Experiment No - 1

* Aim -

Implement multivalued attribute, complex type and inheritance concept in ORDBMS.

* Objectives -

1) To learn multivalued attribute, complex types and inheritance concept in ORDBMS.

* Theory :

ORDBMS Definition :-

An object relational database is also called an object relational database management sm. this system simply puts an object oriented front end on relational database when application interface to this type of database. It will normally interface as through the data is stored as object However the system will convert the object information into data table with rows & columns & handle data the same as relational database.

* About oracle object and object types:

Oracle object types are user defined data type that makes it possible to model complex real world entities such as customer & Purchase order by universal entities "objects" in the db. oracle object technology is a layer of abstraction built on oracle relational technology. New object types can be created from any built in DB

type & any previously created object types, references & collection types.

* Basic Components of oracle objects :-

- Object relational elements :

Object relational functionality introduces a no of new concept & resources this are briefly described in following.

- Object type :

It's a kind of data type. you can use it in the type such as NUMBER or VARCHAR for eg you can specify object type as data type of column you can declare variable of an object type to contain a value of that object type.

① A set of object type does not come ready made with the DB. Instead you define object types you want
② object types are not useless they have just called as attributes & methods

* Type inheritence :

You can specialize an object type by creating subtypes that have some added differentiating features such as additional attributes or method subtypes & supertypes are related by inheritence

specialised versions of their parent, subtypes have all the parent attributes & methods plus any specialization that are defined in subtype itself.

* Implementation Procedure:

1) Create simple type person
2) Create complex type
3) Create table of type
4) Insert records into the table
5) Perform selection queries on table
6) Create type of varray
7) Create table with varray type
8) Insert record into table
9) Perform selection queries on table
10) Create subtype under supertype demonstrate inheritance concept.

Experiment No :- 2

* Aim :-

Implement, Partitioning on the tables.

* Objectives :-

1) To learn table Partitioning & its types.

* Theory :-

Now a days enterprises run databases of hundreads of Gigabytes in size. The database case known as very large Database (VLDB). from oracle ver 8.0 oracle has provided feathers of table partitioning you can partition table according to some criteria for eg. you have SALES table with following structure.

select sum(amt) from sales where year = 1991;
select Product, sum(amt) from sales where year = 1992 group by Product;

Now whenever you given queries like this oracle will search the whole table. If you partition this table according to years, the performance is improve since oracle will scan only a single partition instead of whole table.
Before creating partion tables create tablespaces & Datafiles. In oracle you can partition a table by

- Range Partitioning
- Hash Partitioning
- List Partitioning
- Composite Partitioning

Range Partitioning :

Range partitioning maps data to partitions based on ranges of partition key values that establish each partition.

List partitioning :

List partitioning enables you to explicitly control how rows map to partitions. You do this by specifying a list of discrete values for partitioning key in description for each partition.

Create Partition tables:

After Creating tablespace now. To create partition table given the following statement.

e.g:

Create table Sales(year number(4), Product varchar2(10), amt number(10,2)

Partition by range(year) (

Partition p1 values less than (1992) tablespace u1,

Partition p2 values less than (1993) tablespace u2

Experiment No - 5

* Aim :
To implement deadlock detection algorithm for distributed database using wait for graph to check for deadlock.

* Description :
There are 5 transactions T_1, T_2, T_3, T_4 & T_5 with
- T_1 initiated at site S1 & spawing at site S2
- T_2 initiated at site S3 & spawing at site S1
- T_3 initiated at site S1 & spawing at site S3
- T_4 initiated at site S2 & spawing at site S3
- T_5 initiated at site S3

* Codings :
Connected to
Oracle 9i release 9.0.1.1.1 - Production JServer
release 9.0.1.1.1 - Production.

SQL Create table ddl (trans varchar(20), loc varchar(10), wait varchar(10), site varchar(10));
Table Created.

SQL > insert into ddl values (T, X, Y, S)
1 row Created.

SQL > insert into ddl values (11, X_6, X_2, S_2)
1 row Created.

```
sql> insert into ddl values (t2, x5, x2)
     1 row created.
sql> insert into ddl values (t3, x2, x7, c1)
     1 row created.
sql> select * from ddl;
     8 rows selected

sql> ed ddl;
sql> set serveroutput on;
sql> @ddl
  42
  select teams, loc, wait.

DDl
 declare
  cursor c1 is
 select teams, loc, wait from ddl.
Type c-list is varray(20) of ddl.loc%type;
 vc-list = c-list();
 tc-list = clist();
 c.in integer = 0;
 d integer = 0;
begin
 open c1
dbms-output.put-line('Teams' || '||' loc'||' 'wait')
 loop
 fetch into ss;
```

dbms_output.put_line(ss.teams '||' '||' ss.loc||')
c := c+1;
l1.extend;
l1(c) := ss.loc;
d := d+1;
l2.extend;
l2(d) := sq.coexit;
end loop;

DD3 :=
declare
cursor c1 is
select teams, loc, coexit
from dd1
where site = 'S2'
type c_list is varray(10) of dd1.loc%.type;
t c_list : c_list()
c integer := 0;
begin
open c1;
dbms_output.put_line('Teams ||' '||' loc '||')
loop
fetch c1 into ss;
(d) = ss.coexit; end loop;

Result:
Thus the deadlock detection algorithm distributed database has been developed successfully.

