---------------------------------------DBMS-------------------------------------------------

Experiment – 1.1

-----------------------------------------------------------------------------------------------

Create table employee

(

eid int,

name varchar2(20),

experience number(10),

Salary number(10)

);

INSERT INTO employee values ('1','Harsh','5','4000');

INSERT INTO employee values ('2','Rohan','4','5000');

INSERT INTO employee values ('3','Aditya','2','6000');

ALTER TABLE employee add address varchar(10);

ALTER TABLE employee drop column experience;

ALTER TABLE employee modify address number(10);

ALTER TABLE employee rename column eid to Sr\_no;

SELECT \* FROM employee;

desc employee;

drop table employee;

Experiment – 1.2

CREATE TABLE STUDENT(

SID numeric(10),

Sname varchar(50),

course varchar(50),

fee numeric(20));

select \* from STUDENT;

insert into STUDENT values(1,'rohan','java',1233);

insert into STUDENT values(2,',aditya','python',3422);

insert into STUDENT values(3,'ruhela','network',6868);

insert into STUDENT values(4,'mrinal','java',3453);

insert into STUDENT values(5,'barnwal','english',1987);

insert into STUDENT values(6,'harsh','dbms',5498);

insert into STUDENT values(7,'sameer','java',7833);

insert into STUDENT values(8,'agarwal','java',1233);

insert into STUDENT values(9,'yash','python',2323);

insert into STUDENT values(10,'vasu','network',1233);

describe STUDENT;

Select \* from STUDENT;

SELECT SID,course from student;

select \* from student where fee=1987;

select SID,Sname AS "STUDENT", COURSE from STUDENT;

SELECT SID,Sname,fee from STUDENT where fee between 1500 and 3000;

select \* from student where COURSE ='java' or course='dbms';

SELECT \* FROM STUDENT where COURSE in ('java','dbms');

SELECT \* FROM STUDENT where COURSE not in ('java','dbms');

SELECT sum(fee) FROM STUDENT ;

SELECT min(fee) FROM STUDENT ;

SELECT max(fee) FROM STUDENT ;

SELECT avg(fee) FROM STUDENT ;

SELECT count(\*)FROM STUDENT ;

SELECT \* FROM STUDENT order by fee;

SELECT \* FROM STUDENT order by fee desc;

SELECT \* FROM STUDENT order by COURSE,fee desc;

SELECT course,count(SID)FROM STUDENT group by course;

SELECT Sname FROM STUDENT where Sname like'a%';

SELECT Sname FROM STUDENT where Sname like'\_\_\_\_';

SELECT course,count( SID) FROM STUDENT group by course order by count(SID)desc;

SELECT course,count( SID) FROM STUDENT group by course having count(SID)>2;

Experiment – 1.3

1. amount + interest = Balance

---------------------------------------------------------------------------------

CREATE TABLE bank (

cid number,

amt number(5),

intrest number(5),

bal number(10)

);

INSERT INTO bank VALUES (101, 57000, 12343,0);

INSERT INTO bank VALUES (102, 1322, 4354,0);

declare

a number(5);

b number(5);

c number(10);

begin

SELECT amt,intrest into a,b from bank where cid=101;

c:=a+b;

update bank set bal = c where cid=101;

SELECT amt,intrest into a,b from bank where cid=102;

c:=a+b;

update bank set bal = c where cid=102;

end;

DROP TABLE bank;

SELECT \* FROM bank;

1. Area of circle

declare

r number;

begin

r:=:enter\_radius;

dbms\_output.put\_line('Area of Circle is ' || 3.14\*r\*r);

end;

----------------------------------------------------------------------------------------------

1. Perimeter and Area of Rectangle

-----------------------------------------------------------------------------

declare

l number;

b number;

begin

l:=:enter\_l;

b:=:enter\_b;

dbms\_output.put\_line('Perimeter of rectangle ' || 2\*(l+b));

dbms\_output.put\_line('AreA of rectangle ' || l\*b);

end;

