☑ Autocommit   **Display** [10 ▾]

```
select * from tab
```

Results Explain Describe Saved SQL History

no data found

☑Autocommit   Display [10   ▼]

```
CREATE TABLE Department
        ( did number(10) not null,
          dname varchar2(50) not null,
          CONSTRAINT did_pk PRIMARY KEY (did,dname)
        );
```

Results   Explain   Describe   Saved SQL   History

Table dropped.

0.07 seconds

☑Autocommit Display 10 ▼

```
CREATE TABLE Title
        ( tid number(10) not null,
          tname varchar2(50) not null,
          dtid number(10) not null,
          dname varchar2(50),
          CONSTRAINT tid_pk PRIMARY KEY (tid),
          CONSTRAINT fk_title
              FOREIGN KEY (dtid,dname)
              REFERENCES Department(did,dname)
        );
```

Results Explain Describe Saved SQL History

Table created.

0.01 seconds

☑ Autocommit Display [10 ▼]

```
CREATE TABLE Salary
        ( stotal number(10) not null,
          sdate date not null,
          tsid number(10) not null,
          CONSTRAINT sid_pk PRIMARY KEY (stotal),
          CONSTRAINT fk_salary
             FOREIGN KEY (tsid)
             REFERENCES Title(tid)
        );
```

Results Explain Describe Saved SQL History

Table created.

0.02 seconds

☑ Autocommit Display [10 ▾]

```
CREATE TABLE Employee
       ( eid number(10) not null,
         stotal number(10) not null,
         efname varchar2(50) not null,
         elname varchar2(50) not null,
         age number(10) not null,
         CONSTRAINT empl_pk PRIMARY KEY (eid),
         CONSTRAINT fk_employee
            FOREIGN KEY (stotal)
            REFERENCES Salary(stotal)
       );
```

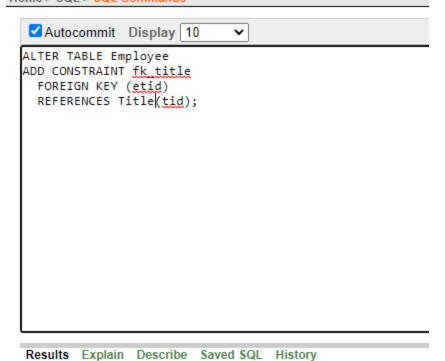**Results** Explain Describe Saved SQL History

Table created.

0.04 seconds

Autocommit   Display 10 ▾

```
ALTER TABLE Employee
ADD CONSTRAINT fk_title
   FOREIGN KEY (etid)
   REFERENCES Title(tid);
```

Results   Explain   Describe   Saved SQL   History

Table altered.

0.06 seconds

☑ Autocommit   Display [10   ▾]

```
select * from tab
```

Results   Explain   Describe   Saved SQL   History

TNAME	TABTYPE	CLUSTERID
EMPLOYEE	TABLE	-
TITLE	TABLE	-
DEPARTMENT	TABLE	-
SALARY	TABLE	-
BINSi0bTwp79RwuaJ1eh/HBpJQ==$0	TABLE	-

5 rows returned in 0.00 seconds   CSV Export

☑Autocommit   Display 10   ▼

```
INSERT ALL
  into department values (1,'Production')
  into department values (2,'Purchasing')
  into department values (3,'Marketing')
  into department values (4,'Accounting and Finance')
  into department values (5,'Research and Development')
  into department values (6,'Human Resource Management')
SELECT * FROM dual;
```

Results   Explain   Describe   Saved SQL   History

6 row(s) inserted.

0.00 seconds

☑ Autocommit Display [10 ▾]

```
INSERT ALL
  into title values (1,'Accountant',4,'')
  into title values (2,'Marketing Manager',3,'')
  into title values (3,'HR Manager',6,'')
  into title values (4,'Finance Manager',4,'')
  into title values (5,'Buisness Analyst',5,'')
  into title values (6,'Product Manager',1,'')
SELECT * FROM dual;
```

Results   Explain   Describe   Saved SQL   History

6 row(s) inserted.

