



DBMS PROJECT

LIBRARY MANAGEMENT SYSTEM

2020-21

SUBMITTED BY:-

U19CS067

JYOTI

U19CS082

SOURABH PATEL



INDEX

Sr. No	Topic	Page No.
1.	Topic ,Team members and index	1-2
2.	Problem statement ,description and constraint	3
3.	ER diagram	4
4.	Entities and their attributes	5-6
5.	Query and Normalizations	7-12
6.	Relationship Model	13
7.	Backend code and query	14-20
8.	plsql	21
9.	Website screen shots	22-30

PROBLEM STATEMENT :

- ☐ To design a library management system that lets user to issue different types of books online Named as C2me. C2me can used to issue different category of books. This system should also keep track of books record.

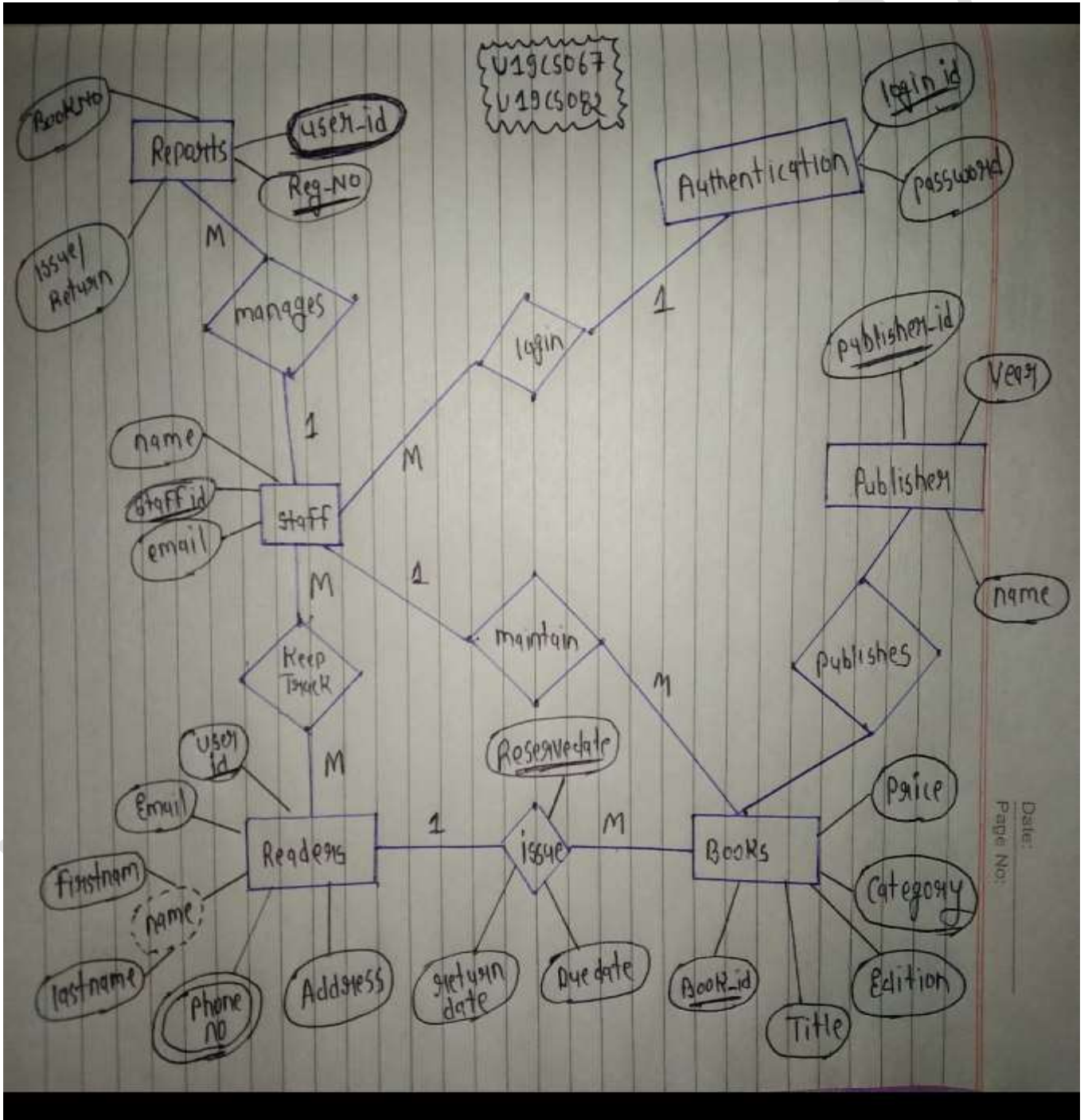
DISCREPTION :

