

**American International University – Bangladesh**

**Course Name: Introduction to Database [A]**

**Course Code: CSC 2107**

**Project Final Report**

**Library Management System**

##### Submitted To: KAWSRE IROM RUSHEE

**Submitted By :**

|  |  |
| --- | --- |
| **Name** | **ID** |
| **ISLAM, MD.NAYMUL** | **15-29950-2** |
| **ALOM, MD.IFTEKAR** | **15-29633-2** |
| **JAWAD, MD.AL ABID** | **15-29859-2** |

|  |  |
| --- | --- |
| **No** | **Content** |
| 01 | **Rule** |
| 02 | **Requirements** |
| 03 | **Integrity Constraints** |
| 04 | **ERD Legend** |
| 05 | **Report** |
| 06 | **Normalization of Database and Functional Dependencies** |
| 07 | **Tables Screen Shot With Data( Table Should Include Constraints)** |
| 08 | **4 View Creation Screen Shot With Data(2 Simple + 2 Complex View)** |
| 09 | **Query With Data Screen Shot(4 Joining Query,4 Sub Query(2 Using Group Function))** |
| 10 | **2 User Creation Query With Screenshot** |
| 11 | **Enable ,Disable Constraints of Table With Screenshot** |

**RULES**

This project of **“ LIBRARY MANAGEMENT SYSTEM”** gives us the complete information about the library. We can enter the record of new books and retrieve the details of books available in the library. We can issue the books to the students and maintain their records and can also check how many books are issued and stock available in the library. In this project we can maintain the late fine of students who returns the issued books after the due date.

This system is like that BOOKS are written by AUTHOR then those books are published by PUBLISHER then library collect those books.While collecting every books are putted in a CATEGORY.After that MEMBERS borrow those books from library.While borrowing a ISSUE STATUS will create. From this discussion, we have identified the data entities required in the data model for this new library management system AUTHOR, BOOK, PUBLISHER, MEMBER, ISSUE STATUS, CATEGORY and ADDRESS of MEMBER.

**Requirements**

Concerning the BOOK:

1. A BOOK has a unique BOOK\_ID,NAME,EDITION and a ISBN number.
2. One BOOK may be issued by exactly one MEMBER.
3. One BOOK may be written by many AUTHOR.
4. One BOOK may be published by many PUBLISHER.
5. One BOOK has exactly one CATEGORY.
6. We track MEMBER for every ISSUE.

Concerning the MEMBER:

1. A MEMBER has a unique MEMBER\_ID,NAME,ADDRESS.
2. A MEMBER can be issue many BOOKS.
3. A MEMBER may have exactly one ISSUE\_STATUS.
4. We track MEMBER\_ID for every ISSUE.

Concerning the ADDRESS:

1. Each ADDRESS has HOUSE\_NO,STREET and CITY.
2. We track ADDRESS for every MEMBER

Concerning the ISSUE\_STATUS:

1. A ISSUE\_STATUS has a uniqe STATUS\_NO,ISSUE\_DATE and RETURN\_DATE.
2. A ISSUE\_STATUS may be issued by exactly one MEMBER.
3. We track ISSUE\_STATUS for every ISSUE.

Concerning the AUTHOR:

1. An AUTHOR has a unique AUTHOR\_NO,NAME and QUALIFICATION.
2. An AUTHOR may write many BOOKS.
3. We track AUTHOR for every BOOKS.

Concerning the PUBLISHER:

1. A PUBLISHER has a unique PUBLISH\_ID,NAME,E-MAIL and PHONE\_NO.
2. A PUBLISHER may publish many BOOKS.
3. We track PUBLISHER for every BOOKS.

Concerning the CATEGORY:

1. A CATEGORY has a unique CATEGORY\_NO and CATEGORY\_NAME.
2. Many BOOKS may be store in one CATEGORY.
3. We track CATEGORY for every BOOK.

**Integrity Constraints**

* One BOOK may be issued by exactly one MEMBER.

One book may borrowed by exactly one member at a time however one member can borrow many books at a time.

* A ISSUE\_STATUS may be issued by exactly one MEMBER.

One member has exactly one issue status at a time and a issue status is exactly for one member at a time.System also stores issue number,issue date,and return issue .

* An AUTHOR may write many BOOKS.

One author may write many books and one book may written by many author.

System stores information of those authors who has written at least one book stored in the library. To identify an author the system stores author ID along with author name.

* A PUBLISHER may publish many BOOKS.