0.00 seconds

☑ Autocommit   Display [10   ▾]

```
INSERT ALL
  into salary values (10000,TO_DATE('02/03', 'mm/dd'),1)
  into salary values (20000,TO_DATE('04/03', 'mm/dd'),2)
  into salary values (30000,TO_DATE('03/03', 'mm/dd'),3)
  into salary values (40000,TO_DATE('05/03', 'mm/dd'),4)
  into salary values (50000,TO_DATE('01/03', 'mm/dd'),5)
  into salary values (60000,TO_DATE('02/03', 'mm/dd'),6)
SELECT * FROM dual;
```

Results   Explain   Describe   Saved SQL   History

6 row(s) inserted.

0.00 seconds

☑Autocommit   Display [10   ▾]

```
INSERT ALL
  into employee values (1,10000,'Abebe','Meshesha',45,1)
  into employee values (2,20000,'Belay','Mengesha',40,2)
  into employee values (3,30000,'Fitsum','Nigatu',41,3)
  into employee values (4,40000,'Netsanet','Admassu',29,4)
  into employee values (5,50000,'Kidist','Belayneh',23,5)
  into employee values (6,60000,'Mekonin','Ashagre',33,6)
  into employee values (7,10000,'Mesfin','Shimeles',30,1)
  into employee values (8,20000,'Worku','Abayneh',52,2)
  into employee values (9,30000,'Daba','Debele',22,3)
  into employee values (10,40000,'Lemma','Abdissa',49,4)
  into employee values (11,50000,'Eden','Melaku',41,5)
  into employee values (12,60000,'Shibeshi','Mekuria',42,6)
  into employee values (13,10000,'Ruta','Asgedom',62,1)
  into employee values (14,20000,'Tsihon','Dawit',55,2)
  into employee values (15,30000,'Lidiya','Mehari',25,3)
SELECT * FROM dual;
```

Results   Explain   Describe   Saved SQL   History

15 row(s) inserted.

0.00 seconds

☑Autocommit Display 15   ⌄

```
insert all
 into employee values(1,10000,'Abebe','Meshesha',45,1, 4,'')
 into employee values(2,20000,'Belay','Mengesha',40,2,3,'')
 into employee values(3,30000,'Fitsum','Nigatu',41,3,6,'')
 into employee values(4,40000,'Netsanet','Admassu',29,4,4,'')
 into employee values(5,50000,'Kidist','Belayneh',23,5,5,'')
 into employee values(6,60000,'Mekonin','Ashagre',33,6,1,'')
 into employee values(7,10000,'Mesfin','Shimeles',30,1,4,'')
 into employee values(8,20000,'Worku','Abayneh',52,2,3,'')
 into employee values(9,30000,'Daba','Debele',22,3,6,'')
 into employee values(10,40000,'Lemma','Abdissa',49,4,4,'')
 into employee values(11,50000,'Eden','Melaku',41,5,5,'')
 into employee values(12,60000,'Shibeshi','Mekuria',42,6,1,'')
 into employee values(13,10000,'Ruta','Asgedom',62,1,4,'')
 into employee values(14,20000,'Tsihon','Dawit',55,2,3,'')
 into employee values(15,30000,'Lidiya','Mehari',25,3,6,'')
select * from dual;
```

Results   Explain   Describe   Saved SQL   History

15 row(s) inserted.

0.00 seconds

☑ Autocommit   Display [15   ▼]

```
DECLARE

 cursor selectDname is select did,dname from Department order by dname;
 cursor Employees is select edid,efname from Employee order by efname;
 counter number;
 ordering number:=0;

PROCEDURE pro_Department_report IS

BEGIN
 FOR dept in selectDname LOOP
  dbms_output.put_line('Department: '|| dept.dname);
  select count(eid) into counter from Employee where edid=dept.did;
  dbms_output.put_line('Total number of Employees: '|| counter);
  dbms_output.put_line('------------');
  FOR emp in Employees LOOP
   if(emp.edid=dept.did) then
    ordering:=ordering+1;
    dbms_output.put_line(ordering||'. '|| emp.efname);
   end if;
  END LOOP;
  ordering:=0;
  dbms_output.put_line('');
 END LOOP;
END;

BEGIN
 pro_Department_report();

END
```

```
Department: Accounting and Finance
Total number of Employees: 5
------------
1. Abebe
2. Lemma
3. Mesfin
4. Netsanet
5. Ruta

Department: Human Resource Management
Total number of Employees: 3
------------
1. Daba
2. Fitsum
3. Lidiya

Department: Marketing
Total number of Employees: 3
------------
1. Belay
2. Tsihon
3. Worku

Department: Production
Total number of Employees: 2
------------
1. Mekonin
2. Shibeshi

Department: Purchasing
Total number of Employees: 0
------------

Department: Research and Development
Total number of Employees: 2
------------
1. Eden
2. Kidist


Statement processed.
```

