**PORCEDURE**

1. **Create\_customer**

create or replace procedure creaet\_customer(

name IN customer.cus\_name%type,

ph\_no in customer.cus\_no%type,

address in customer.address%type,

uname in customer.user\_name%type,

pass customer.password%type) as card\_id number;

id customer.cus\_id%type;

begin

card\_id := card\_seq.nextval;

select cus\_id into id from customer where user\_name = uname;

if id>0 then

dbms\_output.put\_line('User is alread exists....');

end if;

exception

when no\_data\_found then

id :=cus\_seq.nextval;

insert into card values(card\_id,0,'on');

insert into customer values(id,name,ph\_no,card\_id,pass,uname,sysdate,address);

dbms\_output.put\_line('Wlecome '||name||' your account is been created....');

dbms\_output.put\_line(' your account id is....'|| id);

end;

1. **Create\_Employee**

create or replace procedure create\_employee(

name IN employee.emp\_name%type,

ph\_no in employee.emp\_no%type,

address in employee.emp\_address%type,

uname in employee.user\_name%type,

pass employee.password%type,

b\_name employee.branch\_name%type

) as id employee.emp\_id%type;

branch\_rebranch%rowtype;

e\_invalid\_branch exception;

begin

select emp\_id into id from employee where user\_name = uname;

if id>0 then

dbms\_output.put\_line('User is alread exists....');

end if;

select \* into branch\_re from branch where branch\_name= b\_name;

if sql%notfound then

raise e\_invalid\_branch ;

end if;

exception

when no\_data\_found then

insert into card values(card\_id,0,'on');

insert into employee values(

emp\_seq.nextval,

name,

address,

ph\_no,

pass,

uname,

5660,

sysdate,

b\_name );

dbms\_output.put\_line('Welcome '||name||' your account is been created....');

when e\_invalid\_branch then

insert into branch(branch\_name) values(b\_name);

end;

1. **Create\_account**

create or replace procedure create\_accoutn (type in varchar) as fname customer.cus\_name%type;

p\_nocustomer.cus\_no%type;

address customer.address%type;

unamecustomer.user\_name%type;

pass customer.password%type;

add customer.address%type;

begin

if type = 'customer' then

Fname := '&Fname' ;

uname := '&user\_name';

pass:= '&password';

p\_no := &phoneno;

add :='&address';

cre\_cus(fname,p\_no,add,uname,pass);

end if;

end;

1. **Delete\_customer**

create or replace procedure delete\_customer (id in customer.cus\_id%type) as

card number;

uname varchar2(20);

r\_id number;

begin

uname := user\_validity(id);

if uname=null then

dbms\_output.put\_line('account id dpoesntoexsists');

else

select card\_id into card from customer where cus\_id=id;

select rent\_id into r\_id from rent where cus\_id = id;

delete from rent where cus\_id=id;

delete from customer where cus\_id =card;

delete from card where c\_id=card;

dbms\_output.put\_line(uname || ' your Account id : '|| id|| ' is been deleted...' );

end if;

exception

when no\_data\_found then

delete from customer where cus\_id=id;

delete from card where c\_id=card;

dbms\_output.put\_line(uname || ' your Account id : '|| id|| ' is been deleted...' );

end;

1. **Delete\_employee**

create or replace procedure delete\_employee (id employee.emp\_id%type) as

uname employee.user\_name%type;

begin

select user\_name into uname from employee where emp\_id=id;

if sql%notfound then

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

else

delete from employee where emp\_id =id;

dbms\_output.put\_line(uname || ' , your id '|| id || 'is been deleted');

end if;

end;

1. **Rent\_pro**

create or replace procedure rent\_pro (

name in varchar2,

customer\_id in customer.cus\_id%type) as id number;

card\_id number;

card\_fine\_rup number;

valid varchar2(10);

begin

valid := user\_validity(customer\_id);

if valid is null then

dbms\_output.put\_line(customer\_id || ' does not exsists....Please check again');

return;

else

card\_id :=find\_card\_id(customer\_id);

if card\_id is null then

dbms\_output.put\_line(customer\_id || ' does not exsists....Please check again');

return;

else

card\_fine\_rup := card\_fine(card\_id);

if card\_fine\_rup> 0 then

return;

else

dbms\_output.put\_line(customer\_id || ' everything is perfect');

end if;

end if;

end if;

end ;