A publisher may publish many books and one books may published by many publisher.

For example Data Structure and Programming Design book have many publisher like Digital Libraries. MIT Press, ISBN 0-262-01180-8, 2000. It also publish Large-Scale Numerical Optimization. SIAM, 1990. (With Yuying Li)

* Many BOOKS may be store in one CATEGORY.

One book has exactly one category however many books may be stored in one category.

A book has exactly one category and for a category there must be at least one book. Each category has a name and the unique property of each category is a category number.

**Report**

Library Management System is an application which refers to library systems which are generally small or medium in size. It is used by librarian to manage the library using a computerized system where he/she can record various transactions like issue of books, return of books, addition of new books, addition of new students etc.

Books and student maintenancemodules are also included in this system which would keep track of the students using the library and also a detailed description about the books a library contains. With this computerized system there will be no loss of book record or member record whichgenerally happens when a non computerized system is used.

In addition, report module is also included in Library Management System. If user’s position is admin, the user is able to generate different kinds of reports like lists of students registered, list

of books, issue and return reports.

All these modules are able to help librarian to manage the library with more convenience and in a more efficient way as compared to library systems which are not computerized.

Throughout the project the focus has been on presenting information and comments in an easy and intelligible manner. The project is very useful for those who want to know about Library Management System.

NORMALIZATION

UNF:- mid, mname, phone\_no,address ( house\_no, street\_no,city) cno, cname, ano, aname, qualification , pid, pname, pemail, pphone\_no, bid, bname, edition, isbn, sno, issue\_date, return\_date

1NF:-Member :- mid, mname, phone\_no, bid, house\_no

Address:- house\_no, street\_no,city

Category:- cno, cname

Author:-ano, aname, qualification

Publisher:-pid, pname, pemail, pphone\_no

Book:-bid, bname, edition, isbn, pid , ano, cno

Issue\_status :- sno, issue\_date, return\_date, mid

2NF:-Member :- mid, mname, phone\_no, bid, house\_id

Address:- house\_no, street\_no,city

Category:- cno, cname

Author:-ano, aname, qualification

Publisher:-pid, pname, pemail, pphone\_no

Book:-bid, bname, edition, isbn, pid , ano, cno

Issue\_status :- sno, issue\_date, return\_date, mid

3NF:-Member :- mid, mname, phone\_no, bid, house\_id

Address:- house\_no, street\_no,city

Category:- cno, cname

Author:-ano, aname, qualification

Publisher:-pid, pname, pemail, pphone\_no

Book:-bid, bname, edition, isbn, pid , ano, cno

Issue\_status :- sno, issue\_date, return\_date, mid

Table Creation

create table address(house\_no varchar(20) constraint p1 primary key,street\_no varchar(20),city varchar(20))

create table category(cno varchar(20) constraint p2 primary key,cname varchar(20))

create table author(ano varchar(20) constraint p3 primary key,aname varchar(20),qualification varchar(50))

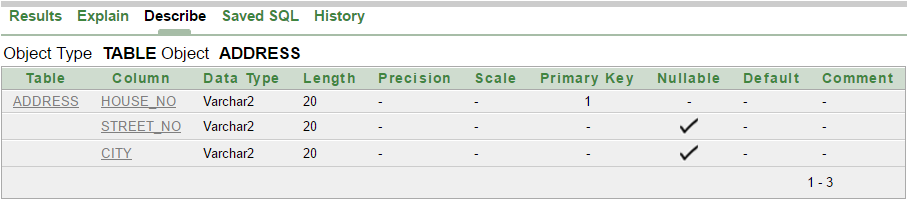
create table publisher(pid varchar(20) constraint p4 primary key,pname varchar(20),pemail varchar(20),pphone\_no number)

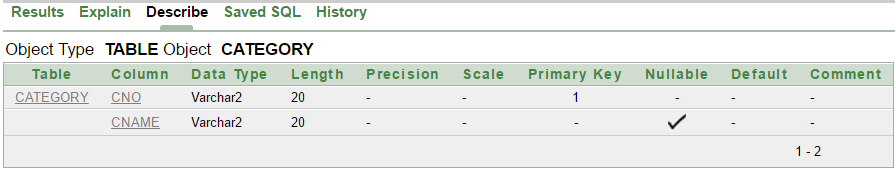
create table book(bid varchar(10) constraint p5 primary key,bname varchar(40),edition varchar(10),isbn varchar(30),pid varchar(20),ano varchar(20),cno varchar(20),constraint p6 foreign key(pid) references publisher(pid),constraint p7 foreign key(ano) references author(ano),constraint p8 foreign key(cno) references category(cno))

