**PL/SQL**

**BASIC STRUCTURE**

-- DECLARE

-- message varchar2(100):= 'Hello, World!';

-- BEGIN

-- dbms\_output.put\_line('WELCOME '||message);

-- END;

-- DECLARE

-- NAME varchar2(20):='CHANDRA';

-- BEGIN

-- dbms\_output.put\_line('WELCOME '||NAME);

-- END;

**EMPLOYEE PROBLEM**

DECLARE

EID number(5):=101;

ENAME varchar2(20):='CHANDRA';

ESALARY number(20) default 5000;

BEGIN

dbms\_output.put\_line('EMPLOYEE ID: '||EID);

dbms\_output.put\_line('EMPLOYEE NAME: '||ENAME);

dbms\_output.put\_line('EMPLOYEE SALARY: '||ESALARY);

END;

**INPUT TAKING FROM USER IN PL-SQL**

DECLARE

N1 number(5);

N2 number(5);

BEGIN

dbms\_output.put\_line('ENTER THE VALUE FOR');

N1 := &N1;

dbms\_output.put\_line('ENTER THE VALUE FOR NUM2');

N2 := &N2;

dbms\_output.put\_line('ADDITION: '||(N1+N2));

dbms\_output.put\_line('SUBSTRACTION: '||(N1-N2));

dbms\_output.put\_line('MULTIPLICATION: '||(N1\*N2));

dbms\_output.put\_line('DIVISION: '||(N2/N1));

END;

**Fetching data type and values from another table and using**

create table tbl\_employee(eid number(10),ename varchar2(20),esalary number(10));

insert into tbl\_employee values(101,'lakshmipathi',12345);

insert into tbl\_employee values(102,'shiva',16745);

insert into tbl\_employee values(103,'prasad',40000);

insert into tbl\_employee values(104,'omsai',5000);

select \*from tbl\_employee;

-- DECLARE

-- name tbl\_employee.ename%type;

-- salary tbl\_employee.esalary%type;

-- BEGIN

-- select ename,esalary into name,salary from tbl\_employee where eid=103;

-- dbms\_output.put\_line('employee name : '||name);

-- dbms\_output.put\_line('employee salary : '||salary);

-- end;

DECLARE

name tbl\_employee.ename%type;

salary tbl\_employee.esalary%type;

BEGIN

select ename,esalary into name,salary from tbl\_employee where eid=&eid;

dbms\_output.put\_line('employee name : '||name);

dbms\_output.put\_line('employee salary : '||salary);

end;

**NESTED BLOCKS**

DECLARE

outer\_variable varchar(20):='global variable';

begin

DECLARE

inner\_variable varchar(20):='local variable';

begin

dbms\_output.put\_line(outer\_variable);

dbms\_output.put\_line(inner\_variable);

end;

dbms\_output.put\_line(outer\_variable);

end;

**Conditional Statements**

**If-else**

Here we can use “elsif” in pl/sql;

**DECLARE**

**num1 number(10);**

**num2 number(20);**

**num3 number(10);**

**begin**

**dbms\_output.put\_line('enter the num1:');**

**num1:=&num1;**

**dbms\_output.put\_line('enter the num2: ');**

**num2:=&num2;**

**dbms\_output.put\_line('enter the num3: ');**

**num3:=&num3;**

**if num1>num2 and num1>num3 then**

**dbms\_output.put\_line('A is Big');**

**elsif num2>num1 and num2>num3 then**

**dbms\_output.put\_line('B is Big');**

**else**

**dbms\_output.put\_line('C is Big')**

**end if;**

**end;**

**Switch Condition**

declare

color char(1);

begin

dbms\_output.put\_line('enter the color :');

color:=&color;

case color

when 'r' then

dbms\_output.put\_line('red');

when 'g' then

dbms\_output.put\_line('green');

when 'b' then

dbms\_output.put\_line('blue');

else

dbms\_output.put\_line('unknown color');

end case;

end;

**LOOPING CONDITIONS**

**SIMPLE LOOP**

declare

sum1 number(5) default 0;

x number(3) default 0;

begin

loop

sum1:=x+sum1;

x:=x+1;

exit when x=11;

end loop;

dbms\_output.put\_line(sum1);

end;

**WHILE LOOP**

declare

sum1 number(5) default 0;

x number(3) default 0;

begin

WHILE x<=11

LOOP

sum1:=x+sum1;

x:=x+1;

end loop;

dbms\_output.put\_line(sum1);

end;

