* Delete division

create or replace procedure deleteDivision(ids in varchar2)  
IS  
begin  
delete from division where id = ids;  
end;

​

* Delete Employee

create or replace procedure deleteEmployee(ids in number)  
is  
begin  
delete from employee where id = ids;  
end;

​

* Delete leave

create or replace procedure deleteLeave(ids in number)  
is  
begin  
delete from leave where id = ids;  
end;

​

* Delete Position

create or replace procedure deletePosition(ids in varchar2)  
is  
begin  
delete from position where id = ids;  
end;

​

* Delete Take Leave

create or replace procedure deleteTakeLeave(ids in varchar2)  
is  
begin  
delete from takeleave where id = ids;  
end;​

* Get Divisiion

create or replace procedure getDivision  
IS  
 cursor d\_data is select \* from division;  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
 dbms\_output.put\_line('----------');  
 dbms\_output.put\_line('id : '||d\_rec.id);  
 dbms\_output.put\_line('name : '||d\_rec.name);  
end loop;  
end;

* Get Employee

create or replace procedure getEmployee  
IS  
 cursor d\_data is select \*   
 from employee;  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
dbms\_output.put\_line('----------');  
dbms\_output.put\_line('ID                 : '||d\_rec.id);  
dbms\_output.put\_line('Name               : '||d\_rec.name);  
dbms\_output.put\_line('Gender             : '||d\_rec.gender);  
dbms\_output.put\_line('Marriage           : '||d\_rec.marriage);  
dbms\_output.put\_line('Child              : '||d\_rec.children\_total);  
dbms\_output.put\_line('Username           : '||d\_rec.username);  
dbms\_output.put\_line('Password           : '||d\_rec.password);  
dbms\_output.put\_line('Last Year Quota    : '||d\_rec.last\_year);  
dbms\_output.put\_line('This Year Quota    : '||d\_rec.this\_year);  
dbms\_output.put\_line('Division           : '||d\_rec.division\_id);  
dbms\_output.put\_line('Position           : '||d\_rec.position\_id);  
dbms\_output.put\_line('Religion           : '||d\_rec.religion);  
dbms\_output.put\_line('Manager           : '||d\_rec.manager\_id);  
end loop;  
end;

​

* Get leave

create or replace procedure getLeave  
IS  
 cursor d\_data is select \* from leave;  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
 dbms\_output.put\_line('----------');  
 dbms\_output.put\_line('id     : '||d\_rec.id);  
 dbms\_output.put\_line('name   : '||d\_rec.name);  
 dbms\_output.put\_line('status : '||d\_rec.status);  
 dbms\_output.put\_line('days   : '||d\_rec.days);  
end loop;  
end;

​

* Get Leave Emp

create or replace procedure getLeaveEmp(ids in number)  
IS  
 cursor d\_data is select \* from employee where id=ids;  
 cursor consatu is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Menikah';  
 cursor condua is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Menikah' and name != 'Cuti Umroh' and name != 'Cuti Haji';  
 cursor contiga is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Melahirkan';   
 cursor conempat is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Melahirkan' and name != 'Cuti Umroh' and name != 'Cuti Haji';   
 cursor conlima is select name from leave where name != 'Cuti Melahirkan' and name != 'Cuti Menikah';   
 cursor conenam is select name from leave where name != 'Cuti Menikah' and name != 'Cuti Melahirkan' and name != 'Cuti Umroh' and name != 'Cuti Haji';  
 cursor contuju is select name from leave where name != 'Cuti Melahirkan' and name != 'Cuti Sunat';  
 cursor kelapa is select name from leave where name != 'Cuti Melahirkan' and name != 'Cuti Sunat' and name != 'Cuti Umroh' and name != 'Cuti Haji';   
 consatu\_rec  consatu%rowtype;  
 condua\_rec   condua%rowtype;  
 contiga\_rec  contiga%rowtype;  
 conempat\_rec conempat%rowtype;  
 conlima\_rec  conlima%rowtype;  
 conenam\_rec  conenam%rowtype;  
 contuju\_rec  contuju%rowtype;  
 kelapa\_rec   kelapa%rowtype;  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
if(d\_rec.gender = 'F' and d\_rec.marriage = 'Y' and d\_rec.religion = 'Islam') then  
  
for consatu\_rec in consatu loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||consatu\_rec.name);  
end loop;  
  
