**19-Oct-2021**

Oracle: RDBMS Software PRODUCT, where used to store the data and perform operations on the DATA.

Oracle :

1. SQL
2. PLSQL

What is DATA?

It is a collection of Raw facts.

**Example**: 101 dinesh 2000

102 mahesh 3000

In above example, there no meaningful data such data is known as Raw facts.

What is INFORMATION?

**INFORMATION**: It is a collection of meaningful data or processed data.

Example: EmpID Ename Salary

101 dinesh 2000

102 mahesh 3000

What are different ways to store the data?

**DATA STORE:** It is a place where we can store data or information.

1) Books & Papers

2) Flat files

3) Database

**FLAT FILES**: This is a traditional mechanism which is used to store data or information in

individual unrelated files. These files are also called as Flat Files.

What all challenges if you store the data in flat files?

**Drawbacks of Flat files:**

1) Data Retrieval

2) Data Redundancy

3) Data Integrity

4) Data Security

5) Data Indexing

**20-Oct-2021**

1**) Data Retrieval:** If we want to retrieve data from flat files then we must develop

application program in high level languages, where as if we want to retrieve data from

databases then we are using Sequel Language.

SEQUEL (Structured English Query Language)

2) **Data Redundancy:** Sometimes we are maintaining multiple copies of the same data in

different locations this data is also called as Duplicate data or Redundant data. In Flat files

mechanism when we are modifying data in one location it is not effected in another

location. This is called INCONSISTENCY.

In databases, every transaction internally having 4 properties. These properties are known as

ACID properties.

**ACID Properties:**

A mean Atomicity (ROLLBACK)

C mean Consistency

I mean Isolation

D mean Durability (COMMIT)

These properties only automatically maintains consistent data in databases.

**3) Data Integrity:** Integrity means to maintain proper data. If we want to maintain proper

data then we are defining set of rules, these rules are also called as “ Business rules”. In

databases, we are maintaining proper data using „constraints‟, „triggers‟. If we want to

maintain proper data in flat files we must develop application programs in high level

languages like COBOL, JAVA, ETC…..

**4) Data Security:** Data stored in flat files cannot be secured because flat files doesn‟t

provides security mechanism. Whereas databases provides “ROLE based security”.

**5) Data Indexing:** If we want to retrieve data very quickly from database then we are using

indexing mechanism. Whereas flat files doesn‟t provide indexing mechanism.

To overcome all the above problems, a new software used by all organization to store data or

information in secondary storage devices. This is called DBMS software.

What all the advantages if you store the data in Oracle DB?

By using RDBMS we can achieve the above drawbacks.

**ACID Properties:**

**Flat File Scenario, How the data looks like**

|  |  |  |
| --- | --- | --- |
| Invoice info | Sales Info | Customer Info |
| Cust No  Customer Name  Invoice Details | Cust No  Customer Name  Sale Info | Cust No  Customer Name  Customer Address |

**Oracle DBMS scenario:**

|  |  |
| --- | --- |
| **Customer Info**  Cust No (Primary Key)  Customer Name  Customer Address  Customer Country  Customer City | **Invoice Info**  Cust No (Foreign Key in My Invocie)  Invoice Details |
| **Sale Info**  Cust No(Foreign Key in My Sale Info)  Sales Info |

Select

custno,

Customer name,

Invoice\_details,sales info

from

Customer\_info CI,

INVOICE\_info II,

SALE\_INfO SI

WHERE

CI.CUST NO=II.CUST NO

AND CI.CUSTNO=SI.CUSTNO;

SELECT CUSTNO AS “Customer No” FROM Customer\_info CI WHERE CI.CUSTNO=’001’ ;

SELECT \* FROM INVOICE\_info;

SELECT \* FROM SALE\_INFO;

ORACLE RDBMS : ORACLE is a Relational Database product which is used to store data permanently in secondary storage devices.

**21-Oct-2021**

**DATABASE:** It is an organized collection of interrelated data used by application program in

an organization. Once data stored in database it can be shared by number of users

simultaneously and also this data can be integrated.

**DBMS Architecture:** American National Standard Institute(ANSI) has established three level

architecture for database. This architecture is also called as “ansi/sparc” (Standard Planning

And Requirements Committee) architecture.

Main objective of DBMS architecture is to separate users view of the database from the where

physically it is stored.

This architecture mainly consists of three levels. They are:

1) External level

2) Conceptual level

3) Internal level



**DBMS Architecture or Three Level Architecture**

DBMS architecture provides “DATA INDEPENDENCE”.

**Data Independence:** Upper levels are unaffected by changes in the lower levels is called as

“Data Independence”. DBMS architecture have two types of Data Independences:

1) Logical Data Independence

2) Physical Data Independence

**1) Logical Data Independence:** Changes to the conceptual level do not required to change

to the external level this is called Logical Data Independence.

**Example**: Adding a new entity in conceptual level does not effect in external level.

**2) Physical Data Independence:** Changes to the internal level do not required to changes

in conceptual level. This is called Physical Data Independence.

