CHAPTER 10 BASIC SYSTEM DATA BASE

The figure is a picture of cellular telephone which is also known as hand phone with Contact List application. The application is fairly familiar for those who has used a hand phone. The application keeps friend's name, parents as well as other information such as name and telephone number. Even sometimes complete with email address, office address, fax number and other data.

Examine the Contact List application, when you input someone name telephone number then the entered name and telephone number should not be switched with name or telephone number other people. All should be well sorted and arranged. You may find someone's name by only typing some letters part of the name. Please note that the name will always be alphabetically sorted in sequence, although your input may not be in sequence or sorted.

Figure 10.1. Contact list facility on a handphone.

Data name, telephone number and others data in application Contact List are stored in a database. In this chapter, we will study basic database concepts. The basic competence in **database management system (DBMS**) is part of standard competence that make application based on Microsoft Access and will be explain ed in detail in Chapter 11. The end of this chapter will be closed with summary and exercises.

THE OBJECTIVES

After studying this chapter, the reader is hoped to be able to,

- Explain the concept of data, database and database management system (DBMS).
- Explain Entity-Relationship Diagram.
- Explain database relational

10. 1. DATA, DATA BASE AND DATABASE MANAGEMENT SYSTEM

10.1.1. Database

Database is a collection of data that relates to each other, it is stored in computer hardware and use software to do the manipulation. Database is one of the main component in information system, as the base of information system for the users. (Fathansyah, 1999; Post, 1999).

Imagine a database as a cupboard in administrative school which keep various archives. Each type archives in group, arrange and keep in certain place. So there will some archives such as teacher's archives, subject archives,financial archives, student's archives, and others. Difference is only storage media. If filing cabinet use cupboard from wood, iron or plastic, but database use saving media electronics like disks (hard disc, CD, or tape). Figure 10,2 give illustration similarity filing cabinet and database.

Important is, database not only as electronically data storage. And not all electronically data storage called database. If archives do not use rules of arrange, separate or good organize, then we can not say storage data as data base. Figure 10.2 showing apply principles, illustration, organize or separately, for filing cabinet or and database.

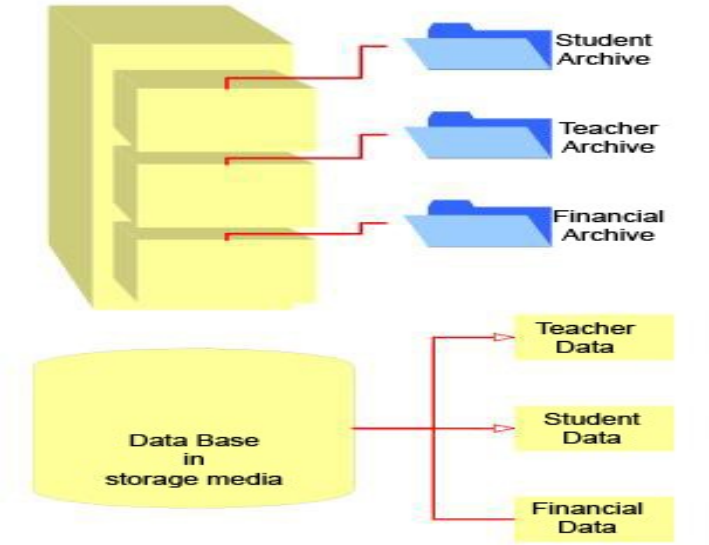


Figure 10.2. Archive and data base cupboard.

Main principle in database is independence data concept such as separation data from application program (Lewis et al., 2002; Post, 1999). While main goal of a

database is to help user in abstraction of a system. There are three (3) levels abstraction that usually use: physical level, conceptual level and view level (Figure 10.3). Physical level shows how data will be kept. Conceptual level relates type of data and the relation between data. View level is the highest level, it explains section of database for certain usage. (Ramakrishnan and Gehrke, 2000).

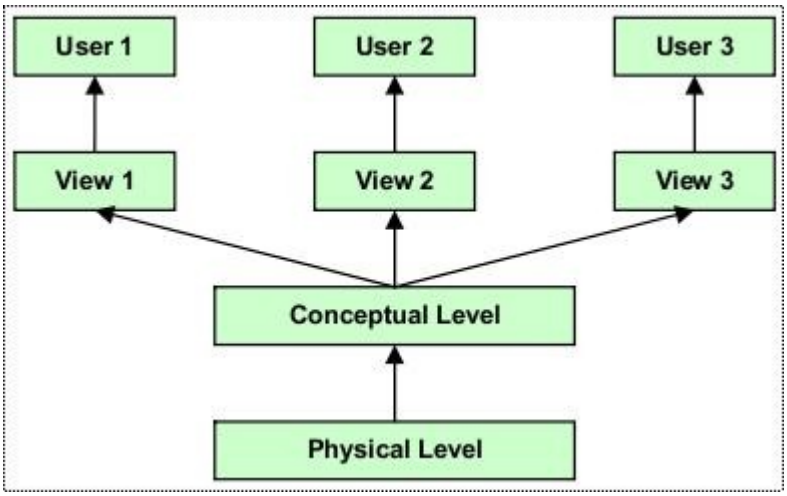


Figure 10.3. Abstraction at data level (Lewis et al., 2002).

Database have several important criteria, such as:

1. Should be data oriented and not program oriented.
2. Can be used by any application programs without changing the database.
3. Can be easily developed, both volume and structure.
4. Can easily satisfy new system requirements.
5. Can be used in many ways.

The structure of basic database operation is shown in Figure 10.4. These operations include.

- Create database. It is similar to the creation or purchase of new filing cabinet.
- Drop database. It is similar to the destruction filing cabinet.
- Create table. It is similar to add new archives group. This operation can only be used if the database is available.
- Drop table. It is similar to the destruct old archives group. This operation can only be used if table is available at database.

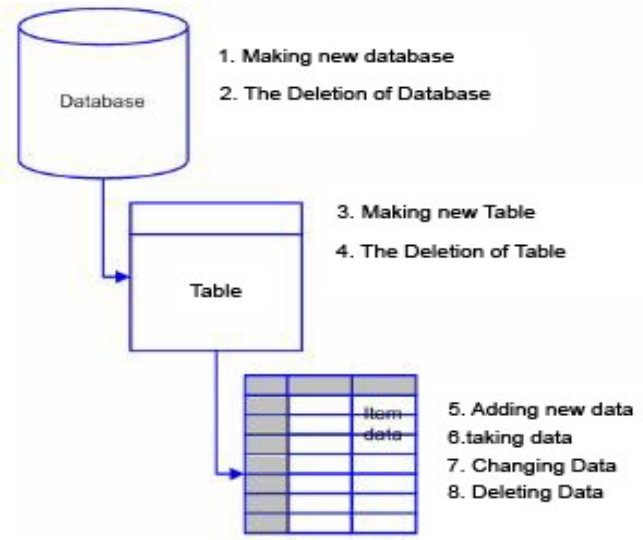


Figure 10..4. Database basic operations

- Insert data into table. This Operation is similar to add new sheet archives in archives group. Operation will be permitted if table is available.
- Retrieve data. This operation is similar to search sheet archives in archives group storage.
- Update data. This operation is similar to contents update of sheet archives in group archives.
- Delete data. This operation is similar to removal of sheet archives in archives group.

Database is build for certain data management goals, such as,

1. Efficiency covers speed, storage/space and accuracy
2. Handled data in large quantities.
3. Shareability.
4. Remove duplication and data inconsistency.

