Date 6-10-22
Page 8

DBMS

first lab DBMS

-> create table Student
(
Roll_no. number(2);
Name varchar(20),
E-mail varchar(15));

-> Drop table table_name;
-> Truncate table table_name;

* Rename a table name:

(1) ALTER TABLE table_name Rename to new_tablename

* Rename the column of table:

(2) ALTER TABLE table_name Rename column old_name to new_name;

* Add column.

(3) ALTER TABLE table_name Add Datatype;

* Drop column.

(4) ALTER TABLE table_name Drop column column_name;

* Modify

(5) ALTER TABLE table_name modify Datatype;

Date 7-10-22
Page

☆ count(), AVG(), and sum().

(1) Select count (column_name) from table_name where condition;
e.x: Select count (Product_Id) from Products;

(2) Select Avg (Price) from Products;

(3) Select sum (Quantity) from Products;

(4) Select Distinct column_name from table_name;

(5) Update table_name set column name = 'value' where Id = 5;

(6) Delete from table_name where Id = 7;

(7) ALTER TABLE table_name ADD column_name;

8-10-22

(1) Top():
Select Top 3 * from table_name;
Select * from table_name where rownum <= 3;

(2) min():
Select min (column_name) from table_name;

(3) max():
Select max (column_name) from table_name;

(4) like:

Select * from table_name where column_name like '%y%';

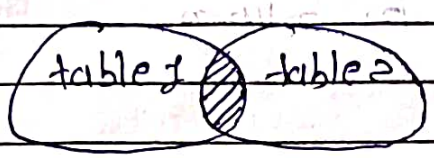
(5) In:

Select * from table_name where column_name in (value1, value2);

(6) Between:

Select * from table_name where column_name between value1 and value2;

* (7) ~~inner join~~: Inner join:

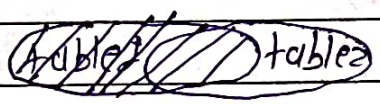


~~Syntax~~ Syntax:

Select columns from table1 inner join table2 on table1.column = table2.column.

ex.: select order.order_Id, customer.c_name from order inner join customer on order.c_Id = customer.c_Id;

* (2) left join:



Ex: Select customer.C_name, order.c_id from customer left join order on customer.c_id = order.order_id;

2-11-22

★ constraint:

(1) Default: It is used to set a default value for a column.

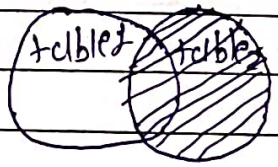
(2) Not null: Value has not null. You need to enter same value in column.

(3) Primary key

(4) foreign key

★ create table student
(
id int Not null,
name varchar(20),
city varchar(25) Default 'Ahmedabad');

★ Right Join:



Return all records from right table and matching records from the left table.

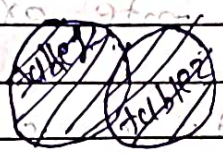
Select columns from table1 Right Join table2 on table1. column = table2. column.

Product			order		
P_Id	Name	Price	O_Id	P_Id	O_number

★ Full outer Join :

Returns all records from both left and right table.

→ Select columns from table1 full outer Join table2 on table1. column = table2 column ;



Product			order		
P_Id	name	Price	O_Id	P_Id	O_number

★ PL /SQL

PL - Procedural language / structured query language

→ PL/SQL is a Block structured language.
→ The Program of PL/SQL are logical Blocks that can contant any number of nested Sublock.
→ PL/SQL includes like condition and loops it allow declaration of constant and variable. It can also support oracly.

Date
Page

⇒ PL/SQL is nam for is combination of data manipulation manipulation power of SQL with data processing power of procedural language.

★ ⇒ how to declare a variable in PL/SQL :-

Variable_name ~~datatype~~ datatype.
number = 5;

datatype name :- number, char, string, Date/time, int, varchar, double, float, Bullion.

★ Rules of PL/SQL variable :-

(1) The variable name should not excessed 30 character.

(2) variable name should note be a same as ~~of table~~ the table, tables column of that blocks.

(3) The PL/SQL is note case sensitive so it could be either low case or upper case.

Ex :- V_data.
V_DATA.

(4) You should make your variable easy to read and understand after the first character you can underscore sign.

Procedure in PL/SQL :

PL/SQL has two types of subprograms called Procedure and Function. Generally You use a Procedure to Perform an action and Function to compute a value. The main difference between Procedure and Function is a Function must always return a value. and on the other hand a Procedure may or may not return a value.

Example - Procedure :

```
create or Replace Procedure greetings
AS
Begin
dbms_output.put_line("Hello world");
END;
        // Procedure created.
```

Execute greeting :

```
=> Hello world.
    drop Procedure greetings; // delete Procedure.

Declare
        a number;
Procedure addreNum(a in out number) is
Begin
    a:=a*a;
END;
Begin
        a := 23;
addreNum(a);
```

Date ____
Page ____

dbms_output.put_line ("Square of (a):" || a);
END;

* create function function_name (parameter)
Return Return_datatype
Begin
<function body>
End [function_name];

* create function add (n1 in number, n2 in number)
return number
is
n3 number;
BEGIN
n3 := n1 + n2;
return n3;
END;

* PL / SQL operators:-

Arithmetic = +, -, *, /
Relational = =, !=, >, <, ≥, ≤
Comparison = like, Between, IN
logical = and, or, not

* PL/SQL - constants / and literals :-

A constant holds a value that declared, does not change in the program
ex:- PI = 3.141592654
PI constant number = 3.14

Date
Page

A literals:

numeric = 05 078 -140 +32767
character = 'A', '>', '%', '?'
string = "Hello world", '19-Nov-12'
Boolean = True, false.
Date and time.