A Library Management System is a software built to handle the primary housekeeping functions of a library. Library management systems also involve maintaining the database for entering new books and recording books that have been borrowed with their respective due dates.

☐ **CONSTRAINTS :**

- ☐ A Login id have one user.
- ☐ A User use many books.
- ☐ A book is used by one user.
- ☐ A user make many issue oprations.
- ☐ A issue operation is made for one book.
- ☐ A book list have many books.
- ☐ A book have one publisher.
- ☐ A publisher have many books.

□ ENTITY – RELATIONSHIP (E-R) DIAGRAM :



□ ENTITIES :-

There are total six entities in this system. All of Six have their own attribute, which acts as primary key. So all are strong attributes.

User	: Library information about User.
Books	: Library information about books.
Reports	: Library system made issue operation by user.
Publishers	: Library information about Publisher of books.
Staff	: Library staff information

ALL ENTITIES WITH ATTRIBUTES

□ user:-

1. fname
2. lname
3. mobileNumber
4. address
5. email **(PRIMARY KEY)**
6. password

□ book:-

1. bookid **(PRIMARY KEY)**
2. bookname
3. title
4. edition
5. bookcategory
6. price
7. publisher_email **FOREIGN KEY [REFERENCES publishers(email)]**

8. quantity
9. publishing_date

□ **reports:-**

1. reg_no
2. user_email **FOREIGN KEY [REFERENCES user(email)]**
3. issue_date
4. return_date
5. bookid **FOREIGN KEY [REFERENCES book(bookid)]**
6. **PRIMARY KEY**(reg_no)

□ **publishers:-**

1. fname
2. lname
3. email **(PRIMARY KEY)**
4. address
5. moblienumbr

□ **staff:-**

1. fname
2. lname
3. email **(PRIMARY KEY)**
4. address
5. moblienumbr
6. password

QUERY TO CREATE DATABASE:

```
CREATE DATABASE librarydatabase;
```

ALL DATABASES:

```
Command Prompt - mysql -u root -p
mysql> SHOW TABLES;
+-----+
| Tables_in_librarydatabase |
+-----+
| book                       |
| publishers                 |
| reports                   |
| staff                     |
| user                      |
+-----+
5 rows in set (0.00 sec)
```

QUERY TO CREATE USER TABLE:

```
CREATE TABLE user(
  fname varchar(30),
  lname varchar(30),
  moblieNumber varchar(25),
  address varchar(100),
  email varchar(50),
  password varchar(15),
  PRIMARY KEY (email));
```

OUTPUT OF USER TABLE:

```
Command Prompt - mysql -u root -p
mysql> desc user;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| fname      | varchar(30)   | YES  |     | NULL    |       |
| lname      | varchar(30)   | YES  |     | NULL    |       |
| moblieNumber | varchar(25)   | YES  |     | NULL    |       |
| address     | varchar(100)  | YES  |     | NULL    |       |
| email       | varchar(50)   | NO   | PRI | NULL    |       |
| password    | varchar(15)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> _
```

```
Command Prompt - mysql - console
Select Command Prompt - mysql -u root -p
mysql> SELECT * FROM user;
```

fname	lname	moblieNumber	address	email	password
sourabh	patel	9691619115	mp	exmp@gmail.com	12345
qddhbqd	Doejdedn	NULL	madhaya pradesh	gurjarsourabh0802@gmail.com	322233444
hhguyguihihiuhh	patel	NULL	madhaya pradesh	jyotibabroli2003@gmail.com	322233444
sourabh	patel	5	madhaya pradesh	pso@gmail.com	sourabh@111
sourabh82	patel	9691619115	bhopal mp	psourabh080299@gmail.com	30003000

```
6 rows in set (0.02 sec)
mysql>
```