10.1.2. Database Management System

Database management may not be directly done by the users, handled by a special/ specific Software or DBMS that will determine how to organize data, keep, change and retrieve. It also applies security data mechanism, sharing, forced data accuracy &

accountability, etc. In summary, structure some DBMS may be seen in Figure 10.5.

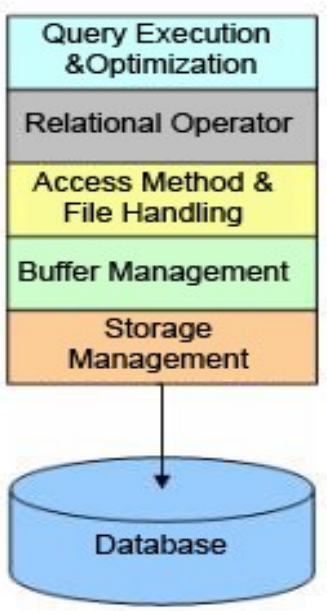


Figure 10.5. DBMS Structures

The additional applications are optional and usually available in DBMS as additional function. For example, application to make report, application to design form, application to make diagram or chart, application to monitoring system, and other applications.

There may be tens or hundreds DBMS software available. Each with different specification. From very simple to most complex. In this part we will explain five (5) kinds DBMS familiar to DMBS users, such as, Microsoft Access, MySQL, Microsoft SQL Server, PosgreSQL, and Oracle.

- **Microsoft Access**

  Microsoft Access, is also known as Microsoft Office Access, is a relational DBMS produced by Microsoft in Microsoft Office package. Microsoft Access make combination engine relational Microsoft Jet Database, Graphical User Interface (GUI) and software developer.

Figure 10.6. Logo MS Access

Microsoft Access can use data stored in Microsoft Jet Database, Microsoft SQL Server, Oracle or other ODBC (Open Database Connectivity) compatible.

Microsoft Access is often used in Rapid Application Development, especially to build prototype and application stand-alone. Microsoft Access can use as database for simple web based application. However in more complex application, for web or not, Microsoft Access is not a good choice. Especially, to handle multi-user. This because Microsoft Access is a personal single user database. Microsoft Access is also incomplete in database triggers and stored procedures.

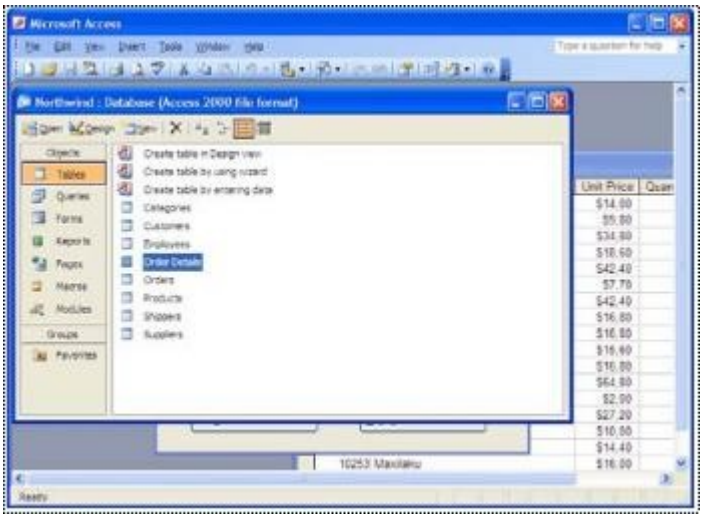


Figure 10.7. View Microsoft Access.

One of the advantage of Microsoft Access for the programmer is that compatibility to SQL (structured query language) is relatively high. In Microsoft Access, we can make query based on text or GUI, and execute to get results easily.

- **MyQSL**

MySQL is a multi-user and multi-threaded SQL-DBMS. MySQL may be run as a server to serve many users to access many databases. MySQL is very popular in the web based application as database component. MySQL is available for free. In addition,

Figure 10.8. Logo MySQL

MySQL is influenced by popular web server Apache and programming language PHP. All technical term such as LAMP (Linux- Apache-MySQL-PHP/Perl/Python), MAMP (Mac-Apache-MySQL-PHP/Perl/Python) and WAMP (Windows-Apache-MySQL-PHP/ Perl/Python) become very famous. Plenty web base applications are build by using these software combinations, such as, WordPress, Drupal, Mambo, Wikipedia, PHP-Nuke.

Unlike Microsoft Access, the MySQL default installation provides no GUI interface for users to interact with the MySQL database. User interact with commands line interface. However, there are many GUI developed to facilitate interaction with the MySQL database, either for application stand-alone (for example MySQL-Front, MySQL-GUI, etc) or for web based (for example, phpMyAdmin). Moreover by the use of MyODBC component, MySQL may access Microsoft Access database similar to other ODBC compatible.

Figure 10.9. Opening view of phpMyAdmin.

- **Microsoft SQL Server**

Microsoft SQL Server is a relational DBMS produced by Microsoft same Microsoft Access. The main query language is used a variant ANSI SQL known as T-SQL (Transact-SQL). User can access this language to create stored procedure to increase efficiency access to the database. DBMS also facilities clustering and mirroring. Cluster is a collection of servers with identical configuration to enable distribution of work between server. Whereas mirroring facility allows DBMS to make a complete replica of the contents database to be use in another server.

Figure 10.10. Logo Microsoft SQL Server

Microsoft SQL Server is available in several distribution versions. Joint distribution between Microsoft SQL Server - Microsoft Office or Microsoft Visual Studio, normally call as MSDE (Microsoft SQL Server Database Engine), but no GUI provided. User may use text based client interface. Similar to MySQL. While the higher version as personal or professional version are available with GUI facilities as shown in Figure 10.11.

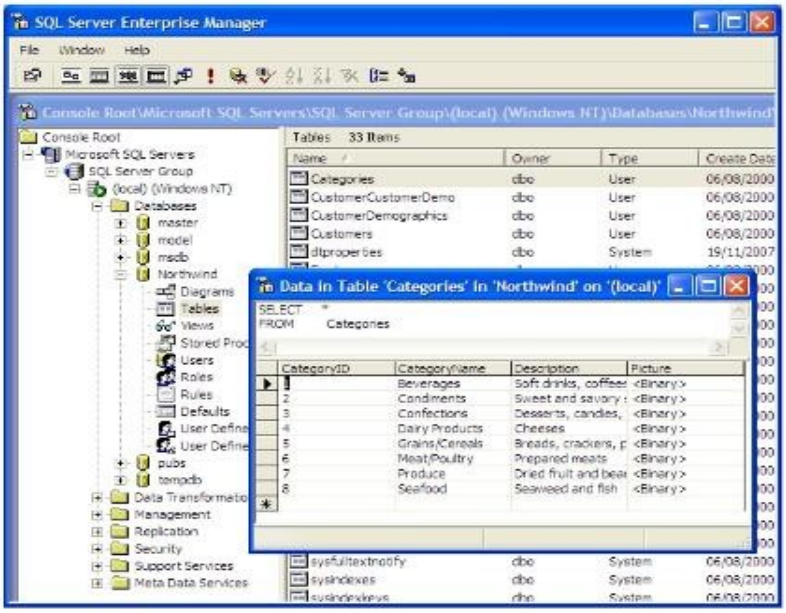


Figure 10.11. GUI on Microsoft SQL Server

- **PostgreSQL**

PostgreSQL, also known as Postgres, is an Object-Relational Database Management System (ORDBMS). An ORDBMS is a DBMS that uses not only principles relational database but also object oriented approach in the database models. Postgres is a free software and open-source, and, thus, not controlled by one or two companies.