**FOR LOOP**

declare

sum1 number(5) default 0;

x number(3) default 0;

begin

FOR x in 1..11

LOOP

sum1:=x+sum1;

end loop;

dbms\_output.put\_line(sum1);

end;

**RECORDS**

create table tbl\_employee(eid number(10),ename varchar2(20),esalary number(10));

insert into tbl\_employee values(101,'lakshmipathi',12345);

insert into tbl\_employee values(102,'shiva',16745);

insert into tbl\_employee values(103,'prasad',40000);

insert into tbl\_employee values(104,'omsai',5000);

select \*from tbl\_employee;

declare

my\_record tbl\_employee%rowtype;

begin

select \* into my\_record from tbl\_employee where eid=101;

dbms\_output.put\_line('employeee id :'||my\_record.eid);

dbms\_output.put\_line('employeee id :'||my\_record.ename);

dbms\_output.put\_line('employeee id :'||my\_record.esalary); end;

**PROCEDURES**

create table tbl\_employee(eid number(10),ename varchar2(20),esalary number(10));

insert into tbl\_employee values(101,'lakshmipathi',12345);

insert into tbl\_employee values(102,'shiva',16745);

insert into tbl\_employee values(103,'prasad',40000);

insert into tbl\_employee values(104,'omsai',5000);

select \*from tbl\_employee;

create or replace procedure Myproce is

name varchar(20);

begin

select ename into name from tbl\_employee where eid=101;

dbms\_output.put\_line('employee name; '||name);

end;

/

exec Myproce;

**Addition Procedure with out return value**

create or replace procedure Myproce(a number,b number) is

add number(5);

begin

add:=a+b;

dbms\_output.put\_line('employee name; '||add);

end;

/ exec Myproce(10,20987);

**Default procedure**

Here IN parameter will acts as a constant

**In parameter**

It is constant parameter we can initialize inside the procedure we cannot pass the value from the out side the function

create or replace procedure Myproce(v\_eno IN number) is

begin

v\_eno=v\_eno+10;

dbms\_output.put\_line(v\_eno);

end;

/

exec Myproce(20);

**Out parameter**

For out parameter we need to initialize inside the function Default parameter is IN

create or replace procedure Myproce(v\_eno out number) is

begin

v\_eno:=10;

dbms\_output.put\_line(v\_eno);

end;

/

declare

no number(3):=20;

begin

Myproce(no);

end;

**Function with return type**

create or replace function Myproce(a number,b number) return number is

add number(5);

begin

add := a + b;

return add;

end;

/

declare

res number;

begin

res := Myproce(10,20);

dbms\_output.put\_line('addition '||res);

end;

**CURSOR**

declare

id tbl\_employee.eid%type;

name tbl\_employee.ename%type;

salary tbl\_employee.esalary%type;

cursor c\_employee is

select eid,ename,esalary from tbl\_employee;

begin

open c\_employee;

loop

fetch c\_employee into id,name,salary;

exit when c\_employee%notfound;

dbms\_output.put\_line(id ||' '||name||' '||salary);

end loop;

close c\_employee;

end;

/

**EXCEPTION HANDLING**

declare

id tbl\_employee.eid%type:=0;

name tbl\_employee.ename%type;

salary tbl\_employee.esalary%type;

BEGIN

SELECT ename,esalary into name,salary from tbl\_employee where eid=id;

dbms\_output.put\_line('name: '||name);

dbms\_output.put\_line('salary '||salary);

exception

when no\_data\_found then

dbms\_output.put\_line('invalid employee data');

when others then

dbms\_output.put\_line('error!');

end;

/

create or replace procedure excep(v\_age number) is

AGE\_ERROR exception;

BEGIN

if v\_age<=0 then

raise AGE\_ERROR;

else

dbms\_output.put\_line('you are '||v\_age);

end if;

exception

when AGE\_ERROR then

dbms\_output.put\_line('age is must be more than 0');

end;

/

begin

excep(10);

end;

**TRIGGERS**

create or REPLACE trigger my\_trigger

after delete or insert or update on tbl\_employee

for each row

BEGIN

DBMS\_OUTPUT.PUT\_LINE('employee table has been modified');

end;

/

delete from tbl\_employee where eid=101;

insert into tbl\_employee values(136,'charan',89756);