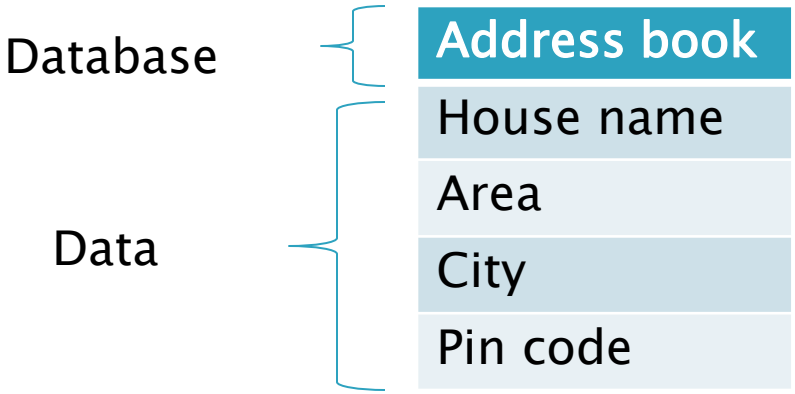
DBMS & RDBMS USING ORACLE

PREPARED BY :
P.V.Thummar

CHAPTER – 1
DBMS OVERVIEW, SQL & SQL * PLUS

INTRODUCTION OF DBMS

- DATA : “The name of any object is called data.”
- Database : “ A database is a collection of related & meaningful data.
- Example :



Introduction to DBMS

- DBMS : DBMS is a system (software) which is used to manage the database."
- For creating & using database in our computer effectively, we need a software which can insert, update, delete and process the data.
- To manage all the above tasks, we need a software which is called database management system.
- Examples : oracle, ms access, FoxPro, FoxBASE, Sybase, dbase 3+ etc.

BENEFITS OF DBMS

- WE can reduce duplication of data.
- Stored data can be shared to multiple users.
- We can enter only accurate data without error.
- Security of data can be maintain.

INTRODUCTION OF RDBMS

- Definition:
- RDBMS is a DBMS, which is used to set relationship between related tables.
- OR
- RDBMS is a DBMS, that is based on relational model, created by dr. E.F.Codd.

RDBMS

- RDBMS stores the data in the form of related tables. It means, if we have multiple tables, which are related to each other, we can get collected data from them.
- Opposite to it, we can spread a single table across several tables.
- Examples of RDBMS: oracle, MS SQL Server,
- Sybase SQL server, and IBM's DB2.

DBMS v/s RDBMS

	DBMS	RDBMS
1	Relationship between 2 tables are maintained programmatically.	Relationship between 2 tables can be specified at the time of table creation.
2	DBMS does not support client/server architecture.	RDBMS supports client/server architecture.
3	It does not support distributed database.	It supports distributed database.
4	In DBMS, there is no security of data.	It provides multiple levels of security 1. OS level 2. Command level 3. Object level.
5	Each table is given an extension.	Many tables are grouped into one database.
6	It satisfies less than 7 to 8 rules of dr. E.F.Codd.	It satisfies more than 7 to 8 rules of dr. E.F. Codd.

12 rules by Dr. E.F.Codd for RDBMS

Rule 1	Information rule
	All data should be in tabular format. All information(including metadata)
Rule 2	Guaranteed access rule
	All data should be accessible easily without any problem. The system should provide a guarantee, that the data stored in it will be accessible in the future. Table Name + Primary Key(Row) + Attribute (column).
Rule 3	Systematic treatment of null values.
	A field of a table should be allowed to remain empty, which is called null value. It means, if user don't want to enter any value in a field, it should allowed. Primary key must not be null

12 rules by Dr. E.F.Codd for RDBMS

Rule 4	Dynamic online catalog based on relational model
	The structure of a table can be accessed by the same tools (commands) that are used to access the table data. description of the complete Database
Rule 5	Comprehensive data sub language rule.
	The database must support one language that includes all functions for data creation, manipulation, data integrity and transaction commands. Most of RDBMS uses SQL (structure query language) as their supportive language. Example: SQL, etc.
Rule 6	View updating rule.
	If multiple users at different places work on the same database, and any user make changes in database, that changes should be displayed to all other users. updatable by the system as well.