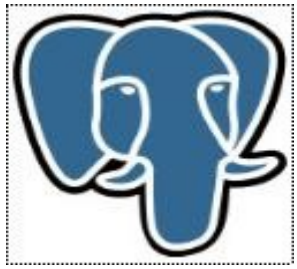


Figure 10.12. PostgreSQL Logo

The advantage of Postgres as compared with other DBMS is in its characteristics as open source and free, supported by an extraordinary documentation. Its flexibility and feature is not different from commercial DBMS. In addition to support for object-relational data model, Postgres also supports spatial database usually for Geographic Information System. Postgres also supports multi-user and multi-threaded operation. In fac, it is better than MySQL from security side.

Similar to MySQL and Microsoft SQL Server, we can interact with Postgres by using commands client side with tool, such as, psql. GUI interface has also been developed, such as, phpPgAdmin, PgAdmin, etc.

- **Oracle Database**

  Name of Oracle Database or Oracle RDBMS is well known in the DBMS world. Oracle is developed by Oracle Corporation.

Figure 10.13. Oracle Logo

  Oracle keep data in logically in tablespaces form and physically in data files form. Tablespaces can contain various memory, such as, data part, index part etc. These parts contain one or more area. The areas contain collection block data. Oracle can automatically keep and store procedure and function stand alone.

10.2. ENTITY-RELATIONSHIP DIAGRAM

E-R Model is normally shown in diagram and called Entity-Relationship Diagram (ERD). ERD is a graphic notation of conceptual data model that is used to make structural data and relations between data models. With ERD we can test model by ignoring further process. With ERD we try to answer questions, such as; what data we need? How one data connect with others? ERD use notations and symbol to show structure and relations between data. It has basically three (3) symbols, such as.

1. Entity: Entity is shown in a rectangle form.

2. Attribute: Attribute is shown in form of ellipse and connect with entity where attribute is.

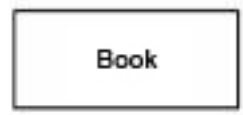


Figure 10.14. Notation entity on ER-Diagram

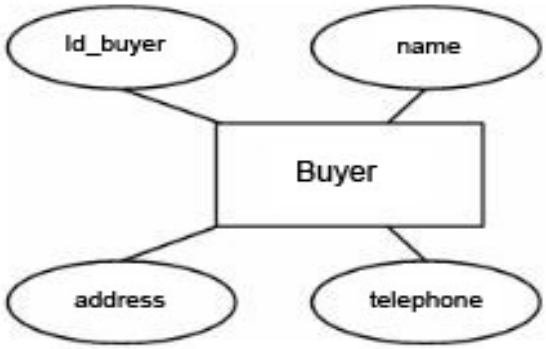


Figure 10.15. The use of attribute notation to ER-Diagram.

3. Relationship: Relationship is shown in diamonds form.

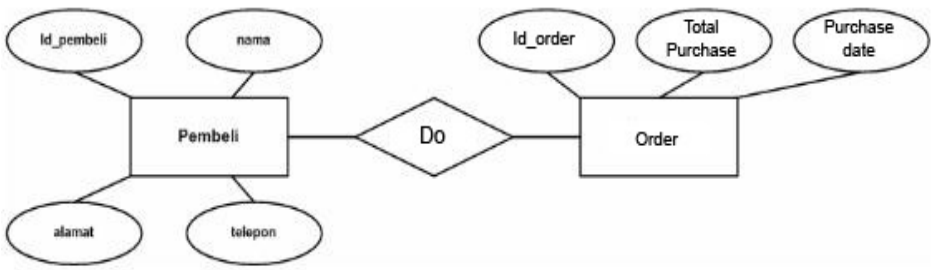


Figure 10,16. Use notation relationship to ER-Diagram.

10.2.1. Entity

Entity is an individual to represent a real matter / thing and may be differentiate to one another. Could be an element from an environment, a resources or a transaction that very important in the built model.

Example of Entity set:

- All Teacher or A Teacher.
  This association has member: Mr Fahri, Ms Fitri, Mr Joko and other teachers.

- All Student or A Student.

This association has member: Joni, Ridho, Fanny, Donny and other students.

- All Cars or A Car.
  This association has member: Daihatsu car, Toyota car, Suzuki car, and other cars.

Its not easy to Identify the existence of an entity. The existence of noun words in a problem is usually an entity candidate. For example when we will develop the database in school library, then we will find book, student, teacher, library officer as strong candidate entity.

10.2.2. Attribute

Each entity has attributes. An attribute has characteristic to be distinguished between one entity and other entity.

Example of Attribute:

- Student Entity.

  Has attribute, such as, : Student ID number, name, address, telephone number, birthplace, date birth etc.

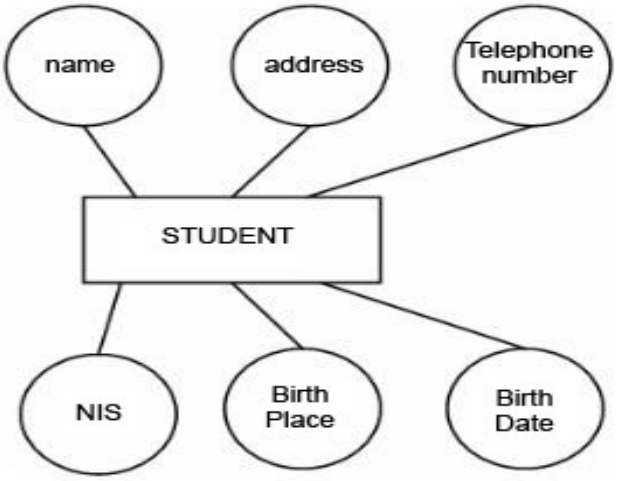


Figure 10.17. Student's entity and attributes.

- **Teacher entity.**

  Has attribute such as: NIP, name, address, rank, telephone number, birthplace, date of birth, expertise field, etc.

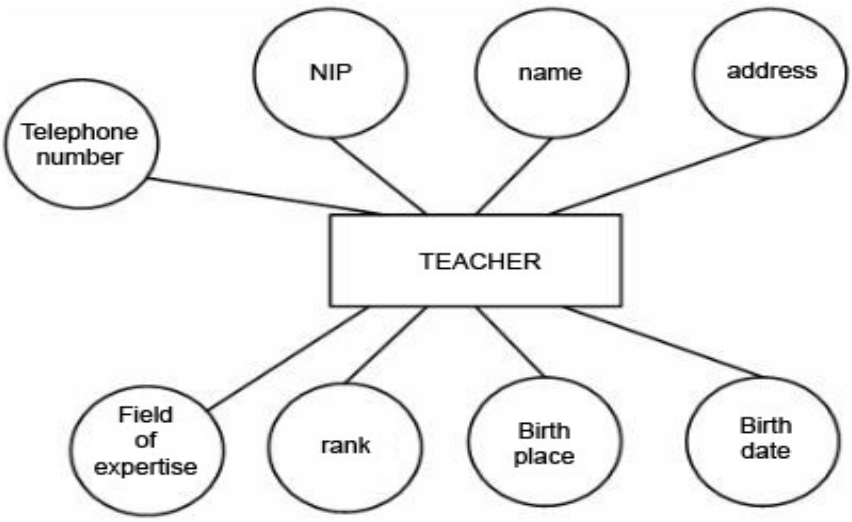


Figure 10.18. Teacher's entity and attribute.

- **Car entity.**

  Has attributes, such as, machine number, framework number, color, year produced, machine type, fuel, etc.