1. Swap two Numbers

declare

a number;

b number;

begin

a:=:enter\_a;

b:=:enter\_b;

a:= a+b;

b:= a-b;

a:= a-b;

dbms\_output.put\_line('value of a and b is ' ||a||' '||b);

end;

1. Sum of Digits of a Number

Declare

n integer;

temp\_sum integer;

r integer;

Begin

n:=1234;

temp\_sum:=0;

WHILE n <> 0 LOOP

r := MOD(n, 10);

temp\_sum := temp\_sum + r;

n := Trunc(n / 10);

END LOOP;

dbms\_output.put\_line('sum of digits =' || temp\_sum);

END;

Experiment – 2.1

1. Greatest of Three numbers

DECLARE

a number;

b number;

c number;

BEGIN

a :=:Enter\_a;

b :=:Enter\_b;

c :=:Enter\_c;

IF a>b then

IF a>c then

dbms\_output.put\_line(a || ' is greatest');

ELSE

dbms\_output.put\_line(c || ' is greatest');

END if;

ELSE

IF b>c then

dbms\_output.put\_line(b || ' is greatest');

ELSE

dbms\_output.put\_line(c || ' is greatest');

END if;

END if;

END;

1. Number +ve, -ve or 0

DECLARE

a number;

BEGIN

a :=:Enter\_a;

IF a>0 THEN

dbms\_output.put\_line(a ||' is Positive');

ELSIF a<0 THEN

dbms\_output.put\_line(a ||' is Negative');

ELSE

dbms\_output.put\_line(a ||' is neither positive nor negative ');

END IF;

END;

1. Grade of a student

DECLARE

marks number;

BEGIN

marks :=:Enter\_marks;

IF marks>=90 THEN

dbms\_output.put\_line('GRADE A');

ELSIF marks>=70 THEN

dbms\_output.put\_line('GRADE B');

ELSIF marks>=50 THEN

dbms\_output.put\_line('GRADE C');

ELSIF marks>=40 THEN

dbms\_output.put\_line('GRADE D');

ELSE

dbms\_output.put\_line('GRADE F');

END IF;

END;

1. Project --- Bonus

CREATE TABLE information(

EID number(5),

Ename varchar2(20),

Project number(10),

Bonus number(10)

);

INSERT INTO information values (1, 'Harsh', 6, NULL);

INSERT INTO information values (2, 'Rohan', 4, NULL);

INSERT INTO information values (3, 'Aman',1, NULL);

SELECT \* from information;

DECLARE

accproj number;

accbon number;

BEGIN

SELECT Project into accproj from information where EID=1;

IF accproj>=5 THEN

accbon:=10000;

ELSIF accproj>=3 THEN

accbon:=5000;

ELSE

accbon:=2000;

END IF;

UPDATE information SET Bonus=accbon where EID=1;

SELECT Project into accproj from information where EID=2;

IF accproj>=5 THEN

accbon:=10000;

ELSIF accproj>=3 THEN

accbon:=5000;

ELSE

accbon:=2000;

END IF;

UPDATE information SET Bonus=accbon where EID = 2;

SELECT Project into accproj from information where EID=3;

IF accproj>=5 THEN

accbon:=10000;

ELSIF accproj>=3 THEN

accbon:=5000;

ELSE

accbon:=2000;

END IF;

UPDATE information SET Bonus=accbon where EID=3;

END;

Drop table information;

Experiment – 2.2

1. Sum of digits of a no. using while loop

Declare

n integer;

temp\_sum integer;

r integer;

Begin

n:=1234;

temp\_sum:=0;

WHILE n <> 0 LOOP

r := MOD(n, 10);

temp\_sum := temp\_sum + r;

n := Trunc(n / 10);

END LOOP;

dbms\_output.put\_line('sum of digits =' || temp\_sum);

END;

1. Factorial of a number using simple loop

Declare

num number;

a number;

fact number;