NORMALIZATION USER TABLE:

- **FUNCTIONAL DEPENDENCIES**

email->fname,lname,moblieNumber,address,password

Candidate Keys:email

Primary Key:email

- **1nf**->no multivalued attribute
- **2nf**->single attribute primary key(i.e. email) all no-prime attribute are fully dependent on candidate key.so user table is in 2nf
- **3nf and bcnf**->for every non-trival functional dependency X->Y
 - X is a super key

QUERY TO CREATE BOOK TABLE:

```
CREATE TABLE book(
bookid varchar(154),
bookname varchar(30),
title varchar(30),
edition int(100),
bookcategory varchar(30),
price varchar(1628),
publisher_email varchar(255),
quantity varchar(255),
publishing_date date,
language varchar(30),
PRIMARY KEY (bookid));
```

OUTPUT OF USER TABLE


```

Command Prompt - mysql -u root -p
mysql> desc book;
+-----+-----+-----+-----+-----+-----+
| Field                | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| bookid               | varchar(154)  | NO   | PRI | NULL    |       |
| bookname              | varchar(30)   | YES  |     | NULL    |       |
| title                | varchar(30)   | YES  |     | NULL    |       |
| edition              | int           | YES  |     | NULL    |       |
| bookcategory          | varchar(30)   | YES  |     | NULL    |       |
| price                 | varchar(1628) | YES  |     | NULL    |       |
| publisher_email       | varchar(255)  | YES  |     | NULL    |       |
| quantity              | varchar(255)  | YES  |     | NULL    |       |
| publishing_date       | date          | YES  |     | NULL    |       |
| language              | varchar(30)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>

```

```

Command Prompt - mysql -u root -p
mysql> SELECT * FROM book;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| bookid | bookname          | title                | edition | bookcategory | price | publisher_email | quantity | publishing_date | language |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1      | Plane Trigonometry | maths                | 2       | JEE MAINS    | 299 RS | NULL            | 6        | 2021-07-02     | english  |
| 10     | introduction to dbms | dbms                 | 1       | btech        | 1199 RS | NULL            | 8        | 2021-05-09     | english  |
| 11     | Differential Calculus | maths                | 1       | JEE ADVANCED | 599 RS | NULL            | 7        | 2019-07-02     | english  |
| 12     | 41 Years Paper      | maths                | 1       | JEE ADVANCED | 599 RS | NULL            | 9        | 2018-07-02     | english  |
| 13     | higher mathematics  | maths                | 1       | JEE ADVANCED | 699 RS | NULL            | 9        | 2011-07-02     | english  |
| 14     | concept of plsql    | plsql                | 1       | btech        | 599 RS | NULL            | 6        | 2014-07-02     | english  |
| 2      | Concept of physics  | Physics              | 3       | JEE MAINS    | 399 RS | NULL            | 10       | 2020-07-02     | english  |
| 3      | newclres            | data structure and algo | 11      | DATA STRUCTURE | 999 RS | NULL            | 8        | 2021-04-17     | english  |
| 4      | introduction to ds   | data structure       | 6       | DS           | 999 RS | NULL            | 5        | 2021-04-07     | hindi    |
| 5      | introduction to ds   | data structure       | 10      | DATA STRUCTURE | 499 RS | NULL            | 11       | 2021-04-07     | english  |
| 6      | introduction to ds   | data structure       | 3       | ALGO ANALYSIS | 599 RS | NULL            | 11       | 2021-04-29     | hindi    |
| 7      | bk jeswal            | chemistry            | 3       | JEE ADVANCED | 399 RS | NULL            | 5        | 2021-04-06     | hindi    |
| 8      | 12th Physics        | Physics              | 1       | NCERT        | 150 RS | NULL            | 11       | 2021-05-09     | english  |
| 9      | sspatel             | maths                | 3       | NCERT        | 999 RS | NULL            | 2        | 2021-05-01     | hindi    |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
14 rows in set (0.01 sec)

mysql>