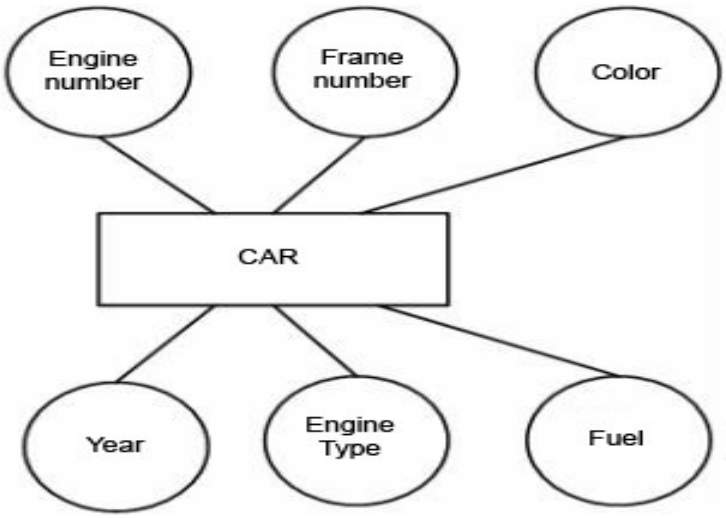


Figure 10.19. Car entity and attribute.

Not all entity characteristics is important for a scope problem. For example problem in library's database, student's shoe number is not an important characteristics to be used as an attribute. But if the scope of problem is in database for ordering student's uniform and student's shoes, then shoes number is an important attribute.

In addition to the most important entity, we must choose one of the attribute as the primary key.

In the above attribute example, we can easily decide the primary key from available entity.

Example Primary key:

- **Student Entity.**

  The available attributes are Student ID number, name, address, telephone number, birth place, and date birth. From these six (6) attributes the most appropriate attribute to become the primary key is Student ID number as it is a unique attribute. No student has same Student ID Number. Student name may be the same, but not in Student ID Number.

- **Teacher entity.**

  The available attributes are NIP, name, address, rank, tel number, birthplace, date birth, and expertise field. The most appropriate attribute to become the primary key is the NIP because it is unique. There is no teacher with the same NIP. So NIP can use identify teacher's entity.

- **Car entity.**

  The available attributes are machine number, frame number, color, year produced, machine type, and fuel. There are two unique attributes, namely, machine number and frame number. In this case we can choose one of them as primary key.

As guidance, here are some attribute characteristics that is considered as identifier (candidate key):

- Value does not change.
- Impossible contain null value (empty not zero)
- Not contain data name or location that possibly change

10.2.3. Relationship

Relationship or relations is the connectivity between entity. For example from student's entity there is a student who has entity NIS = “GHI007” and student name = “Donny” has relations with entity program subject = “RPL” and program name = “Software Engineering”. Relations between these two (2) entities means student is currently taking the program subject in certain school.

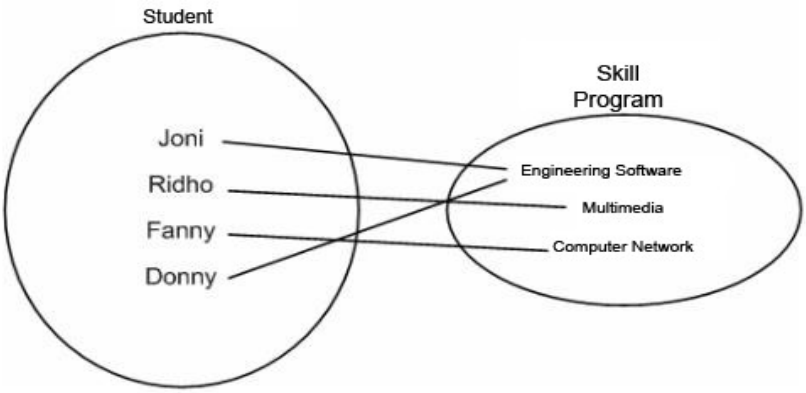


Figure 10.20. Relationship.

Ramakrishnan and Gehrke (2000) describe that the relationship concept in E-R model is different from the relational concept in data relational model. Relationship is mechanism that connect between entity. In implementation into DBMS both entity and relationship will be represented in relational table form.

Each relationship must have a cardinality. Cardinality or Relations Level shows the maximum number of entity that may be relate to other entity in other entity association.

In Figure 10.20, we can see cardinality between association student's entity and association entity of program subject. Student can make a relation with only one entity at association entity of program subject. On the other hand one entity in program subject may have relation with many students. See in the figure that Donny can only connect with Software Engineering, whereas Software Engineering can connect with Donny and Joni.

There are several level relations (cardinality) between one entity and other entity. Cardinality show the maximum number entity in association of entity and can relate with entity in other association entity. Generally there three (3) forms of cardinality between association entity, such as:

- One to one. Relations with cardinality one-to-one means that one member of entity set may only be connected with one member of other entity set. Relations between entity husband and wife can be group in relations one-to-one.

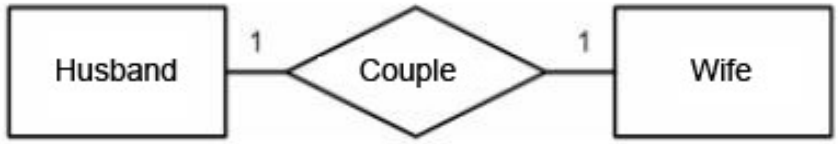


Figure 10.21. Relations one-to-one husband and wife.

- One to many/many to one. Cardinality one to many and many to one are the same because the analysis cardinality always see from both sides. For example in school has rule that one class consist many students but not the other way, one student can not learn in different class.

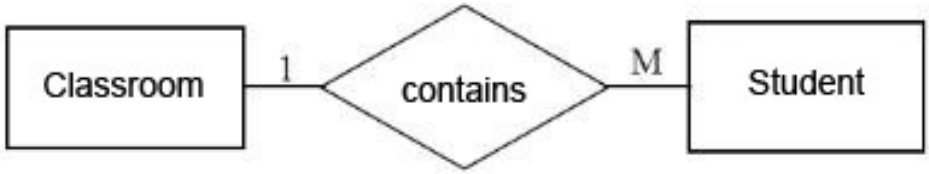


Figure 10.22. Relations one-to-many class with student.

- Many to many. Cardinality is complicated enough to explain but we often see this problem. For example , student's relations with subject has many-to-many cardinality. Student has right to study more than one subject and each subject may take study more than one student.

10.3. RELATIONAL DATA BASE

10.3.1. RELATIONAL DATA BASE MODEL

Relational database is introduced for the first time by E.F. Codd during 1970”s. This model data is based on mathematical structure to be easy and natural, namely is relation table. Data operation manipulation is originated from logical mathematics. Thus, it makes the expressions in table may be analyzed and optimized (Lewis et.al., 2002).

The main form in relational data model is the relation table. Relation consists of two (2) important matters, namely, schema and instance. Relation instance is a table not

more than two dimensions, with row and column. Row is normally called tuple, it has the same meaning as record in file. But different from file record, all tuple have the same number of columns and there is no tuple in same relation instance. Column in relation instance is also populer as attribute or column (Ramakrishnan and Gehrke, 2000; Lewis et.al., 2002). Figure 10.23 show the relation between table / file / relation, row / record / tuple and column / field / attribute. This Figure also shows the composition of the relations.