Experiment No - 4

* Aim :
Implement database security using grant & revoke operation.

* Objectives :
1) To learn database security Concept.

* Theory :
Database security concerns the use of broad range of information security controls to protect database against compromises of their confidentiality, integrity & availability. It involves various types or categories of controls such as technical, procedural / administrative & physical.

- The account level : At this level, the DBA specifies the particular privileges that each account holds independently of relations in database.

- The selection or table level : At this level DBA can control privilege to access each individual selection or view in database.

* In SQL the following types of privileges can be granted on each individual section R.

- SELECT Privilege on R: Gives account retrieval Privilege. In SQL this gives account Privilege to use SELECT stmt to retrieve tuples from R.

- Modify Privileges on R: this gives the account the capability to modify tuples of R.

- Divided into UPDATE, DELETE, & INSERT Privilege to apply the corresponding SQL Command to R.

- REFERENCES Privilege on R: this gives the account the capability to reference relation R when specifying integrity constraints. the Privilledge can also be restricted to specific attributes of R.

- Create view, the account must have SELECT Privilledge on all relations involved in view definition.

* DCL Commands are used to enforce database security in multiple users database environment. Two types of DCL Commands are Grant & revoke. only database Administrator's or owner's of database object can Provide / remove Privileges on database object.

The Syntax for GRANT Command is

GRANT Privilege_name
ON object_name
TO {user_name | PUBLIC | role_name} [with Grant opt];

* The Revoke Command removes user access rights or Privileges to database objects

Syntax for REVOKE:

REVOKE Privilege_name
ON object_name
FROM {user_name | PUBLIC | role_name}

* Implementation Procedure:

1) Create a user account using System user.
2) Give the account level Privileges to created users.
3) Using users account, Create tables.
4) Give relation level Privilege on tables.
5) Perform all Grant & Revoke operation.

Experiment No - 5

* Aim :-
  Create star schema data model in SQL Server using microsoft toolset.

* Objectives :- To learn star schema data model in SQL Server using the microsoft toolset.

* Theory :- To practice creating a star schema data model from scratch we first reviewed some data model concept & that the SQL server management studio (ssms) has capacity for data modeling.

1) Conceptual Data model :
   It is used to describe entities & their relationship High level business user such as executive managers can comprehend the model diagram.

step 4: Create a dimension table in SSMS.
step 5: Save the new diagram.
step 6: Create all dimension tables.
step 7: Create a fact table.
step 8: Establish relationship between dimension table and fact table.

2) Logical data model:
It specifies all attributes for each entity and identifying relationships among these entities the logical data model varies from team different types of database management system.
1) Open database System.
2) Specify all attributes for one entity.
3) Specify all attributes for every entity.
4) Present the logical data model.
5) Produce a logical data model.

3) Physical data model:
It is created for specific DBS e.g oracle microsoft, SQL servers.
step 1 - Drop all foreign keys.
2 - Import into data Project
3 - Review table structures.
4 - Create indexes on fact tables
5 - Configures database System.
6 - check in the database Project.

Experiment No - 6.

* Aim :-
Implementation of OLAP queries.

* Objectives -
1. To learn about on line Analytical Processing.

* Theory :
OLAP :
Online Analytical Processing or OLAP is an approach to quickly providing answers to analytical queries that are multidimensional in nature. OLAP is part of the broader category business intelligence, which also includes Extract transform load (ETL), relational reporting and data mining. The typical applications of OLAP are in business reporting for sales, marketing, management reporting, business process management. budgeting and forecasting, financial reporting and similar areas.

Types :
OLAP systems have been traditionally categorized using the following taxonomy.

Multidimensional :
MOLAP is the 'classic' form of OLAP and is sometimes referred to as just OLAP MOLAP uses database structures that are generally optimal for

attributes such as time Period, location, Product or Account Code. the way that each dimension will be aggregated is defined in advance by one or more hierarchies.

Relational :-

ROLAP works directly with relational databases. the base data and the dimension tables are stored as relational tables and new tables are created to hold the aggregated information. Depends on a specialized schema design.

Hybrid : there is no clear agreement across the industry as to what constitutes "Hybrid OLAP". except that a database will divide data bet" relational and specialized storage.

Implementation Procedure :

1. Create the table Product, location, time and sales with attributes using Oracle Database.
2. Insert records into created tables.
3. Implement following queries.
   a) find the total sales.
   b) find the total sales for each city
   c) find the total sales for each state.
   d) find the top five Products ranked by total sales
   e) find the total sales by month.
   f) find the total sales by month for each city.
   g) find the Per. change in total monthly sales for each Product.

Experiment No - 7

* Aim -
Implementation of SSAS Server.