```

NORMALIZATION USER TABLE:

- **FUNCTIONAL DEPENDENCIES**

bookid->bookname,title,edition ,bookcategory,price
publisher_email,quantity,publishing_date,language

Candidate Keys:bookid

Primary Key:bookid

- **1nf**->no multivalued attribute
- **2nf**->single attribute primary key(i.e. bookid) all no-prime attribute are fully dependent on candidate key.so user table is in 2nf
- **3nf and bcnf**->for every non-trival functional dependency X->Y
 - X is a super key

QUERY TO CREATE REPORTS TABLE:

```
CREATE TABLE reports(  
  Reg_no int NOT NULL AUTO_INCREMENT,  
  User_email varchar(50),  
  issue_date date,  
  return_date date,  
  bookid varchar(154),  
  PRIMARY KEY (reg_no)  
);
```

OUTPUT OF REPORTS TABLE:

Command Prompt - mysql -u root -p

```
mysql> desc reports;
```

Field	Type	Null	Key	Default	Extra
reg_no	int	NO	PRI	NULL	auto_increment
user_email	varchar(50)	YES	MUL	NULL	
issue_date	date	YES		NULL	
return_date	date	YES		NULL	
bookid	varchar(154)	YES		NULL	

5 rows in set (0.00 sec)

```
mysql>
```

Command Prompt - mysql -u root -p

```
mysql> SELECT * FROM reports;
```

reg_no	user_email	issue_date	return_date	bookid
2	exmp@gmail.com	2021-04-19	2021-04-23	3
3	exmp@gmail.com	2021-04-19	2021-04-23	3
4	exmp@gmail.com	2021-04-19	2021-04-23	3
5	exmp@gmail.com	2021-04-19	2021-04-23	3
11	gurjarsourabh0802@gmail.com	2021-04-02	2021-05-02	7
12	gurjarsourabh0802@gmail.com	2021-04-23	2023-06-23	3
13	gurjarsourabh0802@gmail.com	2021-04-24	2021-05-02	3
14	psourabh080299@gmail.com	2021-04-15	2021-05-02	10
15	psourabh080299@gmail.com	2021-04-07	2021-04-27	4
16	psourabh080299@gmail.com	2021-04-28	2021-05-09	11
17	psourabh080299@gmail.com	2021-04-26	2021-05-07	14

11 rows in set (0.01 sec)

```
mysql>
```

NORMALIZATION USER TABLE:

- FUNCTIONAL DEPENDENCIES

Reg_no->user_email,issue_date,return_date,bookid

Candidate Keys:reg_no

Primary Key:reg_no

- **1nf**->no multivalued attribute
- **2nf**->single attribute primary key(i.e. reg_no) all no-prime attribute are fully dependent on candidate key.so user table is in 2nf
- **3nf and bcnf**->for every non-trivial functional dependency $X \rightarrow Y$
 - X is a super key

QUERY TO CREATE PUBLISHER TABLE:

```
CREATE TABLE publishers(  
  fname varchar(30),  
  lname varchar(30),  
  email varchar(50),  
  address varchar(100),  
  moblieNumber varchar(15),  
  PRIMARY KEY (email)  
);
```

OUTPUT OF PUBLISHERS TABLE:

Command Prompt - mysql -u root -p

mysql> desc publishers;

Field	Type	Null	Key	Default	Extra
fname	varchar(30)	YES		NULL	
lname	varchar(30)	YES		NULL	
email	varchar(50)	NO	PRI	NULL	
address	varchar(100)	YES		NULL	
moblienummer	varchar(15)	YES		NULL	

5 rows in set (0.01 sec)

mysql> _

NORMALIZATION USER TABLE:

- **FUNCTIONAL DEPENDENCIES**
email->fname,lname,address,moblieNumber

Candidate Keys:email

Primary Key:email

- **1nf**->no multivalued attribute
- **2nf**->single attribute primary key(i.e. email) all no-prime attribute are fully

dependent on candidate key.so user table is in 2nf

- **3nf and bcnf**->for every non-trivial functional dependency $X \rightarrow Y$
 - X is a super key

