Object Databases

Object Relational Data Model

Oracle Objects

- Oracle objects are user-defined types that makes it possible to model real world entities as object
- Oracle object technology is layer of abstraction built on top of oracle relational technology.
- "**object**" data type as other primitive data type
  - Attributes (primitive data type)
  - Methods (stored procedure / function)

Oracle Objects : syntax

create or replace type <typename> **as object** (

datamember <data type>,

datamember <data type>, ….

map member function <function name> **return** <data type>

) **not final**;

not final : to indicate, this can be Base Type from which any class can be derived/inherited.

create or replace type body <typename> **as member function** <function-name> **return** <data type>

is

begin

…..

end

end

Oracle Objects : syntax

create table <tablename> **of** <typename>;

insert into <tablename> **values (** <typename>**(**‘ ‘, ‘ ‘**));**

select value(a**) from** <tablename> a **where** **a.**<fieldname1> any condition

select a.<fieldname1>,**a.**<fieldname2>….. **from** <tablename> **where** <anycondition>

Example

```
create type person_type as object(
        idno number,
        name varchar2(30),
        phone varchar2(30),
 map member function get_idno return number)
                                        not final;
create type body person_type as
   map member function get_idno return number is
   begin
            return idno;
   end;
end;
```

Example

```
create table contacts(
                        contact person_type,
                        contact_date date);
```

```
insert into contacts values
(person_type(101,’B.Raju’,’64-1,HYD’),’7-Jan-2008’);
```

```
select contact.get_idno() from contacts;
```

True Object Table

Special kind of table in which all rows contains objects

```
create table person_obj_table of person_type;

Insert into person_obj_table values(102, ‘A Sani’,’Alp K’);

Select value(p) from person_obj_table p where p.name=‘xyz’;
```

Exercise

Create object table “student” containing field “name” of size 50 characters with member function retWordCount()

retWordCount() function will count the no. of words in field “name”

select name from student where retWordCount()<=10;

Object Identity & Reference Types

REF datatype

- *Each* row object is identified by unique, non-changeable identifier : **OID**
- Syntax :

  <variable name> **REF** <objectTypename>
- Example :

```
declare
      x ref objectName;       pointer
      y objectName;           actual object variable
begin
    select ref(p) into x from objectName p; first extract address of object i.e. pointer
    select deref(x) into y from dual;           ref/get the object from that address pointer
    dbms_output.put_line(y.<fieldname>);
end;
```

Arrays & Multiset types in SQL

SQL support two collection types :

- **arrays** : ordered elements
- **multisets** : un-ordered collection elements may occur multiple times

```
create type Publisher as (name varchar2(20),
                          branch varchar2(20))

  create type Book as
  (title varchar2(20),
  author_array varchar2(20) array[10],
  pub_date date,
  publisher Publisher,
  Keyword_Set varchar2(20) multiset)
```

Arrays & Multiset types in SQL

create table books of book;

```
insert into books values('DBMS',
        array['korth','sudarshan'],'08-Jan-2019',
        new Publisher('MGH','NEW YORK'),
        multiset['database','design']);
```

```
select title from books where 'database' in
(unnest(keyword_set))
```

```
select author_array[1],author_array[2] from books
where title='DBMS';
```