* Objective -
Cube in SSAS it also helps to analyze data in multidimensional format for smarter business decisions.

* Theory :-
Create Data Sources :-
- Select data sources in solution employee & right click on data sources - new data sources click next now
→ enter server name when sql is get installed
→ choose database name → ok.

- Now set Connection string created in left side of data source wizard → next → use → service account → next → use service account → next → enter dataSource name → select data source name view in solution employee. right click on data.

Create new named Calculation
select & right click climate to create new named calculation → click ok

To Create these dimension like date, Product & Customer →

select to Create new dimension → next → use an listing table → next → select "dimDate" → select → "DateKe" as key Column & Date A key as name column → next → select Column name which will appear in Cube dimension.

Type a name of new dimension & click to finish to Create the date dimension.

for Customer & Product dimension we have mentioned steps.

Create Hierarchies.

1) Create attribute relationship.

2) view data in browse.

3) Create cube.

4) Cube deployment.

Experiment No - 8

* Aim -
  Implement a Apriori algorithm in data

* Objectives -
  To learn about a Apriori algorithm.

* Theory :-

Data mining :

In Computer science and data mining, A Apriori is classic algorithm for learning association rule. A Apriori is designed to operate on database containing transactions other algorithms are designed for finding the association rules in data have no transactions

A Apriori uses BFS and hash tree structure to count candidate item sets efficiently structure to count D generates candidate item sets of length k from sub patterns. According to download closure lemma. the candidate set containing all frequent k length items. After that it scans the transactions database to determine frequent item set among candidate for determining frequent items quickly the algorithm uses a hash tree to store candidate item sets this hash tree has item sets at the leaves & hash tables at internal nodes.

This hash tree has item sets at the leaves and hash tables at internal nodes note that this is to not the same kind of hash tree used in tree instance P2P sim.

Key Concept

Data mining, Decision tree, C4 Point algorithm.

Experiment No - 9

* Aim :-
  Implement k-means clustering algorithm.

* Objectives -
  1) To learn clustering process.

* Theory :-
  Partition a set of records into groups such that all records within group are similar to each other & records that belong to two diff. groups are dissimilar. Similarity b/n records measured computationally by distance function the output of clustering algorithm consist of summarized representation of each cluster.
  e.g. if spherical clusters : Center C (mean) & radius R given collection of records $e_1, e_2 \dots e_n$
  $C = \Sigma e_i$, $R = \Sigma (e_i - C) / n$

* k-means clustering :
  In simple words, it is algorithm to classify or to group your objects based on attributes / features into k number of groups. k is positive integer number. The grouping is done by minimizing sum of squares of distance b/n data & corresponding cluster centroid.

K- means clustering algorithm steps :-

step 1 :- Begin with decesion on value of K= number of clusters.

step 2 :- Put any initial partition that classifies data into k clusters, You may assign training samples randomly or systematically as follows.

1) Take first k training samples as single element cluster
2) Assign each of remaining (N-k) training sample to the cluster with nearest centroid.

step 3: Take each sample in sequence & Compute its distance from centroid of each of clusters. If sample is not currently in cluster with closest centroid switch this sample to that cluster & update centroid of cluster gaining new sample & cluster lossing sample.

Experiment No -10

* Aim:
  Implement decision tree for data mining Problem

* objective:
  To learn decision tree.

* Theory:
  A decesion tree is geaphical representation of collection of classification rules Given a data record tree directs the record from the root to a leaf internal nodes: labeled with Predictor attribute outgoing edges: labelled with Predicates that involves splitting attribute of node.
  leaf nodes: labeled with value of dependent attribute.

Tree Constructed into two Phases:
Phase 1: growth phase
  Construct very large tree
Phase 2: Pruning phase.

Build the tree greedily top down
At the root node examine database & Compute locally best splitting Criteria.
Partition database into two parts.
Recurse on each child.

Implementation steps:

1) start
2) Create table InsuranceInfo (customer_name, age, caretype)
3) Insert records into Created table.
4) Do database Connectivity

Experiment No - 11

* Aim :-
Install & deploy mongoDB Cloud database on windows platform write application to demonstrate the Crud operation with mongoDB backend cloud databases

* Objectives :
1) To learn installation of mongoDB Cloud database
2) To Perform CRUD operation on MongoDB

* Theory :
MongoDB is document oriented NoSQL database used for high volume data storage. MongoDB is database which come into light around the mid 2000s.

* Collection :
Collection is a group of mongoDB documents it is equivalent of RDBMS table. A collection lists within a single database Collection do not enforce schema.

* Document :-
A document is set of key value pair Document have dynamic schema. Dynamic schema means that documents in same Collection do to create 3 dimension like data, Product & customer →

select to Create new dimension → next → use listing table → next → select dimdate → select 'Datekey' as key Column & DateAltKey as name Column → next → select Column name which will appear in cube dimension.

For Customer & Product dimension we above mentioned steps.

1) Create attribute relationship
2) new data in browser
3) Create Cube
4) Cube deployment.