QUERY TO CREATE STAFF TABLE:

```
CREATE TABLE staff(  
  fname varchar(30),  
  lname varchar(30),  
  email varchar(50),  
  address varchar(100),  
  moblieNumber varchar(15),  
  password varchar(20),  
  PRIMARY KEY (email));
```

OUTPUT OF SAFF TABLE

```
Command Prompt - mysql -u root -p  
mysql> desc staff;  
+-----+-----+-----+-----+-----+-----+  
| Field      | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| fname      | varchar(30)   | YES  |     | NULL    |       |  
| lname      | varchar(30)   | YES  |     | NULL    |       |  
| email      | varchar(50)   | NO   | PRI | NULL    |       |  
| address    | varchar(100)  | YES  |     | NULL    |       |  
| moblienumber | varchar(15)   | YES  |     | NULL    |       |  
| pasword    | varchar(20)   | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
6 rows in set (0.00 sec)
```

NORMALIZATION USER TABLE:

- **FUNCTIONAL DEPENDENCIES**
email->fname,lname,address,moblieNumber,password
Candidate Keys:email
Primary Key:email
- **1nf**->no multivalued attribute
- **2nf**->single attribute primary key(i.e. email) all no-prime attribute are fully dependent on candidate key.so user table is in 2nf
- **3nf and bcnf**->for every non-trivial functional dependency $X \rightarrow Y$
 - X is a super key

RELATIONAL-MODEL

	ATTRIBUTES	DATA TYPE
	fname	varchar(30)
	lname	varchar(30)
	moblieNumber	varchar(25)
	address	varchar(100)
PR	email	varchar(50)
	password	varchar(15)

USER TABLE

	ATTRIBUTES	DATA TYPE
PR	Reg_no	int
FK	User_email	varchar(50)
	Issue_date	date
	Return_date	date
FK	bookid	varchar(154)

REPORTS TABLE

	ATTRIBUTES	DATA TYPE
	fname	varchar(30)
	lname	varchar(30)
PR	email	varchar(25)
	address	varchar(100)
	moblieNumber	varchar(15)

PUBLISHERS TABLE

	ATTRIBUTES	DATA TYPE
PR	bookid	varchar(154)
	bookname	varchar(30)
	title	varchar(30)
	edition	int
	bookcategory	varchar(30)
	price	varchar(1628)
FK	Publisher_email	varchar(255)
	quantity	varchar(255)
	Publishing_date	date
	language	varchar(30)

BOOK TABLE

BACKEND QUERY:

```
const express=require('express');
const mysql=require('mysql');
const bodyParser=require('body-parser');
const cors=require('cors');
const app=express();
app.use(cors());
app.use(express.json());
app.use(bodyParser.urlencoded({extended: true}));

//database creation
const db=mysql.createConnection({
  host:'localhost',
  user:'root',
  password:'sourabh',
  database:'LibraryDatabase'
});

//all user list for admin
app.get('/user',(req,res)=>{
  const sqlSelect="SELECT * FROM user;"
  db.query(sqlSelect,(err,result)=>{
    res.send(result);
  });
})

//signup for user
app.post('/signup',(req,res)=>{
  const sqlInsert="INSERT INTO user (fname,lname,moblieNumber,address,email,password) VALUES (?, ?, ?, ?, ?, ?);"
  const fname=req.body.fname;
  const lname=req.body.lname;
  const moblieNumber=req.body.moblieNumber;
  const address=req.body.address;
  const email=req.body.email;
  const password=req.body.password;
  db.query(sqlInsert,[fname,lname,moblieNumber,address,email,password],[err,result]=>{
    if(err){
      console.log(err);
    }
    else{
      console.log('signup data inserted into user table');
```

```

    }
  })
})

//signup admin
app.post('/signup/admin',(req,res)=>{
  const sqlInsert="INSERT INTO staff (fname,lname,moblieNumber,address,email,password) VA
LUES (?,? ,? ,? ,? ,? );"
  const fname=req.body.fname;
  const lname=req.body.lname;
  const moblieNumber=req.body.moblieNumber;
  const address=req.body.address;
  const email=req.body.email;
  const password=req.body.password;
  db.query(sqlInsert,[fname,lname,moblieNumber,address,email,password] ,(err,result)=>{
    if(err){
      console.log(err);
    }
    else{
      console.log('signup data inserted into staff table');
    }
  })
})

//delete book by admin
app.delete('/delete/book:bookid',(req,res)=>{
  const sqlDelete="DELETE FROM book WHERE bookid = ? ;";
  const bookid=req.params.bookid;
  db.query(sqlDelete,bookid,(err,result)=>{
    if(err){
      console.log(err);
    }
    else{
      console.log('book deleted');
    }
  });
})

//update book by admin
app.put('/update/book',(req,res)=>{

  const sqlUpdate="UPDATE book SET  bookname = ?,title = ? ,edition = ? ,bookcategory = ?
,price = ?,publisher_email = ? , publishing_date = ? ,quantity = ?  WHERE bookid = ?";