12 rules by Dr. E.F.Codd for RDBMS

Rule 7	High – level insert, update and delete
	In relational database, many tables are related to each other. So, any operations like insert, update or delete applied in one table, should be automatically applied to all related tables. Set operation like Union, Intersection and minus should also be supported.
Rule 8	Physical data independence
	The user should be un aware about physical storage of database. There is no need for user to know that where the database is stored in hard disk. some file supporting table is renamed or moved from one disk to another,
Rule 9	Logical data independence
	When the structure of data is changed, the view of data should not be changed. join of the two tables. This rule is most difficult to satisfy.

12 rules by Dr. E.F.Codd for RDBMS

Rule 10	Integrity independence
	When user input the data, it must be checked that the inputted data are valid or not. Hence, only accurate data are allowed to store. Key and Check constraints, trigger etc, should be stored in Data Dictionary. This also make **RDBMS** independent of front-end.
Rule 11	Distribution independence
	The user should be un aware about database is distributed or not. A database should work properly regardless of its distribution across a network.
Rule 12	Non sub version rule
	As we know that, every dbms or rdbms support any one language. So, data & structure should be modified by same language. There is no requirement of subversion or any other language. system it should not be able to subvert or bypass integrity rules to change the data.

E-R Diagram

- This is one of the data model that was developed by P.P.chen in 1976.
- Entity relationship approach on the entity relationship model and is one of the best known approaches to the problem of DB design.
- It represents

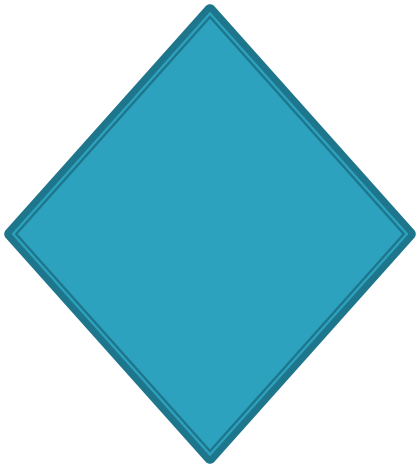
- 1.ENTITIES.
- 2.ATTRIBUTES.
- 3.RELATIONS.

Entities

- A person, thing concept about which you want to store data.
- Ex. student , courses.
- symbol :Entity that represent rectangle.

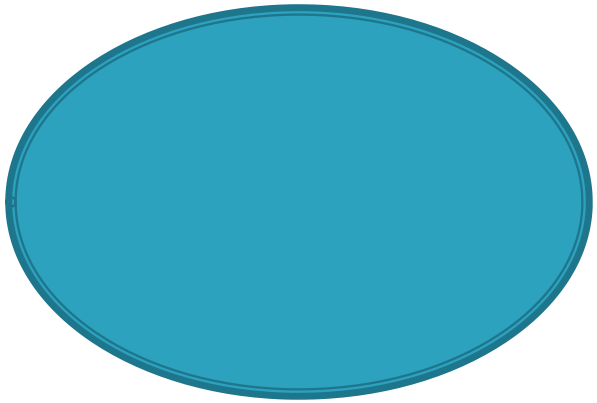
relation

- Used to represent a connection among related entities like a link.
- Ex. student -to- college , Client -to- server
- Symbol:



attributes

- Data values of an entities.
- Characteristics of person ,thing or concept.
- Ex. stud_id, stud_name, stud_class.
- Symbol:



First normal form (1NF)

Example: Suppose a company wants to store the names and contact details of its employees. It creates a table that looks like this:

emp_id	emp_name	emp_address	emp_mobile
101	Herschel	New Delhi	8912312390
102	Jon	Kanpur	8812121212 9900012222
103	Ron	Chennai	7778881212
104	Lester	Bangalore	9990000123 8123450987

emp_id	emp_name	emp_address	emp_mobile
101	Herschel	New Delhi	8912312390
102	Jon	Kanpur	8812121212
102	Jon	Kanpur	9900012222
103	Ron	Chennai	7778881212
104	Lester	Bangalore	9990000123
104	Lester	Bangalore	8123450987