elsif(d\_rec.gender = 'F' and d\_rec.marriage = 'Y' and d\_rec.religion != 'Islam') then  
for condua\_rec in condua loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||condua\_rec.name);  
end loop;  
  
elsif(d\_rec.gender = 'F' and d\_rec.marriage = 'N' and d\_rec.religion = 'Islam') then  
for contiga\_rec in contiga loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||contiga\_rec.name);  
end loop;  
  
elsif(d\_rec.gender = 'F' and d\_rec.marriage = 'N' and d\_rec.religion != 'Islam') then  
for conempat\_rec in conempat loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||conempat\_rec.name);  
end loop;  
  
elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'Y' and d\_rec.religion = 'Islam' and d\_rec.children\_total>0) then  
for conlima\_rec in conlima loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||conlima\_rec.name);  
end loop;  
  
elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'Y' and d\_rec.religion != 'Islam' and d\_rec.children\_total>0) then  
for conenam\_rec in conenam loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||conenam\_rec.name);  
end loop;  
  
elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'N' and d\_rec.religion = 'Islam') then  
for contuju\_rec in contuju loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||contuju\_rec.name);  
end loop;  
  
elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'N' and d\_rec.religion != 'Islam') then  
for kelapa\_rec in kelapa loop  
dbms\_output.put\_line('-----------');  
dbms\_output.put\_line('Nama Cuti: '||kelapa\_rec.name);  
end loop;  
  
else  
dbms\_output.put\_line('apalah');  
end if;  
end loop;  
end;

​

* Get log

create or replace procedure getLog  
IS  
 cursor d\_data is select l.log\_date,e.name  
   from log l join employee e  
   on l.employee\_id = e.id;  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
 dbms\_output.put\_line('----------');  
 dbms\_output.put\_line('log\_date : '||d\_rec.log\_date);  
 dbms\_output.put\_line('employee : '||d\_rec.name);  
end loop;  
end;

​

* Get login

create or replace procedure getlogin(usernames in varchar2)  
is  
 cursor userpass is select e.username, e.password, p.name as name from employee e join position p on e.position\_id=p.id where e.username = usernames;  
 userpass\_rec userpass%rowtype;  
begin  
for userpass\_rec in userpass loop  
dbms\_output.put\_line('--------');  
dbms\_output.put\_line('username: '||userpass\_rec.username);  
dbms\_output.put\_line('password: '||userpass\_rec.password);  
dbms\_output.put\_line('position: '||userpass\_rec.name);  
end loop;  
end;​

* Get position

create or replace procedure getPosition  
IS  
 cursor d\_data is select \* from position;  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
 dbms\_output.put\_line('----------');  
 dbms\_output.put\_line('id : '||d\_rec.id);  
 dbms\_output.put\_line('name : '||d\_rec.name);  
end loop;  
end;

​

* Get quota

create or replace procedure getQuota(ids in number)  
IS  
 cursor quota is select last\_year, this\_year from employee where id = ids;  
 quota\_rec quota%rowtype;  
 total number;  
 lastyear number;  
 thisyear number;  
begin  
for quota\_rec in quota loop  
lastyear := quota\_rec.last\_year;  
thisyear := quota\_rec.this\_year;  
total := thisyear+lastyear;  
dbms\_output.put\_line('Quota Last Year: '||lastyear);  
dbms\_output.put\_line('Quota This Year: '||thisyear);  
dbms\_output.put\_line('Total Quota: '||total);  
end loop;  
end;

​

* Get take leave manager

create or replace procedure getTakeLeave  
IS  
 cursor d\_data is select e.id,e.name,l.name as cuti,t.description,t.date\_start,t.date\_end,t.approval\_status   
    from takeleave t join employee e on t.employee\_id=e.id join leave l on l.id=t.leave\_id  
    where t.approval\_status='WAITING';  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
 dbms\_output.put\_line('----------');  
 dbms\_output.put\_line('id                 : '||d\_rec.id);  
 dbms\_output.put\_line('name               : '||d\_rec.name);  
 dbms\_output.put\_line('cuti               : '||d\_rec.cuti);  
 dbms\_output.put\_line('description        : '||d\_rec.description);  
 dbms\_output.put\_line('date\_start         : '||d\_rec.date\_start);  
 dbms\_output.put\_line('date\_end           : '||d\_rec.date\_end);  
 dbms\_output.put\_line('approval status    : '||d\_rec.approval\_status);  
end loop;  
end;