```

```

const bookid=req.body.bookid;
const bookname=req.body.bookname;
const title=req.body.title;
const edition=req.body.edition;
const bookcategory=req.body.bookcategory;
const price=req.body.price;
const publisheremail=req.body.publisheremail;
const quantity=req.body.quantity;
const publishing_date=req.body.publishing_date;
db.query(sqlUpdate,[bookname,title,edition,bookcategory,price,publisheremail,publishing_date,quantity,bookid],(err,result)=>{
    if(err){
        console.log(err);
    }
    else{
        console.log('book updated');
    }
});
})

//get book detials for both user and admin
app.get('/book',(req,res)=>{
    const sqlSelect="SELECT * FROM book;"
    db.query(sqlSelect,(err,result)=>{
        res.send(result);
    });
})

//reports of issued books for admin
app.get('/issuedbookforadmin',(req,res)=>{
    const sqlSelect="SELECT * FROM reports;"
    db.query(sqlSelect,(err,result)=>{
        res.send(result);
    });
})

//add new book by admin
app.post('/book',(req,res)=>{
    const sqlInsert="INSERT INTO book (bookid,bookname,title,edition,bookcategory,price,publishing_date,quantity) VALUES (?, ?, ?, ?, ?, ?, ?, ?);"
    const bookid=req.body.bookid;
    const bookname=req.body.bookname;
    const title=req.body.title;

```



```

const edition=req.body.edition;
const bookcategory=req.body.bookcategory;
const price=req.body.price;
const quantity=req.body.quantity;
const publishing_date=req.body.publishing_date;
console.log('!!!!!!!!!!!!1111111',bookcategory)
db.query(sqlInsert,[bookid,bookname,title,edition,bookcategory,price,publishing_date,quantity],(err)=>{
    if(err){
        console.log(err);
    }
    else{
        console.log('your book inserted successfully');
    }
})
})

// issue a book by user
app.put('/issue/book',(req,res)=>{
    console.log('enter');
    const bookid=req.body.bookid;
    const sqlSelect="SELECT quantity FROM book WHERE bookid=?";
    db.query(sqlSelect,[bookid],(err,result)=>{
        if(result[0].quantity>0){
            const issue_date=req.body.issue_date;
            const return_date=req.body.return_date;
            const email=req.body.email;
            const sqlInsert="INSERT INTO reports (user_email,issue_date,return_date,bookid)
VALUES (?, ?, ?, ?);"
            db.query(sqlInsert,[email,issue_date,return_date,bookid],(err)=>{
                if(err){
                    console.log(err);
                }
                else{
                    console.log('data inserted into report');
                }
            });
            const sqlUpdate="UPDATE book SET quantity = ? WHERE bookid = ?";
            const quantity=result[0].quantity-req.body.quantity;
            db.query(sqlUpdate,[quantity,bookid],(err)=>{
                if(err){
                    console.log(err);
                }
                else{
                    console.log('your book is issued completly');
                }
            });
        }
    });
});