1. **Issue\_Book\_Video**

create or replace procedure issue\_book\_video (

name in varchar2,

customer\_id in customer.cus\_id%type) as flag varchar2(6);

card\_id number;

book number :=0;

video number :=0;

avail number ;

item\_id number;

begin

flag :=fine\_from\_cus\_id(customer\_id);

if flag = 'true' then

item\_id :=book\_find(name,avail);

book :=1;

if item\_id< 0 then

item\_id :=video\_find(name,avail);

video :=1;

end if;

if item\_id> 0 then

if avail >0 then

card\_id :=find\_card\_id(customer\_id);

update card set issue\_date=sysdate where c\_id=card\_id;

update card set return\_date=sysdate+15 where c\_id=card\_id;

insert into rent values (

sysdate,

sysdate+15,

customer\_id,

rent\_seq.nextval,

item\_id);

if book>0 then

update book set availability = availability-1 where book\_id=item\_id;

dbms\_output.put\_line('video is been issued by ' || item\_id||' is '||name);

elsif video >0 then

update video set availability =availability-1 where video\_id=item\_id;

dbms\_output.put\_line('video is been issued by ' || item\_id||' is '||name);

else

dbms\_output.put\_line('Something is wrong');

return;

end if;

else

dbms\_output.put\_line(name || ' right now is not available');

return;

end if;

else

dbms\_output.put\_line(name || ' there is not such item available');

return;

end if;

else

return;

end if;

end ;

1. **Book\_Add**

create or replace procedure book\_add(

name in book.b\_name %type,

total in book.total\_book%type) as avail number;

book\_id number;

begin

book\_id :=book\_find(name,avail);

if book\_id>0 then

dbms\_output.put\_line('Book is already in database with id '|| book\_id);

dbms\_output.put\_line('Total availability is '|| total);

else

insert into book values (book\_seq.nextval,total,total,upper(name|| ‘:The Book’),sysdate);

dbms\_output.put\_line('Book is successfully added....');

end if;

end;

1. **Video\_Add**

create or replace procedure video\_add(name in video.v\_name %type,total in video.total\_video%type) as

avail number;

video\_id number;

begin

video\_id :=book\_find(name,avail);

if video\_id>0 then

dbms\_output.put\_line('video is already in database with id '|| video\_id);

dbms\_output.put\_line('Total availability is '|| total);

else

insert into video values (video\_seq.nextval,total,total,upper(name|| ':The video'),sysdate);

dbms\_output.put\_line('Video is successfully added....');

end if;

end;

1. **Book\_Del**

create or replace procedure book\_del(name in book.b\_name %type) as

avail number;

b\_id number;

begin

b\_id :=book\_find(name,avail);

if b\_id>0 then

if avail > 0 then

delete from book where book\_id =b\_id;

dbms\_output.put\_line(name || ' is been deleted');

dbms\_output.put\_line('Total book deleted is/are :'|| avail);

elsif avail < 0 then

dbms\_output.put\_line('Availability of book' || name ||' is/are:' ||avail);

end if;

else

dbms\_output.put\_line('There is no book with name '||name ||' is in database');

end if;

end;

1. **Video\_Del**

create or replace procedure video\_del(name in video.v\_name %type) as

avail number;

v\_id number;

begin

v\_id :=video\_find(name,avail);

if v\_id>0 then

if avail > 0 then

delete from video where video\_id =v\_id;

dbms\_output.put\_line(name || ' is been deleted');

dbms\_output.put\_line('Total video deleted is/are :'|| avail);

elsif avail < 0 then

dbms\_output.put\_line('Availability of video' || name ||' is/are:' ||avail);

end if;

else

dbms\_output.put\_line('There is no video with name '||name ||' is in database');

end if;

end;