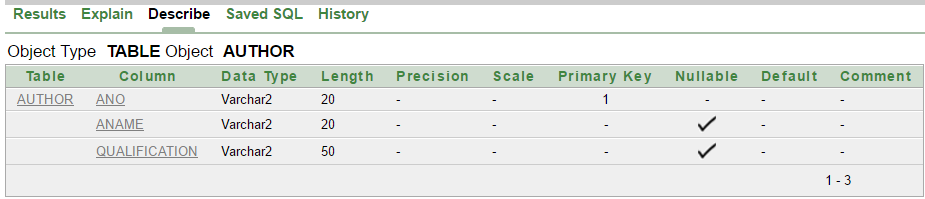
create table member(mid varchar(12) constraint p9 primary key,mname varchar(40),phone\_no number,bid varchar(10),house\_no varchar(20),constraint p10 foreign key(bid) references book(bid),constraint p11 foreign key(house\_no) references address(house\_no))

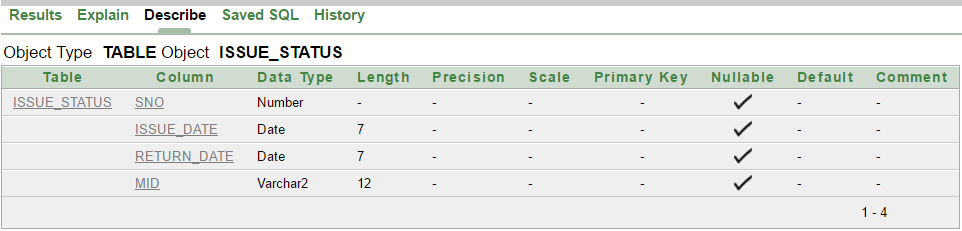
create table issue\_status(sno number,issue\_date date,return\_date date,mid varchar(12),constraint p12 foreign key(mid) references member(mid))

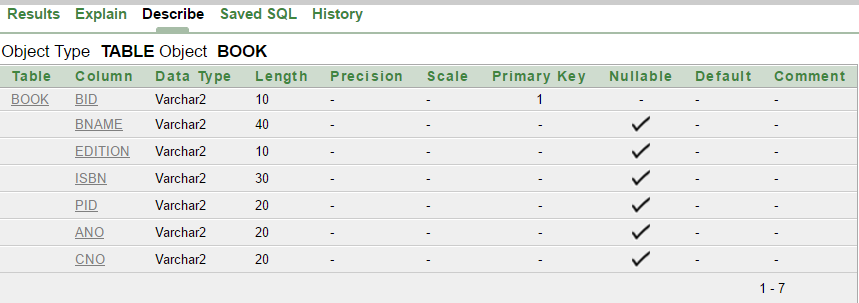
Table Description

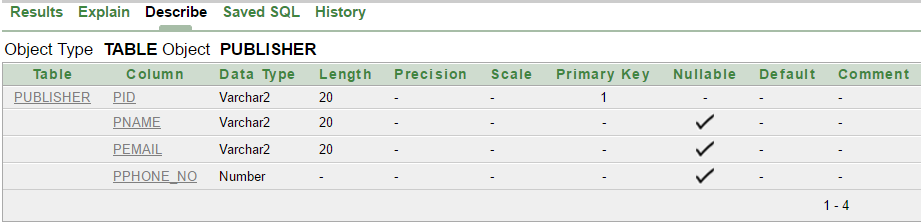












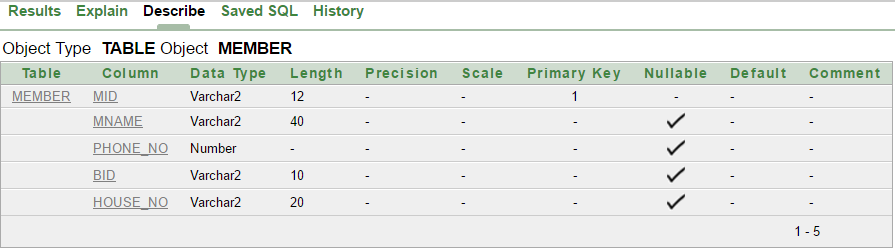
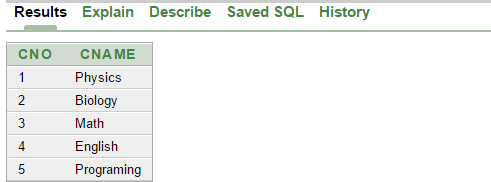


