjukkan bahwa Customers merupakan *many*(**n**) dan Employees merupakan one(**1**). Sehingga relasi antara Employees dan Customers adalah 1:n (one-to-many).

1. Payments - Customers

Diibaratkan seperti banyak pembayaran dilakukan/dibuat (*made*) oleh satu pelanggan. Sesuai dengan atribut berupa primary key pada Customers berada pada entitas Payments. Yang menunjukkan bahwa Payments merupakan *many*(**n**) dan Customers merupakan one(**1**). Sehingga relasi antara Payments dan Customers adalah n:1 (many-to-one) atau (one-to-many).

1. Customers - Orders

Diibaratkan seperti satu Customers melakukan/menempatkan (*places*) banyak Orders. Sesuai dengan atribut berupa primary key pada Customers berada pada entitas Orders. Yang menunjukkan bahwa Orders merupakan *many*(**n**) dan Customers merupakan one(**1**). Sehingga relasi antara Customers dan Orders adalah 1:n (one-to-many).

1. Orders - OrderDetails

Diibaratkan seperti dalam satu Orders memiliki (*has a*) satu OrderDetails. Ditunjukkan bahwa Orders merupakan *one*(**1**) dan OrderDetails merupakan one(**1**). Sehingga relasi antara Orders dan OrderDetails adalah 1:1 (one-to-one).

1. Product - OrderDetails

Diibaratkan seperti satu Product memiliki (*have*) banyak OrderDetails. Sesuai dengan atribut berupa primary key pada Product berada pada entitas OrderDetails. Yang menunjukkan bahwa OrderDetails merupakan *many*(**n**) dan Product merupakan one(**1**). Sehingga relasi antara Product dan OrderDetails adalah 1:n (one-to-many).

1. ProductLines – Product

Diibaratkan seperti satu ProductLines menunjukkan (*belongs to*) banyak Products. Sesuai dengan atribut berupa primary key pada ProductLines berada pada entitas Products. Yang menunjukkan bahwa Products merupakan *many*(**n**) dan ProductLines merupakan one(**1**). Sehingga relasi antara ProductLines dan Products adalah 1:n (one-to-many).