Table	Row	Column
File or Date File	Record	Field
Relations	Tuple	Attributes

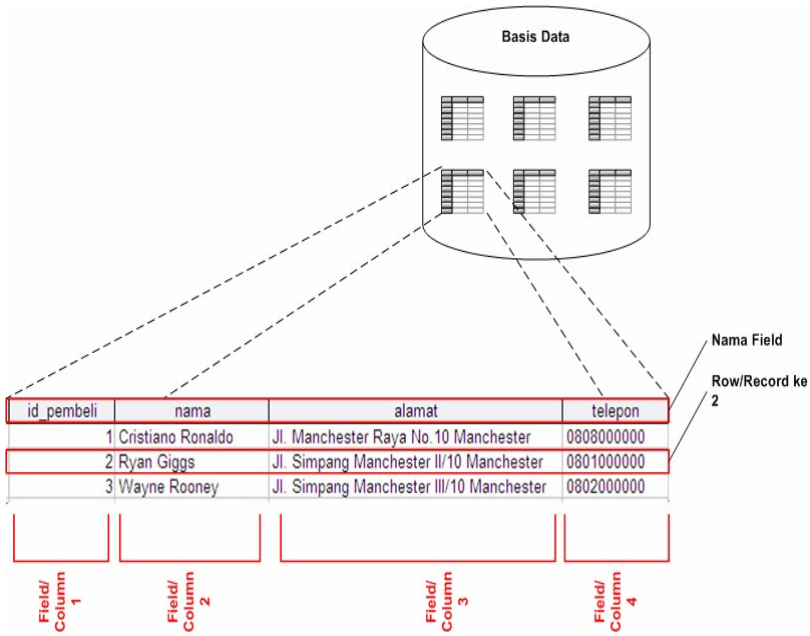


Figure 10.23. Relation between table/file/relation, row/record/tuple and column/field/attribute.

Relation schema consists of relation name, attribute name in the relation with its domain, and integrity constrains. Relation name must be unique in a database, and no other same relation name. Attribute name is a column name from relation and no same attribute name in a relation. Domain name from an attribute relates to the data type used in the attribute. Integrity constraints are the constrains to relational instances to a schema (Ramakrishnan and Gehrke, 2000; Lewis et.al., 2002)

10.3.2. Relational Database Structure

As explained, a table consists of row and column.

- Row / Lines / Tuple.

  Row / Lines / Tuple is a collection of attributes that were connected with each other in one row as shown in Figure 10.23. Row is always repeated in the same structure but different in its content. For example, in Figure 10.23 the first row has the same structure as row to 2 and to 3. However, the data in each row is different. We could identify for buyer with id_buyer = 1 must have name of = Cristiano Ronaldo and id_buyer = 2 must have name of = Ryan Giggs, and so on.

- Field / Column / Attributes

  Field / Column / Attributes shows the data structure from repeated rows. In Figure 10.23 it has four (4) columns, namely, id_buyer, name, address, and telephone. For example, the data in id_buyer column will have the same structure as in form of number and is sorted for id_buyer. Also for column name with only contain buyer name so as to not mixing with other data.

  A column must have column name and data type for data in the column as shown in Figure 10.24. In addition, there are sometimes constraint and domain for data in the column. Data type depends on used attribute and the operations on the database. Domain data has similar understanding with used data type function. However, data type refer to data storage capacity which is more physical meaning. Meanwhile domain data is more stressed in value constraints of an attributes, as seen in real usage.

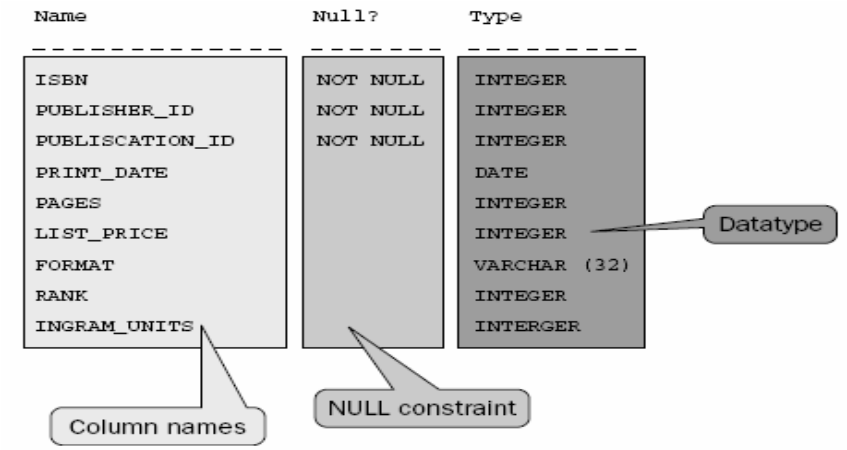


Figure 10.24. Column, constrain and data type (Powell, 2006).

Figure 10,24 shows the column names in a complete table with data type and its constrain. For example, ISBN column is an integer data type, means the column may only be filled by integer number. In addition, the column has no null constrain and, thus, the column may not be emptied during data entry. Pages column with data type date means only contain date data. Pages column does not have constrain null. It means the column can be emptied or not.

The data type depends on the DBMS. For example, MySQL provides more data type options than Microsoft Access. For example, MySQL provides enum data type that is not available in Microsoft Access. An accurate data type selection is very important and may influence data consistency and database performance.

Example of a domain, if use attribute / column on class in Elementary School. This column may be filled by data type number from 1 up to 6, as there is no class no class 7 or 4.5 . It means domain class column is an integer number only between 1 up to 6. Another example, the exam score is a floating point number between 0 up to 100.

There are several types of attribute, namely,

- Simple Attribute is atomic attribute cannot be divided anymore.

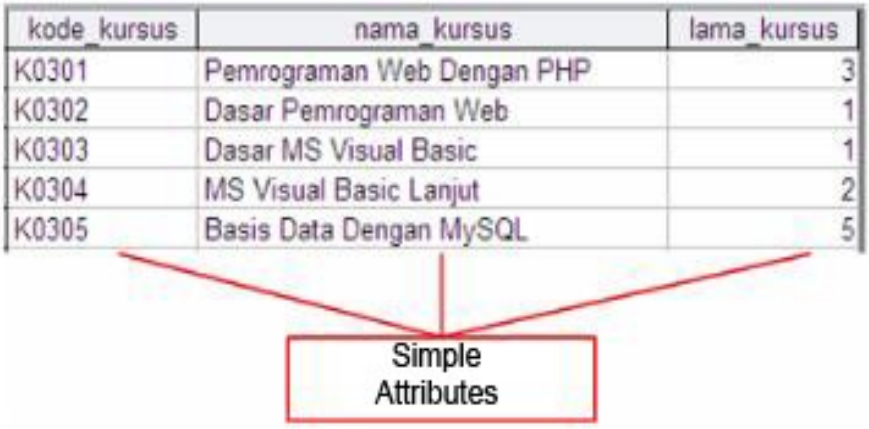


Figure 10,25. Simple attribute

- Composite Attribute is an attribute that can be divided into sub attributes and each has meaning.

Figure 10.26. Composite attribute

- Single-Valued Attribute, is an attribute that has a maximum one value for each data line.
- Multi-Valued Attribute, is an attribute that contains more than one value in same type.

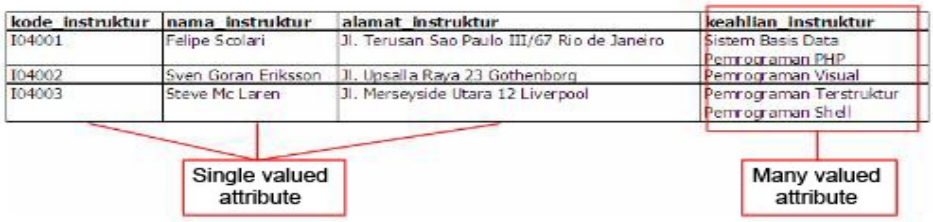


Figure 10.27. Single valuable attribute and multi-value attribute.