Begin

num:=:Enter\_number;

fact:=1;

a:=num;

Loop

fact:= fact\*a;

a:= a-1;

exit when a=1;

End Loop;

dbms\_output.put\_line('Factorial of '|| num ||' is '|| fact);

END;

1. Case statements for +, -, \* of two no.s

Declare

n number;

a number;

b number;

Begin

a:=:Enter\_number\_a;

b:=:Enter\_number\_b;

n:=:Enter\_choice;

Case n

When 1 then

dbms\_output.put\_line(a+b);

When 2 then

dbms\_output.put\_line(a-b);

When 3 then

dbms\_output.put\_line(a\*b);

Else

dbms\_output.put\_line('Wrong Choice');

End Case;

END;

1. Goto statement in Loop

DECLARE

a number := 10;

BEGIN

<<label>>

WHILE a < 20 LOOP

dbms\_output.put\_line (a);

a := a + 1;

IF a = 15 THEN

a := a + 1;

GOTO label;

END IF;

END LOOP;

END;

1. Using loops reduce the bill of the user by 10%

create table Sales (Cid number, Cname varchar(20), Cbill number);

insert into Sales values(1,'Harsh',1000);

insert into Sales values(2,'Rohan',500);

insert into Sales values(3,'Aditya',400);

select \* from Sales;

declare

i number;

bill number;

begin

for i in 1..3 LOOP

select Cbill into bill from Sales where Cid = i;

bill:=Trunc(bill/10);

update Sales set Cbill = bill where Cid = i;

end loop;

end;

select \* from Sales;

drop table Sales;

Experiment – 2.3 {Cursors}

CREATE TABLE Studinfo(

Sid number(10),

Sname varchar2(20),

sub1 number(5),

sub2 number(5),

sub3 number(5),

Total number(6)

);

INSERT into Studinfo values

(1,'Abhishek',50,54,49,153);

INSERT into Studinfo values

(2,'Rohan',20,34,29,83);

INSERT into Studinfo values

(3,'Himanshu',13,12,25,50);

INSERT into Studinfo values

(4,'Aditya Ruhela',1,5,7,13);

INSERT into Studinfo values

(5,'Aditya Barnwal',32,45,49,126);

INSERT into Studinfo values

(6,'Sameer',0,1,2,3);

INSERT into Studinfo values

(7,'Harsh',12,13,45,70);

INSERT into Studinfo values

(8,'Aarush',32,45,49,126);

INSERT into Studinfo values

(9,'Vivek',5,2,36,43);

INSERT into Studinfo values

(10,'Mike',2,90,23,115);

select \* from Studinfo;

drop table Studinfo;

-- Implicit Cursor (Deleting 1 row)

BEGIN

DELETE from Studinfo Where Sid=10;

if sql%found then

dbms\_output.put\_line('Row Deleted');

else

dbms\_output.put\_line('Row not Deleted');

END IF;

END;

select \* from Studinfo;

-- Implicit Cursor (Deleting multiple rows)

DECLARE

n number;

BEGIN

DELETE from Studinfo Where Total<50;

if sql%found then

n:= sql%rowcount;

dbms\_output.put\_line(n ||' rows Deleted');

else

dbms\_output.put\_line('No data found');

END IF;

END;

select \* from Studinfo;

-- Explicit Cursor

DECLARE

cursor c1 is select Sid, Sname, total from Studinfo Where Total<100;

rec c1%rowtype;

BEGIN

open c1;

LOOP

fetch c1 into rec;

EXIT WHEN c1%notfound;

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

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

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

END LOOP;

close c1;

END;

select \* from Studinfo;

drop table Studinfo;

Experiment – 2.4 {Views}

CREATE TABLE Studinfo(

Sid number(10),

Sname varchar2(20),

sub1 number(5),

sub2 number(5),

sub3 number(5),

Total number(6)

);

INSERT into Studinfo values

(1,'Abhishek',50,54,49,153);