Table With Data

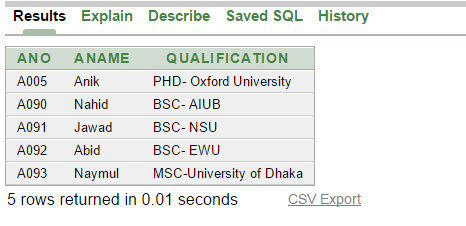
ADDRESS:



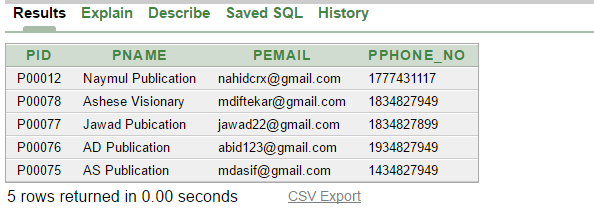
CATEGORY:



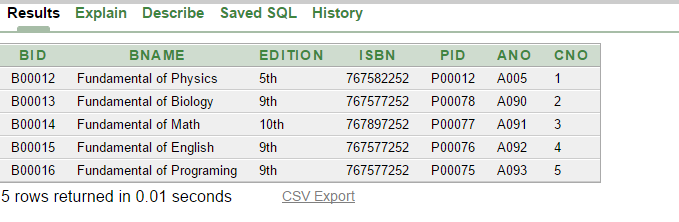
AUTHOR:



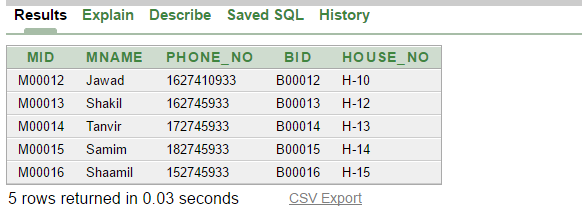
PUBLISHER:



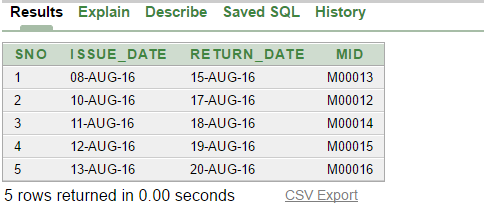
BOOK:



MEMBER:



ISSUE\_STATUS:



View Creation

create view mview(Member\_Id,Member\_Name,Book\_Id) as select mid,mname,bid from member

create view aview(Author\_Id,Author\_Name,Qualification) as select ano,aname,qualification from author

create view bview(Book\_ID,Book\_Name,Publisher\_ID,Publisher\_Name) as select b.bid,b.bname,p.pid,p.pname from book b,publisher p where b.pid=p.pid and p.pid in('P00012','P00075')

create view mmview(Member\_Name,Phone\_Number,City,Book\_Name,ISBN) as select m.mname,m.phone\_no,a.city,b.bname,b.isbn from member m,address a,book b where m.bid=b.bid and m.house\_no=a.house\_no and m.mname like 'S%'

select \* from mmview

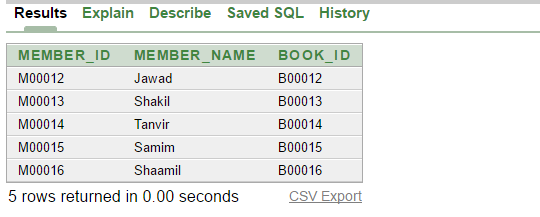
select \* from mview

select \* from aview

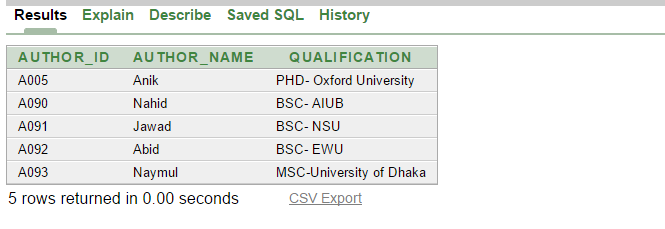
select \* from bview

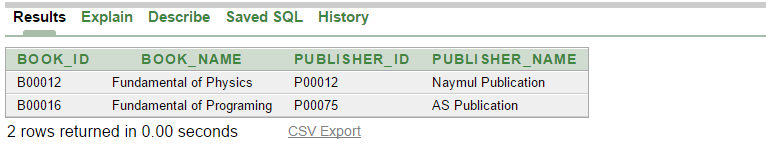
View Table

MVIEW:

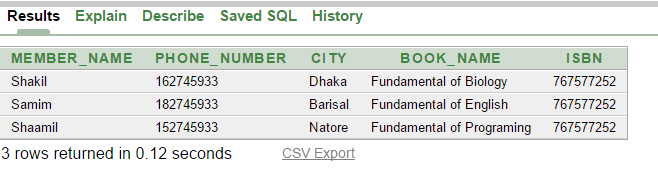


AVIEW:



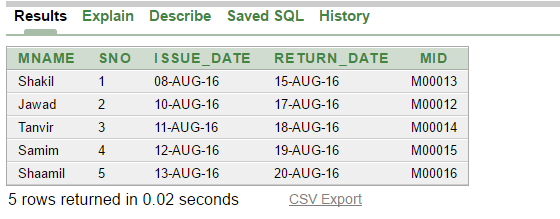
BVIEW:

MMVIE:

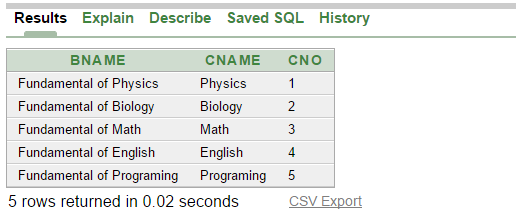


Joining

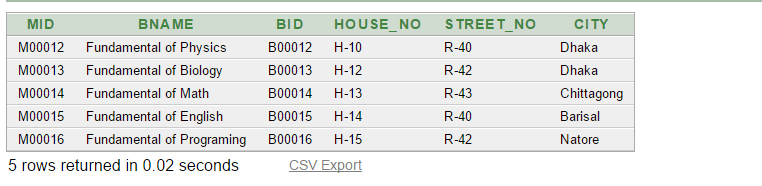
1. select m.mname,i.\* from member m,issue\_status i where m.mid=i.mid



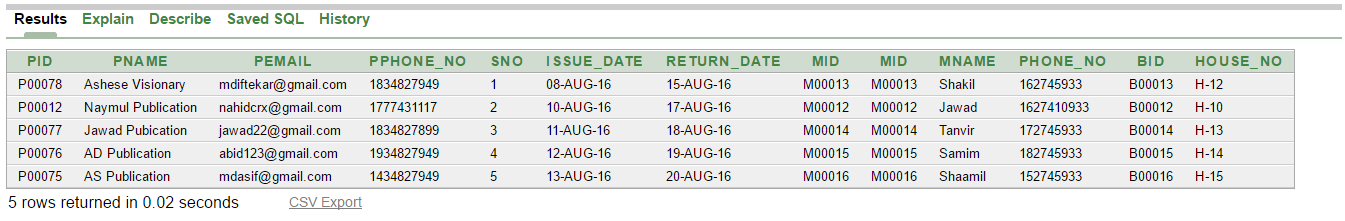
2. select b.bname,c.cname,c.cno from book b,category c where c.cno=b.cno



3. select m.mid,b.bname,b.bid,a.\* from member m,book b,address a where b.bid=m.bid and m.house\_no = a.house\_no

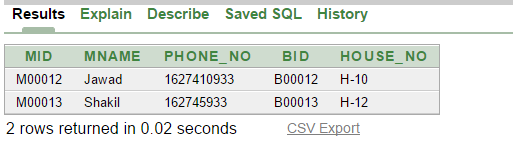


4. select p.\*,i.\*,m.\* from publisher p, issue\_status i, member m,book b where p.pid=b.pid and b.bid=m.bid and m.mid=i.mid

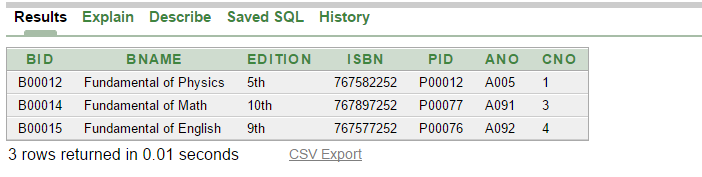


Sub Queries

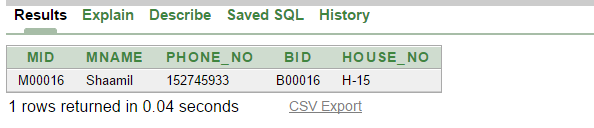
1. select \* from member where house\_no=Any(select house\_no from address where city='Dhaka')