**Example**: Adding an index to the internal level it is not affected in conceptual level.

**CONCEPTUAL LEVEL:** It describes logical representation of the database. Conceptual level

defines type of data storing in database and also defines what type of data does not store in

database using constraints and also specifies the relationship between data items.

NOTE: This level does not define how data is stored in database. In relational databases we

are defining conceptual levels through tables.

**EXTERNAL LEVEL:** This level describes end user view of the database. i.e., in this level

some group of users access only part of the database. In this level only we are defining the

view and those views given to the number of users.

**INTERNAL LEVEL:** Internal level describes how physically data is stored in database. This

level is handled by database administrator only. In relational databases indexes, cluster are

available in internal level.



Data Models:

1. **Hierarchical Data Model**

In this data model organizes data in tree like structure, we are representing data in parent child hierarchy. In this data model also data is represented in the format of records and also record type is also same as table in relational data model.

This Data model having more duplicate records because this data model is implemented

based on one- to – many relationships. That is why in this data model always child segments

are repeated. In this data model products, we are retrieved data very slowly because in this data

model products data base servers searching data based on root node onwards.

Root

|  |  |
| --- | --- |
| Node 1 | Node 2 |



1. **Network Model:**

In this data model is implemented based on many – to – many relationships. In this data model also data is stored in format of records. And also records type is also same as table in Relational Data model.



1. **Relational Model**

In this data model we are storing data in 2- dimensional tables. Relational data model mainly consist of 3 components.

1) Collection of Objects. (Tables)

2) Set of Operators. (<,>,=,<>)

3) Set of Integrity rules. (Constraints)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dept No | Dept Name |  | Emp | DeptNo | Joining Date |
| 10 | IT | 1001 | 10 | 01-Sep-2021 |
| 20 | HR | 1002 | 10 | 01-Sep-2021 |
| 30 | SE | 1003 | 20 | 01-Oct-2021 |
| Primary Key | | 1004 | 30 | 15-Oct-2021 |
| Index will be created automatically. | |  | Primary Key Foreign Key references to Dept (Dept NO) | | |

Select

EmpId, Deptno, Dept\_name, Joining\_date

From Employee E, Dept D

Where e.dept\_no=d.dept\_no and emp\_no=1001;

**ORACLE VERSIONS**

1)**Oracle 2.0** -> 1979

-> First public release

-> In this 2.0 only basic SQL functionality is there “joins”.

2) **Oracle 3.0** -> 1983

-> Rewritten in C language.

-> Commit, Roll back.

3) **Oracle 4.0** -> 1984

-> Read Consistancy

-> exp/imp utility programs [export/import].

4) **Oracle 5.0** -> 1985

-> Client server architecture.

5) **Oracle 6.0** -> 1988

-> Introduced PL/SQL

-> Row level locks.

6) **Oracle 7.0** -> 1992

-> Roles

-> Integrity constraints.

-> Stored Procedures

-> Stored Functions

-> Packages

-> Triggers

-> Data type “varchar” changed into “varchar2”

-> Truncate table.

7) **Oracle 7.1** -> 1994

-> Introduced dynamic SQL

-> ANSI/ISO SQL-92.

8) **Oracle 7.2** -> 1995

-> Inline views or sub queries used in from clause.

-> ref cursor (cursor variable)

9) **Oracle 7.3** -> 1996

-> Bit map indexes.

-> utl\_file package.

10) **Oracle 8.0** -> 1997

-> Object technology

-> Columns increased per a table upto “1000”.

-> nested table, varray.

-> Instead of triggers.

11) **Oracle 8i** (i- internet) -> 1999

-> Materialized views.

-> Function based indexes

-> Case conditional statements.

-> Analytical functions.

-> Autonomous trasactions

-> rollup, cube.

-> BULK BIND

12) **Oracle 9i** -> 2001

-> 9i joins or ansi joins

-> merge statements.

-> multi table insert

-> flash back queries.

-> Renaming a column

13) **Oracle 10g (g- grid)** -> 2003

-> recycle bin

-> flash back table

-> indices of clause

-> regular expressions

-> wm\_concat().

14) **Oracle 11g** -> 2007

-> Introduced continue statement in PL/SQL loops

-> Read only tables

-> Virtual Columns

-> Pivot() fuction.

-> Compound trigger

-> enable, disable clauses used in trigger specification

-> follow clause

-> Sequences used in PL/SQL without using dual table.

-> Named, mixed notations are used in a subprogram executed used select statement

**SQL (STRUCUTRED QUERY LANGUAGE):**

It’s a non-procedural language which is used to operate all relational database products.

**22-Oct-21**

**SQL Sub Languages in every database:**

**Data Definition Language (DDL)**

CREATE, ALTER, TRUNCATE, DROP (PURGE to remove permanently from BIN), RENAME

**Ex** : create table test(sno number(7));

**Data Manipulation Language(DML)**

INSERT, UPDATE, DELETE (IN (‘1008’,’1009’), = ‘1008’) rownum<11, MERGE

Ex : insert into test1 values(99.9);

Commit;

**Data Retrieval Language(DRL) or Data Query Language(DQL)**

SELETE

**Ex :** select \* from test1;

**Transactional Control Language (TCL)**

ROLLBACK, COMMIT and SAVEPOINT

**Ex :** insert into test1 values(101.5);

Rollback;

**Data Control Language(DCL)**

GRANT, REVOKE (Type of Grants : SELECT, ALL, INSERT, UPDATE, DELETE)

**Ex :** Grant select on test1 to user(username);

**Data Types:**

Data types specifies type of data within a table column Oracle have following data types.