- Derived Attribute is attribute that receive value from data processing or from attribute or connect other table.

10.3.3. Inter Table Relationship

The advantage of relational database as compared to other database model is in the inter table relationship to be easy to understand and logical. Inter table relation may be directly or indirectly derived from ER-Diagram. Entity in ER-Diagram is usually a candidate from table in relational database. Relationship between table is usually may be identified from relationship between entity in ER-Diagram.

To clarify how relational database handled relations between tables, we will use the following tables. The First table is the Author. This table consists of three (3) columns: id_ author, author and birth_year as shown in Figure 10.28. In this table, id_author is primary key.

Author : Table

id_author	author	birth_date
1	Adams, Pat	
2	Adrian, Merv	
3	Ageloff, Roy	1943
4	Andersen, Virginia	
5	Antonovich Michael P.	
6	Arnott, Steven E	
7	Arntson, L. Joyce	
8	Ault, Michael R	
9	Avison, D. E	
10	Bard, Dick	1941
11	Biegel, Richard A	
12	Blow, Lisa	
13	Bisland, Ralph B	
14	Bowman, Judith S	

Figure 10.28. Table Author.

The Second table is the Publisher. This table has six (6) columns, namely, id_publisher, name, company_name, address, city and telephone as shown in Figure 10,29. The Primary key of this table is the id_publisher.

Penerbit : Table

id_penerbit	nama	nama_perusahaan	alamat	kota	telepon
1	ACM	Association for Computing Machinery	11 W. 42nd St., 3rd flr.	New York	212-869-7440
2	Addison-Wesley	Addison-Wesley Publishing Co Inc.	Rte 128	Reading	617-944-3700
3	Bantam Books	Bantam Books Div of: Bantam Doubleday De	666 Fifth Ave	New York	800-223-6834
4	Benjamin/Cummings	Benjamin-Cummings Publishing Company S	390 Bridge Pkwy.	Redwood City	800-950-2665
5	Brady Pub.	Brady Books Div. of Prentice Hall Pr., Simor	15 Columbus Cir.	New York	212-373-8093
6	Computer Science Press	Computer Science Press Inc Imprint of W H	41 Madison Ave	New York	212-576-9400
7	ETN Corporation	ETN Corp.	RD 4, Box 659	Montoursville	717-435-2202
8	Gale	Gale Research, Incorporated	835 Penobscot Bldg	Detroit	313-961-2242
9	IEEE	IEEE Computer Society Press	10662 Los Vaqueros Circle	Los Alamitos	800-272-6657
10	Intertext	Intertext Publications/Multiscience Press	2633 E. 17th Ave.	Anchorage	
11	M&T Books	M & T Books Div of: M&T Publishing Inc	501 Galveston Dr	Redwood City	800-533-4372
12	Macmillan Education	Macmillan Education Ltd	175 Fifth Ave	New York	212-460-1500
13	McGraw-Hill	McGraw-Hill Inc	1221 Ave of the Americas	New York	212-512-2000
14	Microsoft Press	Microsoft Press Div of: Microsoft Corp	One Microsoft Way	Redmond	800-MSPRESS
15	Morgan Kaufmann	Morgan Kaufmann Publishers Inc.	2929 Campus Dr, Suite 260	San Mateo	415-578-9911

Figure 10.29. Table Publisher.

The Third table is Book. This table has six (6) columns, namely, title, year_issued, ISBN, id_publisher, description, and group as shown in Figure 10,30. The Primary key to this table is the ISBN. There is something strange in this table as column id_publisher whih is one of the column in Table Publisher is added in Table Book. This

is not a mistake, but one of the way for a relational database to handle relations between tables.

Buku : Table

judul	tahun_terbit	ISBN	id_penerbit	deskripsi	kelompok
Database management; developing application systems usin	1989	0-0131985-2-1	17	xx, 441 p. : ill. ;	
Select-- SQL ; the relational database language	1992	0-0238669-4-2	12	xv, 446 p. ; 24 c	
dBase IV programming	1994	0-0280042-4-8	73		
Step-by-step dBase IV	1995	0-0280095-2-5	52		
Guide to ORACLE	1990	0-0702063-1-7	13	xii, 354 p. : ill. ; 2	ORACLE (Computer syste
The database experts' guide to SQL	1988	0-0703900-6-1	10		
Oracle/SQL; a professional programmer's guide	1992	0-0704077-5-4	13	xx, 543 p. : ill. ;	
SQL 400: A Professional Programmer's Guide	1994	0-0704079-9-1	52		
Database system concepts	1986	0-0704475-2-7	13		
Microsoft FoxPro 2.5 applications programming	1993	0-0705015-3-X	61	xiii, 412 p. : ill. ;	
First look at-- dBASE IV, version 1.5/2.0 for DOS	1994	0-0705107-5-X	80	ill. ; 24 cm.	
Applying SQL in Business	1992	0-0705184-2-4	13		
Database design	1977	0-0707013-0-X	13		
Introduction to Oracle	1989	0-0770716-4-6	13	xi, 342 p. ; 24 cı	

Figure 10.30 Table Book

In the real world, we will find that a publisher is not only produce a single book title, but hundreds even possibly millions of book titles. Thus, publisher and book relations may be formally called as cardinality one-to-many. Its ER-Diagram is shown in Figure 10.31. One publisher may publish many book titles and one book title may only published by one publisher. Please examine Figure 10.31, the publisher with id_publisher= 13 (McGraw Hill) publishes 6 books. On the other hand, book with ISBN = 0-0702063-1-7 (Guide To Oracle) is only published by a publisher with id publisher= 13 (McGraw Hill). So that column id_publisher in table book means to presentation Publisher relations to Book. Column id_publisher to table book is usually called as foreign key.

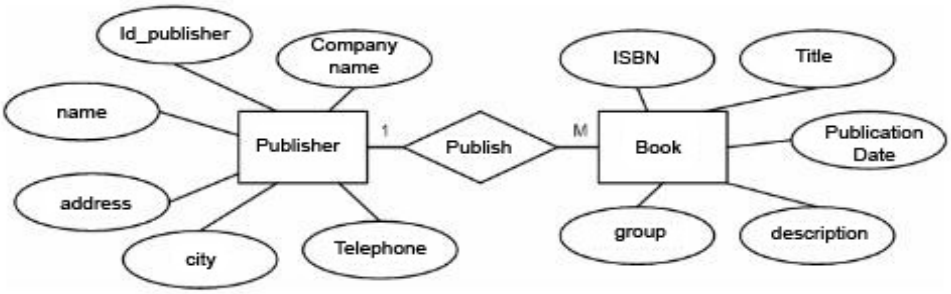
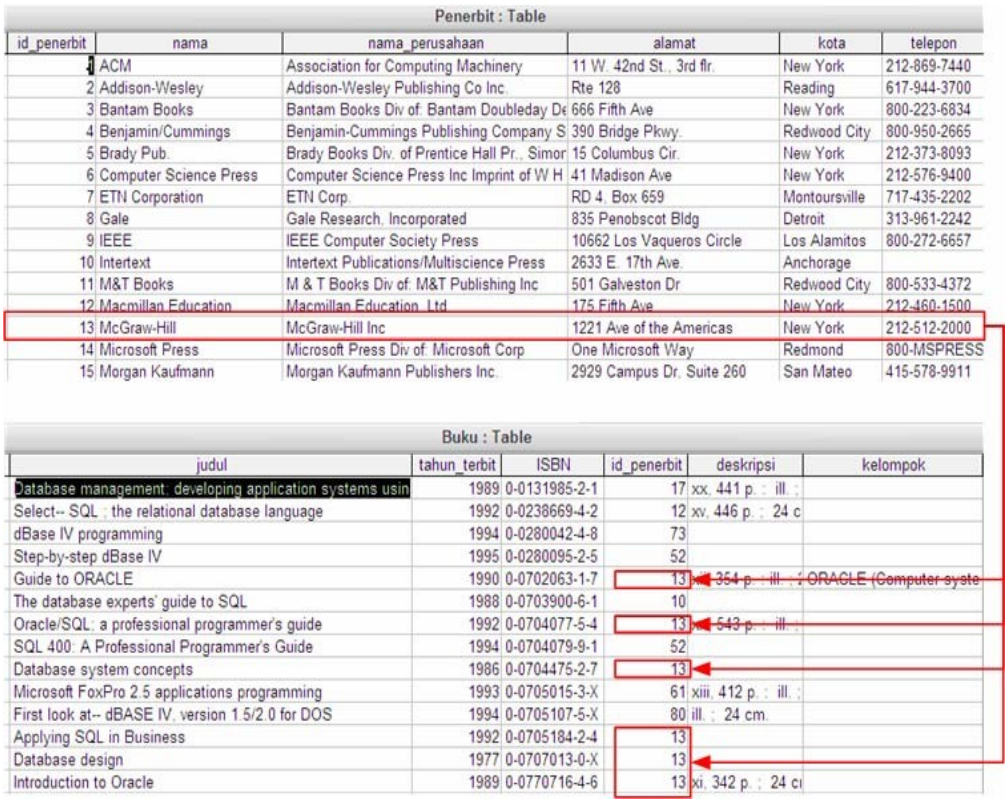