INSERT into Studinfo values

(2,'Rohan',20,34,29,83);

INSERT into Studinfo values

(3,'Himanshu',13,12,25,50);

INSERT into Studinfo values

(4,'Aditya Ruhela',1,5,7,13);

INSERT into Studinfo values

(5,'Aditya Barnwal',32,45,49,126);

INSERT into Studinfo values

(6,'Sameer',0,1,2,3);

INSERT into Studinfo values

(7,'Harsh',12,13,45,70);

INSERT into Studinfo values

(8,'Aarush',32,45,49,126);

INSERT into Studinfo values

(9,'Vivek',5,2,36,43);

INSERT into Studinfo values

(10,'Mike',2,90,23,115);

-- Creating View----------------------

CREATE VIEW Studview AS

SELECT Sid, Sname, Total

FROM Studinfo

WHERE Total>60;

-- Updating Rows in View------------

UPDATE Studview

SET Sname='Rohit'

WHERE Sname='Mike';

-- Inserting Rows in view----

INSERT INTO Studview values

(11, 'Ross', 80);

-- Delete Rows from view-----------

DELETE FROM Studview

WHERE Sid=11;

-- Displaying View

select \* FROM Studview;

--Drop view

DROP view Studview;

select \* from Studinfo;

DROP VIEW Studview;

DROP TABLE Studinfo;

Experiment – 3.1 {Procedures}

1. Create a local subprogram to update the salary of all employees corresponding to “CSE” branch by same amount.

Create table emp(Eid number(10), Ename varchar2(20), salary number(5), dept varchar2(6));

insert into emp values(1, 'rohan', 1000, 'cse');

insert into emp values(2, 'barnwal', 100, 'ece');

insert into emp values(3, 'sameer', 700, 'cse');

insert into emp values(4, 'harsh', 200, 'ai');

insert into emp values(5, 'aditya', 10, 'ai');

insert into emp values(6, 'himanshu', 1000, 'cse');

insert into emp values(7, 'mrinal', 500, 'ai');

declare

amt number;

procedure salary\_raise\_cse(raise number) as

begin

update emp set salary = salary + raise where dept = 'cse';

end salary\_raise\_cse;

begin

amt:=:enter\_amount;

salary\_raise\_cse(amt);

end;

select \* from emp;

1. Delete a employee rows

----------------------------------------------------------------------------------------------

Create Procedure fire\_emp(emp\_no number) as

begin

delete from emp where Eid=emp\_no;

end fire\_emp;

----------------------------------------

declare

emp\_no number;

begin

emp\_no:=:enter\_emp\_no;

fire\_emp(emp\_no);

end;

select \* from emp;

drop table emp;

Experiment – 3.2 { Packages}

-- Creating Table------------

CREATE TABLE Employees(

Eid number(10),

Ename varchar2(20),

sal number(5),

dno number(2)

);

Drop table Employees;

-- Inserting into Table-------------

INSERT INTO Employees values

(1,'Harsh',1000,1);

INSERT INTO Employees values

(2,'Sinu',100,2);

INSERT INTO Employees values

(3,'Sameer',700,1);

INSERT INTO Employees values

(4,'Rohan',200,3);

INSERT INTO Employees values

(5,'Ruhela',10,3);

INSERT INTO Employees values

(6,'Himu',1000,1);

INSERT INTO Employees values

(7,'Mrinal',500,3);

Select \* from Employees;

-- Package Specification---------

Create Package operation as

procedure retrieve (Eno number, name out varchar, salary out number);

Function raise\_sal (Did number, amt number) return number;

End operation;

-- Package Body----------

Create package body operation as

procedure retrieve (Eno number, name out varchar, salary out number) as

begin

select Ename, sal into name, salary from Employees where Eid=Eno;

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

End retrieve;

Function raise\_sal (Did number, amt number) return number as

n number;

begin

update Employees set sal=sal+amt where Did=Dno;

n:= sql%rowcount;

return n;

