Experiment No: 03

Experiment Name: Write SQL Queries to Case Expression.

SQL Queries:

1. create table Employee(ID int(5), NAME varchar(15), AGE int(2), SALARY int(10), DATE\_OF\_JOIN date);
2. select sum(SALARY),case when sum(SALARY) >55000 then 'GOOD' else 'BETTER' end as Remark from Employee order by ID;
3. select NAME from Employee where DATE\_OF\_JOIN > '2014-8-11';
4. select max(SALARY), case when max(SALARY) > 60000 then 'HIGH' else 'LOW' end as Remark from Employee;
5. select sum(distinct SALARY) from Employee;

Experiment No: 08

Experiment Name: Create Table and Update data into it using Trigger.

SQL Queries:

Before Update

1. create table DepartmentOne(D\_ID int(2), D\_Name varchar(15), D\_Sales int(10));
2. create table Before\_Update (D\_ID int(2), D\_Name varchar(10),Action varchar(15), Date date);

Create Trigger:

delimiter /

create trigger update\_details before update

* on DepartmentOne
* for each row
* begin
* insert into Before\_Update (D\_ID, D\_Name, Action, Date)values(old.D\_ID,old.D\_Name, ‘Updated’, now());
* end /

1. insert into DepartmentOne(D\_ID, D\_Name, D\_Sales)values(‘101’, ’Finance’, ’27500’) /
2. update DepartmentOne set D\_Name =’Marketing’ where D\_ID =’101’ /
3. select \* from DepartmentOne /
4. select \* from Before\_Update

SQL Queries:

After Update

1. create table DepartmentTwo(D\_ID int(2), D\_Name varchar(15), D\_Sales int(10));
2. create table After\_Update (D\_ID int(2), D\_Name varchar(10),Action varchar(15), Date date);

Create Trigger:

delimiter /

create trigger update\_detailsTwo after update

* on DepartmentTwo
* for each row
* begin
* insert into After\_Update (D\_ID, D\_Name, Action, Date)values(new.D\_ID,new.D\_Name, ‘Updated’, now());
* end /

1. insert into DepartmentTwo (D\_ID, D\_Name, D\_Sales)values(‘202’, ’Marketing’, ’37500’) /
2. update DepartmentTwo set D\_Name =’Research’ where D\_ID =’202’ /
3. select \* from DepartmentTwo /
4. select \* from After\_Update /