​

* Get take leave admin

create or replace procedure getTakeLeaveAdm  
IS  
 cursor alldata is select e.name, l.name as cuti, t.description, t.date\_start, t.date\_end, t.selisih, t.approval\_status, t.approval\_date,  
    (select ee.name from employee ee where ee.id=e.manager\_id) as manager  
 from takeleave t join employee e on t.employee\_id=e.id  
 join leave l on t.leave\_id=l.id;  
 alldata\_rec alldata%rowtype;  
begin  
 for alldata\_rec in alldata loop  
 dbms\_output.put\_line('Name            : '||alldata\_rec.name);  
 dbms\_output.put\_line('Cuti            : '||alldata\_rec.cuti);  
 dbms\_output.put\_line('Description     : '||alldata\_rec.description);  
 dbms\_output.put\_line('Date Start      : '||alldata\_rec.date\_start);  
 dbms\_output.put\_line('Date End        : '||alldata\_rec.date\_end);  
 dbms\_output.put\_line('Lama Cuti       : '||alldata\_rec.selisih);  
 dbms\_output.put\_line('Approval Status : '||alldata\_rec.approval\_status);  
 dbms\_output.put\_line('Approval Date   : '||alldata\_rec.approval\_date);  
 dbms\_output.put\_line('Manager         : '||alldata\_rec.manager);  
 dbms\_output.put\_line('');  
end loop;  
end;

​

* Get take leave employee

create or replace procedure getTakeLeaveEmp(ids in number)  
IS  
 cursor d\_data is select l.name as cuti,t.description,t.date\_start,t.date\_end,t.approval\_status,t.approval\_date,  
 (select ee.name from employee ee where ee.id=e.manager\_id) as manager  
 from takeleave t join employee e on t.employee\_id=e.id join leave l on l.id=t.leave\_id  
 where e.id=ids;  
 d\_rec d\_data%rowtype;  
begin  
for d\_rec in d\_data loop  
 dbms\_output.put\_line('----------');  
 dbms\_output.put\_line('cuti            : '||d\_rec.cuti);  
 dbms\_output.put\_line('description     : '||d\_rec.description);  
 dbms\_output.put\_line('date\_start      : '||d\_rec.date\_start);  
 dbms\_output.put\_line('date\_end        : '||d\_rec.date\_end);  
 dbms\_output.put\_line('approval status : '||d\_rec.approval\_status);  
 dbms\_output.put\_line('approval date   : '||d\_rec.approval\_date);  
 dbms\_output.put\_line('manager         : '||d\_rec.manager);  
end loop;  
end;

​

* Insert division

create or replace procedure insertDivision(ids in varchar2, names in varchar2)  
IS  
begin  
insert into division values(ids, names);  
end;

​

* Insert employee

create or replace procedure insertEmployee(id in number, name in varchar2, gender in varchar2, marriage in varchar2, children\_total in number, username in varchar2, password in varchar2, last\_year in number, this\_year in number, division\_id in varchar2, position\_id in varchar2, religion in varchar2, manager\_id in number)  
is  
begin  
insert into employee values(id, name, gender, marriage, children\_total, username, password, last\_year, this\_year, division\_id, position\_id, religion, manager\_id);  
end;

​

* Insert leave

create or replace procedure insertLeave(id in number, name in varchar2, status in varchar, days in number)  
is  
begin  
insert into leave values(id, name, status, days);  
end;

​

* Insert log

create or replace procedure insertLog(log\_date in date,employee\_id in number)  
is  
begin  
insert into log values(to\_date(log\_date,'DD/MM/YYYY'),employee\_id);  
end;

​

* Insert position

create or replace procedure insertPosition(id in varchar2, name in varchar2)  
is  
begin  
insert into position values(id, name);  
end;

​

* Insert take leave

create or replace procedure insertTakeLeave(id in varchar2, description in varchar2, date\_start in date, date\_end in date, approval\_status in varchar2, employee\_id in number, leave\_id in number,selisih in number, approval\_date in date)  
is  
begin  
insert into takeleave values(id, description, to\_date(date\_start,'DD/MM/YYYY'), to\_date(date\_end,'DD/MM/YYYY'), approval\_status, employee\_id, leave\_id, selisih, to\_date(approval\_date,'DD/MM/YYYY'));  
end;

​

* Update division

create or replace procedure updateDivision(ids in varchar2, names in varchar2)  
IS  
begin  
update division set name = names where id = ids;  
end;