End raise\_sal;

End operation;

Drop package operation;

-- Main Code--------------------------------

DECLARE

Empid number;

deptno number;

salary number;

amt number;

no\_rows number;

Empname varchar(30);

BEGIN

Empid:=:Enter\_id;

operation.retrieve(Empid, Empname, salary);

deptno:=:Enter\_dept\_no;

amt:=:Enter\_increment\_amount;

no\_rows:= operation.raise\_sal(deptno, amt);

dbms\_output.put\_line('Rows updated: '||no\_rows);

END;

--------------------------------------------------------------------

select \* from Employees;

Experiment 3.3 {Triggers}

-- Creating Table----------------

CREATE TABLE Employees(

Eid number(10),

Ename varchar2(20),

sal number(5),

dno number(2)

);

Drop table Employees;

-- Inserting into Table---------------

INSERT INTO Employees values

(1,'Harsh',1000,1);

INSERT INTO Employees values

(2,'Sinu',100,2);

INSERT INTO Employees values

(3,'Sameer',700,1);

INSERT INTO Employees values

(4,'Rohan',200,3);

INSERT INTO Employees values

(5,'Ruhela',10,3);

INSERT INTO Employees values

(6,'Himu',1000,1);

INSERT INTO Employees values

(7,'Mrinal',500,3);

Select \* from Employees;

-------------------------------------------------------------------------------------

Create Table Audit\_info(

Eno number(5),

name varchar2(20),

salary number(10),

Did number(5),

Operation varchar(8)

);

--------------------------------------------------------------------------------------------------------------

Create trigger audit\_trail after delete or update on Employees for each row

declare

opr varchar2(8);

begin

if updating then

opr:='Update';

end if;

if deleting then

opr:='Delete';

end if;

Insert into audit\_info values

(:old.Eid, :old.Ename, :old.sal, :old.dno, opr);

End audit\_trail;

-------------------------------------------------------------------------------------------

Update Employees SET dno=2 where eid=4;

Select \* from Employees;

Select \* from Audit\_info;

Delete from Employees where eid = 5;

LAB Questions

1. Discount in Movie Tickets

---------------------------------------------------------------------------------------

CREATE TABLE Movie(

Cid number(10),

Cname varchar2(20),

No\_of\_per number(6),

price number(6),

disc number(6)

);

INSERT into Movie values

(1,'Abhishek',50,NULL,NULL);

INSERT into Movie values

(2,'Rohan',20,NULL,NULL);

INSERT into Movie values

(3,'Himanshu',13,NULL,NULL);

INSERT into Movie values

(4,'Aditya Ruhela',2,NULL,NULL);

INSERT into Movie values

(5,'Aditya Barnwal',32,NULL,NULL);

INSERT into Movie values

(6,'Sameer',2,NULL,NULL);

INSERT into Movie values

(7,'Harsh',12,NULL,NULL);

INSERT into Movie values

(8,'Aarush',32,NULL,NULL);

INSERT into Movie values

(9,'Vivek',1,NULL,NULL);

INSERT into Movie values

(10,'Mike',2,NULL,NULL);

DECLARE

persons number;

i number;

cost number;

BEGIN

For i in 1..10

LOOP

Select no\_of\_per into persons from Movie where Cid=i;

cost:= 250\*persons;

UPDATE Movie SET price=cost where Cid=i;

END LOOP;

END;

SELECT Cid,Cname,No\_of\_per from Movie order by No\_of\_per desc;

DECLARE

persons number;

i number;

cost number;

discount number;

BEGIN

FOR i in 1..10

LOOP

Select no\_of\_per into persons from Movie where Cid=i;

Select price into cost from Movie where Cid=i;

IF persons>=5 THEN

discount:=(cost\*7)/100;

ELSIF persons>=3 THEN

discount:=(cost\*5)/100;

END IF;