Figure 10.31. ER-Diagram for Publisher and Book



Penerbit : Table

id_penerbit	nama	nama_perusahaan	alamat	kota	telepon
1	ACM	Association for Computing Machinery	11 W. 42nd St., 3rd flr.	New York	212-869-7440
2	Addison-Wesley	Addison-Wesley Publishing Co Inc.	Rte 128	Reading	617-944-3700
3	Bantam Books	Bantam Books Div of: Bantam Doubleday De	666 Fifth Ave	New York	800-223-6834
4	Benjamin/Cummings	Benjamin-Cummings Publishing Company S	390 Bridge Pkwy.	Redwood City	800-950-2665
5	Brady Pub.	Brady Books Div. of Prentice Hall Pr., Simor	15 Columbus Cir.	New York	212-373-8093
6	Computer Science Press	Computer Science Press Inc Imprint of W H	41 Madison Ave	New York	212-576-9400
7	ETN Corporation	ETN Corp.	RD 4, Box 659	Montoursville	717-435-2202
8	Gale	Gale Research, Incorporated	835 Penobscot Bldg	Detroit	313-961-2242
9	IEEE	IEEE Computer Society Press	10662 Los Vaqueros Circle	Los Alamitos	800-272-6657
10	Intertext	Intertext Publications/Multiscience Press	2633 E. 17th Ave.	Anchorage	
11	M&T Books	M & T Books Div of: M&T Publishing Inc	501 Galveston Dr	Redwood City	800-533-4372
12	Macmillan Education	Macmillan Education, Ltd.	175 Fifth Ave.	New York	212-460-1500
13	McGraw-Hill	McGraw-Hill Inc	1221 Ave of the Americas	New York	212-512-2000
14	Microsoft Press	Microsoft Press Div of: Microsoft Corp	One Microsoft Way	Redmond	800-MSPRESS
15	Morgan Kaufmann	Morgan Kaufmann Publishers Inc.	2929 Campus Dr. Suite 260	San Mateo	415-578-9911

Buku : Table

judul	tahun_terbit	ISBN	id_penerbit	deskripsi	kelompok
Database management: developing application systems usin	1989	0-0131985-2-1	17	xx, 441 p. : ill. ;	
Select-- SQL : the relational database language	1992	0-0238669-4-2	12	xv, 446 p. ; 24 c	
dBase IV programming	1994	0-0280042-4-8	73		
Step-by-step dBase IV	1995	0-0280095-2-5	52		
Guide to ORACLE	1990	0-0702063-1-7	13	x, 354 p. : ill. ; ORACLE (Computer syste	
The database experts' guide to SQL	1988	0-0703900-6-1	10		
Oracle/SQL: a professional programmer's guide	1992	0-0704077-5-4	13	x, 543 p. : ill. ;	
SQL 400: A Professional Programmer's Guide	1994	0-0704079-9-1	52		
Database system concepts	1986	0-0704475-2-7	13		
Microsoft FoxPro 2.5 applications programming	1993	0-0705015-3-X	61	xiii, 412 p. : ill. ;	
First look at-- dBASE IV, version 1.5/2.0 for DOS	1994	0-0705107-5-X	80	ill. ; 24 cm.	
Applying SQL in Business	1992	0-0705184-2-4	13		
Database design	1977	0-0707013-0-X	13		
Introduction to Oracle	1989	0-0770716-4-6	13	xi, 342 p. ; 24 c	

Figure 10.32. Relations Table Publisher and Book.

In the above tables, we don't see the boot title of each Author. To make table contains author and book title, we relates table author and table book. Prior to it, we must see the relations between writer and book in the real world. A author perhaps only write one title book, but possibly more. Whereas one book is possibly written by one or more author. So we can see the cardinality author's relation with book is many-to-many. Figure 10.33 shows the ER-Diagram.

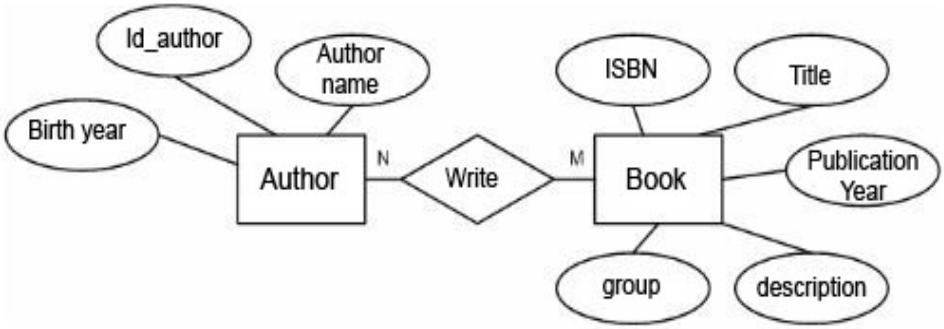


Figure 10.33. ER-Diagram for Author - Book

In case with cardinality many-to-many we can not directly insert a foreign key to other table. We must make new table so that cardinality between table involved may be changed to one-to-many. Table Author_Book is table that perform handle relations Book table with Author table. This table only contains two attributes, namely, ISBN column that originates from table Book and id_author that originates from table Author. Figure 10.34 shows table Author_Book that shows several books may be written by more than one author.

