**SQL**

**What is data actually?**

* It is stored representation of OBJECTS and EVENTS that have meaning and importance in user’s environment.
* Data can be structure or unstructured.
  + Structured🡪EmpName, address.
  + Unstructured🡪EmpPhoto, AddrMap.

**Representation of data**

KING 10 5000 15-02-08

JONES 10 4500 02-03-06

BLAKE 20 3500 15-06-05

SCOTT 20 2000 12-05-02

FORD 30 1100 13-04-00

MILLER 10 1200 14-07-99

**What is Information Actually?**

**Information**

It is data that is processed form, such that it

increases the knowledge of the person who

uses the data.

**Representation of Information:**

|  |
| --- |
| **Employees Information**  **Organization: Date: 21-Aug 2012**  **J4A Techno Solutions** **Place: Nagpur** |
| Name Department Salary DOJ |
| KING 10 5000 15-02-08  JONES 10 4500 02-03-06  BLAKE 20 3500 15-06-05  SCOTT 20 2000 12-05-02  FORD 30 1100 13-04-00  MILLER 10 1200 14-07-99 |

**What is Metadata Actually?**

* It is the data which describes the properties or characteristics of end user’s data and the context of the data.
* Metadata properties can include information such as:

Data name (it is nothing, but user created objects name)

Definitions (it contains attributes and datatypes)

Length of size (it is specified as a size of the data)

Values allowed (it is used to accept a group of data into an attribute)

Source of data

Ownership

* Metadata and data are always separate.
* Metadata enables the database designers and programmers to understand exactly in which form the data should exist within the system.

**Metadata for employees Information**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Item Specification | | |  | |
| Field Name | Type | Length | Min | Max |
| Empname  Deptno  Empsal  Hiredate | Alpha  numeric  Integer Decimal  Date | 30  2  7 | 10  1500 | 90  50000 |

**What is database actually?**

Database is a platform where we can place the data for the future references. Database is a collection of interrelated data, i.e. database always stores data along with its relationships.

**Database Management system**

* Database management system is a software that is used to create, maintain and provide controlled access to user databases.
* Database management systems should provide systematic method of

🡪Creating the Database.

🡪Updating the Database.

🡪Storing the Database

🡪Retrieving of data from Database.

**Expected features of DBMS software**

* Enable end user and application programmers to share the data.
* Enable data shared among multiple application.
* Should not propagate and store data in new files for every new application.
* Should provide facility for….
  + Controlling data access.
  + Enforce data integrity.
  + Manage concurrency control.
  + Restoring the data in system failures.

**Database Management system evolution**

* First time introduced during 1960’s
* Relational model first defined by E.F Codd (IBM) in 1970.

**Objectives behind Evolution**

* Reduce the maintenance cost of software.
* Manage complex data types.
* Provider easier and faster access to data even for novice user.

**Different database Models and their Timelines**

* **Flat file DBMS (1960)**
* **Hierarchical DBMS (1970)**
* **Network DBMS (1980)**
* **Relational DBMS (1990)**
* **Object-Relational DBMS (2000)**

**Database as per oracle**

* As per oracle database is a collection of data in one or more number of files.
* The database is a collection of logical structures and physical structures which are integrated and configured for the integrity of the system.

**Logical Structures**

* In design state the system is represented in the form of Entity relationship model.
* In the database state it is representation of the actual metadata of the database management system software which will be in the form of table spaces, data dictionary objects etc.

**Physical structures**

* In this the system is represented in the form of tables, indexes etc.
* Database should have the ability to provide access to external tables for files outside the database, as if the rows in the files were rows in the table.
* Creating structures (Tables & Indexes)
* Within the oracle database, the basic structure is a table used to store data.

**Relation database properties**

* Should be accessed and modified by executing structured Query language (SQL) statement only.
* Should use set of operators?
* Need not specify the access route to the tables and data.
* There is no need to identify how the data is arranged physically.

**Key Components in ER-MODEL:**

* **ENTITY:** It is a thing of significance about which the information needs to be known.
* **ATTRIBUTES:**

It is something that describes or qualifies an entity.

**\*\***Each attribute may be mandatory or optional, but one attributes Mandatory.

**Row or Tuple:**

* It represents all data required for a particular instance in entity.
* Each row is an entity is uniquely identified by declaring it has PRIMARY KEY or UNIQUE.
* The order of the rows is not significant, while retrieving the data.

**Column or Attribute:**

* It represents one kind of data in a table.
* The column order is not significant when storing the data.

**A Field:**

* It can be found at the Intersection of row and a column.
* A field can have only one value, or may not have a value at all, the absence of value in Oracle is represented as NULL.

**Relating Multiple Tables:**

* Each table contains data that describes exactly only one entity.
* Data about different entities is stored in different tables.
* RDBMS enables the data in one table to be related to another table by using the Foreign keys.
* A Foreign Key is a column or a set of Column that refer to a Primary Key in the same table or another table.

**Rational Database Properties:**

* Should not specify the access route to the tables and should not reveal the physical arrange.
* The Database is accessed using Structured Query Language (SQL)
* The language is a collection of set operators.

**Communicating With RDBMS:**

* The Structured Query Language is used to Communicate with RDBMS.

**Generic features of structure Query Language**

* **It allows the user to communicate with server.**
* **It is highly efficient.**
* **It is easy to learn and use.**
* **It is similar to English language.**
* **It is platform independent and architecture independent.**