Second normal form (2NF)

teacher_id	subject	teacher_age
111	Maths	38
111	Physics	38
222	Biology	38
333	Physics	40
333	Chemistry	40

teacher_details table:

teacher_id	teacher_age
111	38
222	38
333	40

teacher_subject table:

teacher_id	subject
111	Maths
111	Physics
222	Biology
333	Physics
333	Chemistry

Third Normal form (3NF)

Example: Suppose a company wants to store the complete address of each employee, they create a table named employee_details that looks like this:

emp_id	emp_name	emp_zip	emp_state	emp_city	emp_district
1001	John	282005	UP	Agra	Dayal Bagh
1002	Ajeet	222008	TN	Chennai	M-City
1006	Lora	282007	TN	Chennai	Urrapakkam
1101	Lilly	292008	UK	Pauri	Bhagwan
1201	Steve	222999	MP	Gwalior	Ratan

employee table:

emp_id	emp_name	emp_zip
1001	John	282005
1002	Ajeet	222008
1006	Lora	282007
1101	Lilly	292008
1201	Steve	222999

employee_zip table:

emp_zip	emp_state	emp_city	emp_district
282005	UP	Agra	Dayal Bagh
222008	TN	Chennai	M-City
282007	TN	Chennai	Urrapakkam
292008	UK	Pauri	Bhagwan
222999	MP	Gwalior	Ratan

Boyce Codd normal form (BCNF)

Example: Suppose there is a company wherein employees work in **more than one department.** They store the data like this:

emp_id	emp_nationality	emp_dept	dept_type	dept_no_of_emp
1001	Austrian	Production and planning	D001	200
1001	Austrian	stores	D001	250
1002	American	design and technical support	D134	100
1002	American	Purchasing department	D134	600

emp_nationality table:

emp_id	emp_nationality
1001	Austrian
1002	American

emp_dept table:

emp_dept	dept_type	dept_no_of_emp
Production and planning	D001	200
stores	D001	250
design and technical support	D134	100
Purchasing department	D134	600

emp_dept_mapping table:

emp_id	emp_dept
1001	Production and planning
1001	stores
1002	design and technical support
1002	Purchasing department

INTRODUCTION OF ORACLE

- In June 1970, Dr E.F.Codd worked for IBM, introduced a model sponsored by IBM, called "relational model".
- By using this relational model, it became possible to make relationship between different tables.
- In 1979, they establish a company called "Relational Software Incorporation", and released the first commercial version of this relational model called "SQL".
- After some time, the name of "Relational Software Incorporation" was changed to "ORACLE CORPORATION".

INTRODUCTION TO SQL

- The language developed by IBM to work on data of relational model, was originally called "Structure English Query Language(ESQUEL)".
- Later on, the word "English" was being removed and language was known as "Structure Query Language(SQL)".

INTRODUCTION OF SQL

- The SQL is generated by combining different small database languages like DDL,DML, DQL and DCL. These small languages contains the commands which are used to work on data.
- Development of SQL was governed by standards.
- In 1986, SQL was approved by ANSI standards.
- In 1987, SQL was approved by ISO standards.
- In 1992, new version is called "SQL-92".
- In 1999, new version is called "SQL-99" or SQL-3.

INTRODUCTION OF SQL

YEAR	EVENT
1970	Relational model (SQL) introduced by Dr. E.F.Codd in IBM.
1979	The company “Relational Software Incorporation” was established, and commercially released first version of SQL.
1986	Approved by ANSI
1987	Approved by ISO
1992	Standard revised and called SQL-92
1999	Standard revised and called SQL-99 or SQL-3.

Introduction to SQL * plus

- Sql *plus is an oracle tool (product) and submits the sql statements to the oracle server for execution.
- It provides command window where sql and sql * plus commands can be written by user.
- When we start sql *plus, it gives you a prompt window for entering user name, password and connection string (host name).
- After logging into oracle using sql*plus, sql commands can be executed.