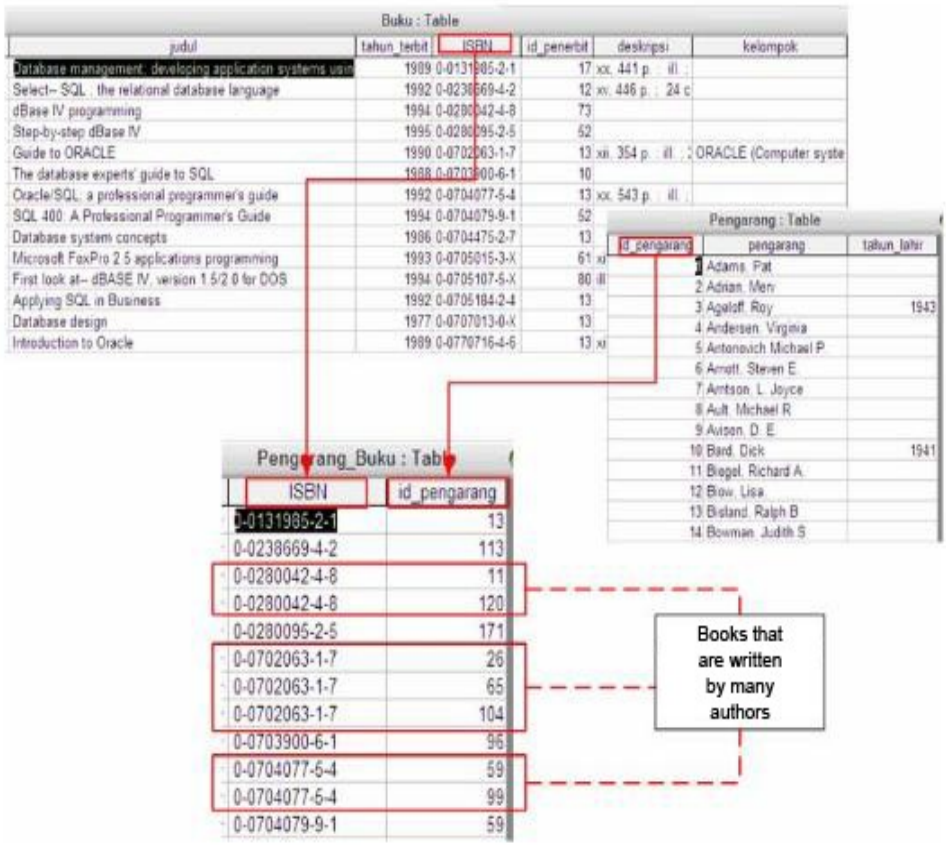


Figure 10.34. Table Relations Author and Book

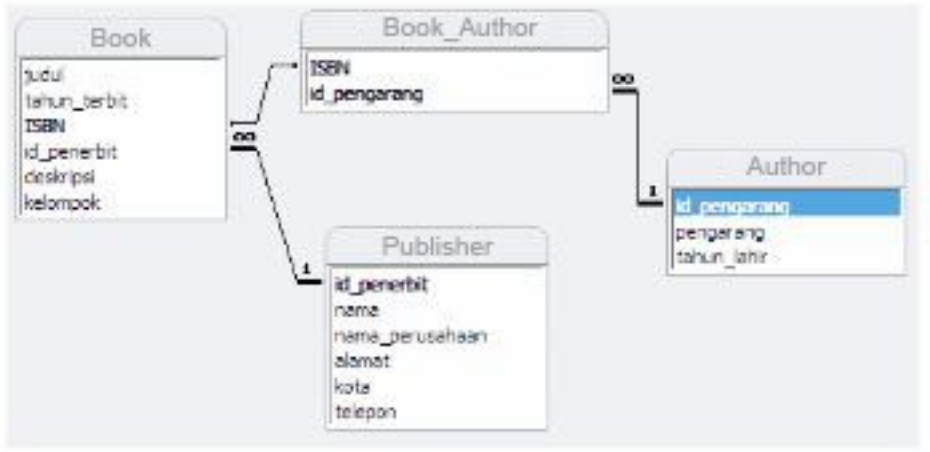


Figure 10.35. Relation between table.

10.4. SUMMARY

- A database is collection of data that connect to each other, stored in a computer hardware and uses a software to manipulate data.

- The basic operation of database consists of creation of new database, drop of database, creation of new table, drop of table, add new data, retrieve data, modify data, remove on data.
- Database Management System is a software especially work to handle database
- Entity-Relationship Diagram is a graphic notation to conceptualized data model and use to make structural data model and relations between data. Element from ERD are entity, attribute, relationship and cardinality.
- The main form of model data relational is a relation (table). Relation instance is a two dimensional table with row / record / tuple and column / field / attribute.
- In relational database, exact identification in attribute relationship in a table and relationship among tables is the key to make a good database.

10.5. EXERCISES

1. Visit your school library, then do a short observation. Make notes to determine who and what involves in the activity in the library school . Pay attention on the entity and attribute. And how the relationship between entity.
2. From no. 1 activity, make tables that shows entity and its relationship. Make attributes from each table.
3. Pay close attention to attribute types.

REFERENCES

Anonymous. 2004. Guide to the Software Engineering Body of Knowledge (SWEBOK). The Institute of Electrical and Electronics Engineers, Inc.

Balter, A. 2006. Sams Teach Yourself Microsoft® SQL ServerTM 2005 Express in 24 Hours. Sams.

Bass, L., P. Clements, and R. Kazman. 2003. Software Architecture in Practice. 2nd Edition. Addison-Wesley.

Cormen, T.H. 2001. Introduction to Algorithm: Second Edition. The MIT Press.

Deek, FP., J.A.M. McHugh, and O.M. Eljabiri. 2005. Strategic software engineering : An Interdisciplinary Approach. Auerbach Publications.

den Haan, P., L. Lavandowska, S.N. Panduranga, and K. Perrumal. 2004. Beginning JSP 2: From Novice to Professional. Apress.

Dobson, R. 1999. Programming Microsoft Access 2000: The Developer's Guide to Harnessing the Power of Access. Microsoft Press.

Felleisen, M, R.B. Findler, M. Flatt, and S. Krishnamurthi. 2001. How to Design Programs; An Introduction to Computing and Programming. The MIT Press.

Kak, A.C. 2003. Programming With Objects: A Comparative Presentation of Object Oriented Programming with C++ and Java. John Wiley & Sons, Inc.

Kaisler, S.H. 2005. Software Paradigm. John Wiley & Sons, Inc.

Kennedy, B. and C. Musciano. 2006. HTML & XHTML: The Definitive Guide, 6th Edition. O'Reilly.

Lafore, R. 1998. Data Structures & Algorithm in Java. Waite Group Press. 2nd Edition.
Laurie, B and P. Laurie. 2001. Apache: The Definition Guide. O'Reilly and Associates, Inc.

Leffingwell, D. and D. Widrig. 2003. Managing Software Requirements: A Use Case Approach. 2nd Edition. Addison-Wesley.

Lischner, R. 2000. Delphi in a Nutshell. O'Reilly and Associates, Inc. Rekayasa Perangkat Lunak A1

Luckey, T. and J. Phillips. 2006. Software Project Management for Dummies. Wiley Publishing, Inc.

McConnel, S. 2003. Professional Software Development: Shorter Schedules, Higher Quality Products, More Successful Projects, Enhanced Careers. Addison-Wesley.

Meyer, B. 2000. Object Oriented Software Construction. 2nd Edition. ISE, Inc.

Musciano, C. and B. Kennedy. 2002. HTML and XHTML: The Definition Guide. 4th Edition. O'Reilly and Associates, Inc.

Navarro, A. 2001. Effective Web Design. 2nd Edition. SYBEX, Inc.

Powell, G. 2006. Beginning Database Design. Wiley Publishing, Inc.

Riordan, R.M. 2005. Designing Effective Database Systems. Addison Wesley Professional.

Robbins, J. N. 2006. Web Design in a Nutshell, 3rd Edition. O'Reilly.

Suehring, S. 2002. MySQL Bible. Wiley Publishing, Inc.

Taylor, D.A. 1998. Object Technology: A Manager's Guide. Addison-Wesley.

Van Roy, P and S. Haridi. 2004. Concepts, Techniques, and Models of Computer Programming. The MIT Press.