UPDATE Movie SET disc=discount where Cid=i;

END LOOP;

END;

SELECT \* from Movie;

Drop TABLE Movie;

1. Students and their grade

------------------------------------------------------------------------------------------

Q1---------------------------------

CREATE TABLE Studinfo(

Sid number(10),

Sname varchar2(20),

sub1 number(5),

sub2 number(5),

sub3 number(5),

total number(6),

grade varchar(2)

);

Q2----------------------------------

INSERT into Studinfo values

(1,'Abhishek',50,54,49,NULL,NULL);

INSERT into Studinfo values

(2,'Rohan',20,34,29,NULL,NULL);

INSERT into Studinfo values

(3,'Himanshu',13,12,25,NULL,NULL);

INSERT into Studinfo values

(4,'Aditya Ruhela',1,5,7,NULL,NULL);

INSERT into Studinfo values

(5,'Aditya Barnwal',32,45,49,NULL,NULL);

INSERT into Studinfo values

(6,'Sameer',0,1,2,NULL,NULL);

INSERT into Studinfo values

(7,'Harsh',12,13,45,NULL,NULL);

INSERT into Studinfo values

(8,'Aarush',32,45,49,NULL,NULL);

INSERT into Studinfo values

(9,'Vivek',5,2,36,NULL,NULL);

INSERT into Studinfo values

(10,'Mike',2,90,23,NULL,NULL);

Q3----------------------------------

SELECT Sname from Studinfo WHERE Sname like'%a%';

Q4----------------------------------

DECLARE

x number;

G char;

s1 number;

s2 number;

s3 number;

i number;

BEGIN

x:=0;

FOR i in 1..10

LOOP

SELECT sub1,sub2,sub3 into s1,s2,s3 from Studinfo where Sid=i;

x:= s1+s2+s3;

UPDATE Studinfo SET total=x where Sid=i;

IF x>=120 THEN

G:='A';

ELSIF x>=100 THEN

G:='B';

ELSIF x>=50 THEN

G:='C';

ELSE

G:='D';

END IF;

UPDATE Studinfo SET grade=G where Sid=i;

END LOOP;

END;

Q5----------------------------------

Select sid,total from Studinfo order by total desc;

select \* from Studinfo;

drop TABLE Studinfo;

1. Customer discount

CREATE TABLE customer (

c\_id number (5),

c\_name varchar (10),

amount number (10),

discount number (10),

dept varchar (10)

);

insert into customer values (10,'SUNNY',11000, NULL,'SPORTS');

insert into customer values (9,'PRANAV',10000, NULL,'CLOTHES');

insert into customer values (8,'MUKUND',8000, NULL,'MOBILE');

insert into customer values (7,'MAYANK',7000, NULL,'WATCHES');

insert into customer values (6,'UMANG',6500, NULL,'SHOES');

insert into customer values (5,'SAHIL',6000, NULL,'COSMETIC');

insert into customer values (4,'AMAN',5500, NULL,'SLEEPER');

insert into customer values (3,'ROHIT',5000, NULL,'SPORTS');

insert into customer values (2,'PRINCE',4000, NULL,'CLOTHES');

insert into customer values (1,'ANKUR',3000, NULL,'SHOES');

SELECT c\_id, c\_name, amount from customer WHERE amount between 4000 and 10000;

SELECT dept, count (dept) from customer group by dept;

create table sales(

cid number(5),

c\_name varchar(10),

bill number(10)

);

DECLARE

x number(5);

c1 number(10);

c2 char(10);

s number(10);

i number(10);

a number(10);

BEGIN

x:=0;

FOR i in 1..10

LOOP

SELECT c\_id,c\_name,amount into c1,c2,a from customer where c\_id=i;

x:=a/10 ;

a:=a-x;

insert into sales values(c1,c2,a);

update customer set discount=x where c\_id=i;

END LOOP;

END;

select \* from sales;

select \* from customer;

Drop table sales;

Drop table customer;