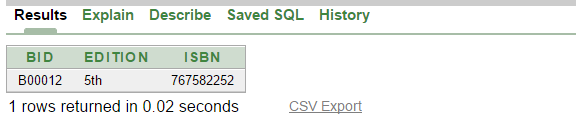
2. select \* from book where ano=any(select ano from author where aname in('Anik','Abid')) or cno=any(select cno from category where cno in(3,4))



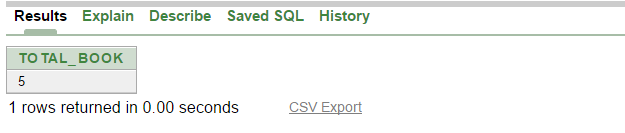
3. select \* from member where mid=(select mid from issue\_status where to\_char(issue\_date,'dd')='13')



4. select bid,edition,isbn from book where cno =(select cno from book where bname='Fundamental of Physics')

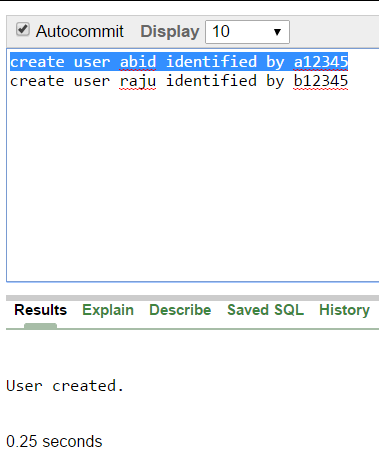


5. select count(bid) Total\_Book from book

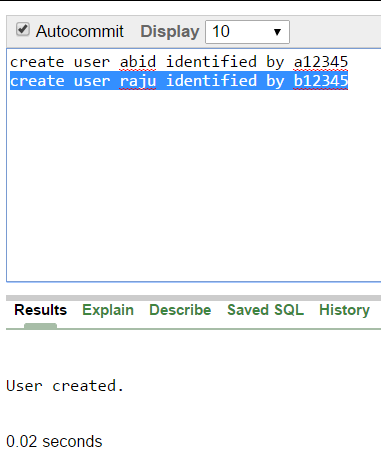


User Creation

1. create user abid identified by a12345

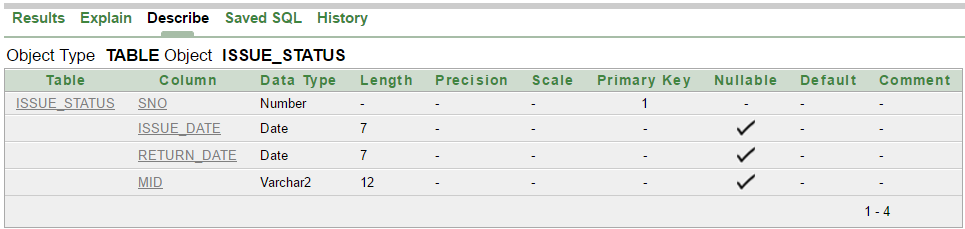


2 . create user raju identified by b12345



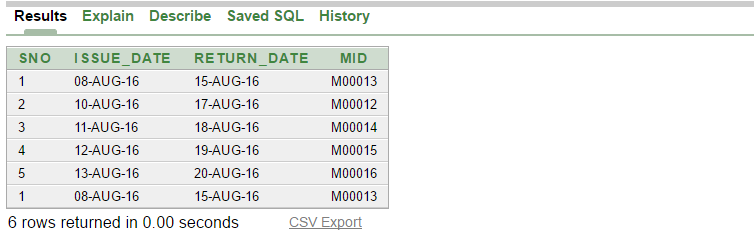
Add, Enable ,Disable Constraints

1. alter table issue\_status add constraint p20 primary key(sno)



2. alter table issue\_status disable constraint p20

insert into issue\_status(sno,issue\_date,return\_date,mid) values(0000001,'8-Aug-16','15-Aug-16','M00013')



3. alter table issue\_status enable constraint p20

insert into issue\_status(sno,issue\_date,return\_date,mid) values(0000001,'8-Aug-16','15-Aug-16','M00013')