SQL V/S SQL*PLUS

NO	SQL	SQL * PLUS
1	SQL is a language for communication with oracle to access the data	Sql *plus recognizes the sql statements and sends them to the server.
2	Sql is approved by ANSI	Sql *plus is a tool of oracle. So that, no standards is given.
3	Sql manipulates the data & structure of table.	Sql *plus does not manipulate the values of table.
4	Sql is entered one or more lines into sql buffer.	Sql *plus passes one line at a time, but not stored in sql buffer.
5	Sql does not have continuous character.	Sql *plus uses a dash(-) sign as a continuous character.
6	It can not be abbreviated.	It can be abbreviated.
7	Sql uses termination character	It does not use termination character.
8	Sql uses functions to format data.	It uses commands to format data.

Components of SQL or Commands of SQL

- SQL commands are divided into following 4 categories.
- (1) DDL (Data Definition language)
- (2) DML (Data manipulation language)
- (3) DCL (Data control language)
- (4) DQL (Data Query Language)

Components of SQL

- (1) DDL (Data definition language)
- It is a set of SQL commands used to create, modify and delete the database structure but not data.
- Ex. : create, alter, drop, truncate, rename.
- (2)DML (Data manipulation language)
- DML commands are used to work and manage the data of database.
- Ex. : insert, update, delete.

Components of SQL

- (3) DCL (Data control language)
- These commands are used to control the data access from user. The user can access the data according to the rights given to user.
- Ex. : grant, revoke, commit, rollback, savepoint.
- Commit, rollback & save point commands are also called TCL(Transaction control commands).

Components of SQL

(4) DQL (Data Query Language)

It is used to get the data from database and display on screen. It contains only one command “SELECT”, and it is the heart of SQL.

Operators in SQL

- An operator is a symbol which is used to perform operation on data and give result".
- There are 2 types of operators.
- (1) unary operators
- (2) binary operators
- (1)unary operators :
- The unary operators has only 1 operand.
- Ex. : +2
- (2) binary operators :
- The binary operators has 2 operands.

Operators in SQL

Arithmetic operators
+
-
*
/
Concatenation Operators
||
Relational operators
<
>
<=
>=
=
!= , <> , ^=

SQL basic data types.

- (1)CHAR : the CHAR data type is a fixes-length alphanumumeric character string.
  - Size can range from a minimum of 1 to a maximum of 2000 characters.
  - If the data is shorter then the defined length, it is space padded on the right.
- For ex. CHAR data type of 5 size. CHAR(5)

s	y	a	m	

Data types of SQL

- Char (size):
- This data type is used to store string values.
- The size in bracket shows the number of characters the cell can hold. Maximum size of this data type is 255.
- It is space padded data type.
- [2] Varchar (size) / varchar2 (size):
- This data type is also used to store string values where the size longer than char size.
- Maximum size of this data type is 4000 char.
- It is non- padded data type.

Count…..

- **VARCHAR**: A VARCHAR datatype is currenty synonymous with the VARCHAR2 datatype.
- **VARCHAR2 :The** VARCHAR2 datatype is a variable-length alphanumeric string having a maximum lenth of size bytes.
  - Only require the amount of space needs to store the data. Its columns can store up to 4000 byte.
- Ex.VARCHAR2 datatype of 5 size .VARCHAR2(5).

G	O	p	i	

Data types of SQL

- Date:
- This data type is used to store data & time. The default format is DD-MON-YY.
- It stores the time in 24 hour format. Default date for date field is first day of current month.
- Number (p,s):
- It is used to store numeric data. Valid values are zero, negative & positive with decimal point(.).
- In bracket, p shows precision which is maximum length of numeric data. S shows scale, which is number of digits to the right side of decimal point.
- Maximum length of p (precision) is 38 digits.

Data types of SQL

- Long :
- It is used to store binary data in ascii format. Maximum length of long data type is 2gb.
- This type of data can not be used by any commands of sql, and only one long value can be stored in one table.
- Raw / long raw:
- It is also used to store binary data such as picture or image.
- Maximum length of raw is 255 bytes and long raw is 2gb.

ASSIGNMENT –1 (chapter –1)

Q. NO.	QUESTIONS
1	Explain DBMS & RDBMS
2	Differentiate : DBMS v/s RDBMS
3	List out and explain Dr. E.F.Codd's rules for RDBMS
4	Write a short not on oracle.
5	Write a short note on SQL
6	What is SQL * plus? Explain difference between SQL & SQL * plus.
7	Explain components of SQL.
8	Explain data types of SQL.

Thank you
?