1. **Number : Number values into the table (38)**
2. **Char : It allows text into table column (2000)**
3. **Date : it allow date value into the table column**
4. **Varchar2 : It allows text into table column (4000)**
5. **Long : It allow more than 4000 Chars (2GB)**
6. **LOBs : Large no.of Data, Images**

**23-Oct-21**

**Questions:**

1. **What is Table and what are rules applicable to create a table?**
2. **What is DDL and List out the DDL commands?**
3. **What is DML and List out the DML commands?**
4. **How to query or retrieve the data from Database?**
5. **What is the difference between Delete and Truncate?**
6. **What is DROP command and Usage of DROP command?**
7. **How to remove a table from Database permanently?**
8. **What is Merge statement and usage of Merge statement.?**
9. **What is the difference between DDL and DML commands?**

**Practice**

1. Create a Employee table with below columns.

|  |  |
| --- | --- |
| Column Name | Datatype |
| Empno | number |
| Ename | Varchar2 |
| Deptno | number |
| Job | Varchar2 |
| Sal | Number |
| Comm | Number |
| Manager | Number |
| Hiredate | Date |
| Inserted Date | Date (Assign Default value as sysdate) |
| Inserted By | Varchar2 (Assign Default value as current system user) |

Insert below data into Employee table.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **EMPNO** | **ENAME** | **JOB** | **MGR** | **HIREDATE** | **SAL** | **COMM** | **DEPTNO** | **Inserted Date** | **Inserted By** |
| 7839 | KING | PRESIDENT |  | 17-NOV-81 12.00.00 AM | 5000 |  | 10 | sysdate | system user |
| 7698 | BLAKE | MANAGER | 7839 | 01-MAY-81 12.00.00 AM | 2850 |  | 30 | sysdate | system user |
| 7782 | CLARK | MANAGER | 7839 | 09-JUN-81 12.00.00 AM | 2450 |  | 10 | sysdate | system user |
| 7566 | JONES | MANAGER | 7839 | 02-APR-81 12.00.00 AM | 2975 |  | 20 | sysdate | system user |
| 7788 | SCOTT | ANALYST | 7566 | 19-APR-87 12.00.00 AM | 3000 |  | 20 | sysdate | system user |
| 7902 | FORD | ANALYST | 7566 | 03-DEC-81 12.00.00 AM | 3000 |  | 20 | sysdate | system user |
| 7369 | SMITH | CLERK | 7902 | 17-DEC-80 12.00.00 AM | 800 |  | 20 | sysdate | system user |
| 7499 | ALLEN | SALESMAN | 7698 | 20-FEB-81 12.00.00 AM | 1600 | 300 | 30 | sysdate | system user |
| 7521 | WARD | SALESMAN | 7698 | 22-FEB-81 12.00.00 AM | 1250 | 500 | 30 | sysdate | system user |
| 7654 | MARTIN | SALESMAN | 7698 | 28-SEP-81 12.00.00 AM | 1250 | 1400 | 30 | sysdate | system user |
| 7844 | TURNER | SALESMAN | 7698 | 08-SEP-81 12.00.00 AM | 1500 | 0 | 30 | sysdate | system user |
| 7876 | ADAMS | CLERK | 7788 | 23-MAY-87 12.00.00 AM | 1100 |  | 20 | sysdate | system user |
| 7900 | JAMES | CLERK | 7698 | 03-DEC-81 12.00.00 AM | 950 |  | 30 | sysdate | system user |
| 7934 | MILLER | CLERK | 7782 | 23-JAN-82 12.00.00 AM | 1300 |  | 10 | sysdate | system user |

1. Create Dept table with below columns and values.

|  |  |  |
| --- | --- | --- |
| **DEPTNO** | **DNAME** | **LOC** |
| 10 | ACCOUNTING | NEW YORK |
| 20 | RESEARCH | DALLAS |
| 30 | SALES | CHICAGO |
| 40 | OPERATIONS | BOSTON |

1. Create a Salgrade table with Below columns and values.

|  |  |  |
| --- | --- | --- |
| **GRADE** | **LOSAL** | **HISAL** |
| 1 | 700 | 1200 |
| 2 | 1201 | 1400 |
| 3 | 1401 | 2000 |
| 4 | 2001 | 3000 |
| 5 | 3001 | 9999 |

**Queries to write**

1. Display the details of all employees.
2. Display the department information from DEPT table.
3. Display the name and job for all the employees
4. Display the name and salary for the all the employees
5. Display the name of all the employees who are working under dept no 10.
6. Display the names of all the employees who are working as clerks, salesman or analyst.
7. Display the names of the employees who are earning comm.
8. Display the names of the employees who do not earn comm.
9. Display sysdate or current date.
10. Display current user.