​

* Update employee

create or replace procedure updateEmployee(ids in number, names in varchar2, genders in varchar2, marriages in varchar2, children\_totals in number, usernames in varchar2, passwords in varchar2, last\_years in number, this\_years in number, division\_ids in varchar2, position\_ids in varchar2, religions in varchar2, manager\_ids in number)  
is  
begin  
update employee set name = names, gender = genders, marriage = marriages, children\_total = children\_totals, username = usernames, password = passwords, last\_year = last\_years, this\_year = this\_years, division\_id = division\_ids, position\_id = position\_ids, religion = religions, manager\_id = manager\_ids where id = ids;  
end;

​

* Update leave

create or replace procedure updateLeave(ids in number, names in varchar2, statuss in varchar, dayss in number)  
is  
begin  
update leave set name = names, status = statuss, days = dayss where id = ids;  
end;

​

* Update position

create or replace procedure updatePosition(ids in varchar2, names in varchar2)  
is  
begin  
update position set name = names where id = ids;  
end;​

* Update take leave

create or replace procedure updateTakeLeave(ids in varchar2, descriptions in varchar2, date\_starts in date, date\_ends in date, approval\_statuss in varchar2, employee\_ids in number, leave\_ids in number, selisihs in number, approval\_dates in date)  
is  
begin  
update takeleave set description = descriptions, date\_start = to\_date(date\_starts,'DD/MM/YYYY'), date\_end = to\_date(date\_ends,'DD/MM/YYYY'), approval\_status = approval\_statuss, employee\_id = employee\_ids, leave\_id = leave\_ids, selisih = selisihs, approval\_date = to\_date(approval\_dates,'DD/MM/YYYY') where id = ids;  
end;

​

* Subtract quota

create or replace trigger subtract\_quota  
after insert on takeleave  
for each row  
declare  
selisihs number;  
eid number;  
  
begin  
selisihs := :new.selisih;   
eid := :new.employee\_id;  
update employee set this\_year = this\_year - selisihs where id = eid;  
end;

​

* Reschedule

create or replace trigger reschedule  
after update on takeleave  
for each row  
declare  
selisihs number;  
selisih\_lama number;  
eid number;  
begin  
selisih\_lama := :old.selisih;  
eid := :new.employee\_id;  
update employee set this\_year=this\_year+selisih\_lama where id=eid;  
selisihs := :new.selisih;   
update employee set this\_year=this\_year-selisihs where id=eid;  
end;​

​

* Cancel leave

create or replace trigger cancel\_leave  
after update on takeleave  
for each row  
when(new.approval\_status='CANCELED')  
declare  
selisihs number;  
eid number;  
begin  
selisihs := :new.selisih;   
eid := :new.employee\_id;  
update employee set this\_year=this\_year+selisihs where id = eid;  
end;​

​

* Rejected

create or replace trigger rejected  
after update on takeleave   
for each row  
when(new.approval\_status='REJECTED')   
declare  
selisihs number;  
eid number;  
begin      
selisihs := :new.selisih;   
eid := :new.employee\_id;  
update employee set this\_year=this\_year+selisihs where id=eid;  
end;​

​

* Reset quota this year

create or replace trigger reset\_quota\_this\_year  
after insert on log   
for each row   
when(new.log\_date=to\_date('02/01','DD/MM'))  
begin  
update employee set this\_year=12;  
end;​

* Reset quota last year

create or replace trigger reset\_quota\_last\_year  
after insert on log   
for each row   
when(new.log\_date=to\_date('01/07','DD/MM'))  
begin  
update employee set last\_year=0;  
end;​

* Birth leave on insert

create or replace trigger birthleaveinit  
after insert on takeleave  
for each row  
declare  
names leave.name%type;  
begin  
select name into names from leave where id = :new.leave\_id;  
if(names = 'Cuti Melahirkan') then  
update employee set children\_total=children\_total+1 where id = :NEW.EMPLOYEE\_ID;  
end if;  
end;

​

* Birth leave

create or replace trigger birthleave  
after update on takeleave  
for each row  
declare  
    names leave.name%type;  
    appstats varchar2(15);  
begin  
 select name into names from leave where id = :new.leave\_id;  
 appstats := :new.approval\_status;  
IF((names = 'Cuti Melahirkan' and appstats = 'CANCELED') or (names = 'Cuti Melahirkan' and appstats = 'REJECTED')) then      
update employee set children\_total=children\_total-1 where id = :NEW.EMPLOYEE\_ID;  
end if;  
end;​