

Devashish kankanwar

```
CREATE table book(
```

```
Book_no varchar2(20) NOT NULL,
```

```
Book_name varchar2(20),
```

```
Author varchar2(20),
```

```
price varchar2(20),
```

```
no_of_books number(4),
```

```
PRIMARY KEY (Book_no)
```

```
);
```

```
INSERT into book values('1','Rich dad poor
```

```
dad','XX','399','30');
```

```
INSERT into book values('2','The alchemist','XX','199','20');
```

```
INSERT into book values('3','The atomic
```

```
habbits','XX','399','50');
```

```
INSERT into book values('4','the heist','XX','599','35');
```

```
select * from book;
```

```
CREATE TABLE MEMBER(
```

```
Mem_no varchar2(20),
```

```
Mem_name varchar2(20) not null,
```

```
Mem_type varchar2(20),
```

```
No_of_books number(4),
```

```
Total_fine number(4),
```

```
CONSTRAINT Member PRIMARY KEY
```

```
(
```

Mem_no

)

ENABLE

);

INSERT INTO MEMBER

VALUES('100','SUMEET','M','4','150');

INSERT INTO MEMBER

VALUES('200','DEVASHISH','Y','10','50');

INSERT INTO MEMBER

VALUES('300','VAIBHAV','L','60','500');

INSERT INTO MEMBER

VALUES('400','GURUNATH','L','45','250');

update member

set Total_fine = NULL

```
where Mem_no = '400';
```

```
create table Trans(
```

```
Book_no varchar2(20),
```

```
Mem_no varchar2(20),
```

```
Issue_date date,
```

```
due_date date,
```

```
return_date date,
```

```
FOREIGN key (Book_no) REFERENCES book(Book_no),
```

```
FOREIGN key (Mem_no) REFERENCES
```

```
MEMBER(Mem_no)
```

```
);
```

```
insert INTO trans values('1','100',sysdate,sysdate+7, null);
```

```
insert INTO trans values('2','200',sysdate,sysdate+7, null);
```

```
insert INTO trans values('3','300',sysdate,sysdate+7, null);
```

```
insert INTO trans values('4','400',sysdate,sysdate+7, null);
```

```
SET SERVEROUTPUT ON;
```

```
create or replace procedure issue(book_add in varchar2 , mem_add in  
varchar2) IS
```

```
hasbook varchar2(20);checkday varchar2(20);
```

```
dateofmembership member.DATE_OF_MEMBERSHIP%type;
```

```
bookname book.book_name%type ;noofbooks
```

```
book.no_of_books%type;
```

```
membertype member.mem_type%type ;noofbooksm member.no_of_books%type;
```

```
-- Cursors are tused to get the needed values during proccessing
```

```
cursor book_aval is Select book_name,no_of_books from book where book_no = book_add;
```

```
cursor member_status is Select mem_type,no_of_books,DATE_OF_MEMBERSHIP from  
member where mem_no = mem_add ;
```

```
cursor check_book_status is Select mem_no from trans where book_no = book_add  
AND mem_no = mem_add;
```

```
BEGIN
```

```
select to_char(sysdate, 'DAY') into checkday from dual;
```

```
-- If Saturday or Sunday do issue
```

```
if checkday='Saturday' OR checkday='Sunday' then DBMS_OUTPUT.PUT_LINE('No Issue  
on Saturday And Sunday'); return;
```

```
end if ;
```

```
open book_aval ;
```

```
open member_status ;
```

```
open check_book_status ;
```

```

fetch book_aval into bookname , noofbooks ;
fetch member_status into membertype , noofbooksm,dateofmembership;
fetch check_book_status into hasbook ;
-- If cursor not found return as data entered is wrong
if book_aval%NOTFOUND THEN DBMS_OUTPUT.PUT_LINE('Book Not Found Enter
Correct Name'); return;
elsif
member_status%NOTFOUND THEN DBMS_OUTPUT.PUT_LINE('Member Not Found
Enter Correct Name');return ;
elsif
check_book_status%FOUND then DBMS_OUTPUT.PUT_LINE('Same Book Cannot be
Taken Again without returning');return ;
end if;
if
-- Check if number of books is never violated OR book is available in stock
noofbooks=0 THEN DBMS_OUTPUT.PUT_LINE('Book Out Of Stock') ; return ;
else CASE
When membertype='Monthly' AND noofbooksm ='4' then
DBMS_OUTPUT.PUT_LINE('Monthy Customer cannot issue more than 4 books');return ;
When membertype='Yearly' AND noofbooksm ='2' then DBMS_OUTPUT.PUT_LINE('Yearly
Customer cannot issue more than 2 books');return ;
When membertype='Lifetime' AND noofbooksm ='6' then
DBMS_OUTPUT.PUT_LINE('Lifetime Customer cannot issue more than 6 books');return ;
else DBMS_OUTPUT.PUT_LINE("");
END CASE;
end if ;
-- Check if users validity is not past the return date
case
when membertype='Monthly' AND dateofmembership + INTERVAL '30' DAY <
sysdate+INTERVAL '7' DAY THEN DBMS_OUTPUT.PUT_LINE('You membership will end
so please restore it before lending'); return;

when membertype='Yearly' AND dateofmembership +INTERVAL '365' DAY <
sysdate+INTERVAL '7' DAY THEN DBMS_OUTPUT.PUT_LINE('You membership will end
so please restore it before lending'); return;
else DBMS_OUTPUT.PUT_LINE("");
END CASE;
-- If al; constraints are checked populate trans table and update member details and book
stock accordingly
insert into trans(Book_no , Mem_no,return_date) VALUES (book_add , mem_add ,null);
update member
set no_of_books = noofbooksm+1
where mem_no = mem_add;
update book
set no_of_books = noofbooks-1
where book_no = book_add;
close book_aval;

```

```
close member_status;  
close check_book_status;  
<<ends>>  
DBMS_OUTPUT.PUT_LINE('Everthing is done ');  
END;  
/
```