☑ Autocommit   Display 15 ▾

```
DECLARE

 cursor selectDname is select did,dname from Department order by dname;
 cursor Employees is select edid,efname,stotal from Employee order by efname;
 counter number;
 ordering number:=0;

PROCEDURE pro_employee_stats IS

BEGIN
 dbms_output.put_line('Employee Name Salary');
 dbms_output.put_line('------------- ------');
 FOR emp in Employees LOOP
   dbms_output.put_line(emp.efname  ||'            '||emp.stotal);
 END LOOP;
END;

BEGIN
 pro_employee_stats ();

END
```

Results   Explain   Describe   Saved SQL   History

```
Employee Name Salary
------------- ------
Abebe          10000
Belay          20000
Daba         30000
Eden         50000
Fitsum         30000
Kidist         50000
Lemma        40000
Lidiya         30000
Mekonin        60000
Mesfin       10000
Netsanet        40000
Ruta         10000
Shibeshi        60000
Tsihon       20000
Worku        20000

Statement processed.


0.00 seconds
```

☑ Autocommit   Display [15 ▾]

```
DECLARE
 cursor titles is select tid,tname,dtid from Title;
 cursor salaries is select tsid,stotal from Salary;
 cursor depts is select did,dname from Department;
 empID number;
 titleID number;
 edname varchar2;
 stotal number;
 eid number;
 ordering number:=0;
 lessThan21 Exception;

PROCEDURE pro_add(efname IN varchar2,elname IN varchar2,age IN number,etitle IN varchar2,edid IN number) IS

BEGIN
 if(age<21) then
  raise lessThan21;
 else
  FOR title in titles LOOP
   if(title.tname=etitle) then
    titleID:=title.tid;
    edid:=title.dtid;
    etitle:=title.tname;
    FOR salary in salaries LOOP
     if (salary.tsid=title.tid) then
      stotal:=salary.stotal;
     else
      dbms_output.put_line('Inappropriate salary for this title.');
     end if;
   else
    select count(tid) into titleID from title;
    FOR dept in depts LOOP
     if(dept.did=edid) then
      edid:=dept.did;
      edname:=dept.dname;
      insert into title values(titleID+1,etitle,edid,edname);
```

Autocommit   Display [15   ∨]

```
   if(title.tname=etitle) then
    titleID:=title.tid;
    edid:=title.dtid;
    etitle:=title.tname;
    FOR salary in salaries LOOP
     if (salary.tsid=title.tid) then
      stotal:=salary.stotal;
     else
      dbms_output.put_line('Inappropriate salary for this title.');
     end if;
   else
    select count(tid) into titleID from title;
    FOR dept in depts LOOP
     if(dept.did=edid) then
      edid:=dept.did;
      edname:=dept.dname;
      insert into title values(titleID+1,etitle,edid,edname);
     else
      dbms_ouptput.put_line('No such department.');
     end if;
    END LOOP;
   end if;
  END LOOP;
  select count(eid) into empID from employee;
  insert into employee values (empid+1,stotal,efname,elname,age,edid,titleID,edname);
 end if;
 EXCEPTION
  when lessThan21 then
   dbms_output.put_line('Illegible.' )
END;
BEGIN
 pro_add('Absalat','Lname',24,'Manager',1);
 pro_add('Kidist','Lname',26,'Analyst',4);

END
```

☑ Autocommit   Display 15 ⌄

```
DECLARE
 cursor empSal select fname,stotal from employee;
FUNCTION calc_avg
 return float IS
   averagesal float;
   maxPaid float;
   minPaid float;

BEGIN
 select MAX(stotal) into maxPaid from Salary;
 select MIN(stotal) into minPaid from Salary;

 dbms_output.put_line('----------------------------');
 FOR emp in empSal LOOP
  if(emp.etotal=maxPaid) then
    dbms_output.put_line('Maximum salary paid employee:'||emp.fname);
   elseif (emp.total=minPaid) then
    dbms_output.put_line('Minimum salary paid employee:'||emp.fname);
  end if;
 END LOOP;
 select ROUND(AVG(stotal),2) into averagesal from salary;
 return averagesal;
END;
BEGIN
 dbms_output.put_line('Avergae Salary: '|| calc_avg());

END
 |
```