```

```

    }
  });
}
});

//all issued booklist for user
app.get('/book/issued:email',(req,res)=>{
  const email=req.params.email;
  const sqlSelect=" SELECT DISTINCT b.*,r.* FROM book b,reports r WHERE r.bookid=b.bookid
AND ( r.bookid IN (SELECT bookid FROM reports  WHERE user_email= ?) AND r.reg_no IN (SELEC
T reg_no FROM reports  WHERE user_email= ?));"
  db.query(sqlSelect,[email,email],[err,result]=>{
    res.send(result);
  });
})

//return book by user
app.put('/book/return',(req,res)=>{
  const email=req.body.email;
  const reg_no=req.body.reg_no;
  const sqlDelete="DELETE FROM reports WHERE (reg_no= ? AND user_email =?) ;"
  db.query(sqlDelete,[reg_no,email],[err,result]=>{
    if(err){
      console.log(err);
    }
    else{
      console.log('your book is submitted successfully');
    }
  });
})

//update book renue date by user
app.put('/book/renue',(req,res)=>{
  const email=req.body.email;
  const reg_no=req.body.reg_no;
  const renuedate=req.body.newrenudate.fname;
  const sqlUpdate="UPDATE reports SET return_date=?  WHERE (reg_no= ? AND user_email = ?)
;"
  db.query(sqlUpdate,[renuedate,reg_no,email],[err,result]=>{
    if(err){
      console.log(err);
    }
  });
})

```

```

    }
    else{
        console.log('your book return date is successfully updated',renuedate,"...",reg
_no,"...",email);
    }
});
})

```

//category of books for user

```

app.post('/user/category',(req,res)=>{
    const sqlSelect='SELECT DISTINCT bookcategory from book;'
    db.query(sqlSelect,(err,result)=>{
        if(err){
            console.log(err);
        }
        else{
            res.send(result)
            console.log("category are sent to user");
        }
    })
})

```

//get book by category

```

app.post('/getbookbycategory',(req,res)=>{
    const bookcategory=req.body.bookcategory;
    const sqlSelect='SELECT * FROM book WHERE bookcategory= ?;'
    db.query(sqlSelect,bookcategory,(err,result)=>{
        if(err){
            console.log(err)
        }
        else{
            res.send(result)
        }
    })
})

```

//get book by id

```

app.post('/getbookbyid',(req,res)=>{
    const bookid=req.body.bookid;
    const sqlSelect='SELECT * FROM book WHERE bookid= ?;'
    db.query(sqlSelect,bookid,(err,result)=>{
        if(err){
            console.log(err)
        }
    })
})

```

```

    }
    else{
        res.send(result)
    }
})
})

//latest books
app.get('/latestbook',(req,res)=>{
    const sqlSelect='SELECT * FROM book LIMIT 18;'
    db.query(sqlSelect,(err,result)=>{
        if(err){
            console.log(err)
        }
        else{
            console.log(result)
            res.send(result)
        }
    })
})

//search book by user
app.post('/user/search',(req,res)=>{
    const sqlSelect='SELECT * FROM book WHERE bookcategory = ? AND title = ?;'
    const bookcategory=req.body.bookcategory;
    const auther=req.body.auther;
    const title=req.body.title;
    const language=req.body.language;
    db.query(sqlSelect,[bookcategory,title],(err,result)=>{
        if(err){
            console.log(err)
        }
        else{
            res.send(result)
            console.log('done.....',result,"..",bookcategory,"...
",title)
        }
    })
})

// for port
app.listen(3001,()=>{
    console.log('running in 3001.....');});

```

PL/SQL:

○ FUNCTION

This function is used to tell the missing due date to the user according to the due date of issued books;

```
CREATE FUNCTION check_missing_date(return_date DATE)
RETURNS VARCHAR(4)
BEGIN
DECLARE missing VARCHAR(4);
IF year(curdate()) > year(return_date) THEN
SET missing = 'Yes';
ELSE
SET missing = 'No';
END IF;
RETURN (missing);
END
```

○ PROCEDURES

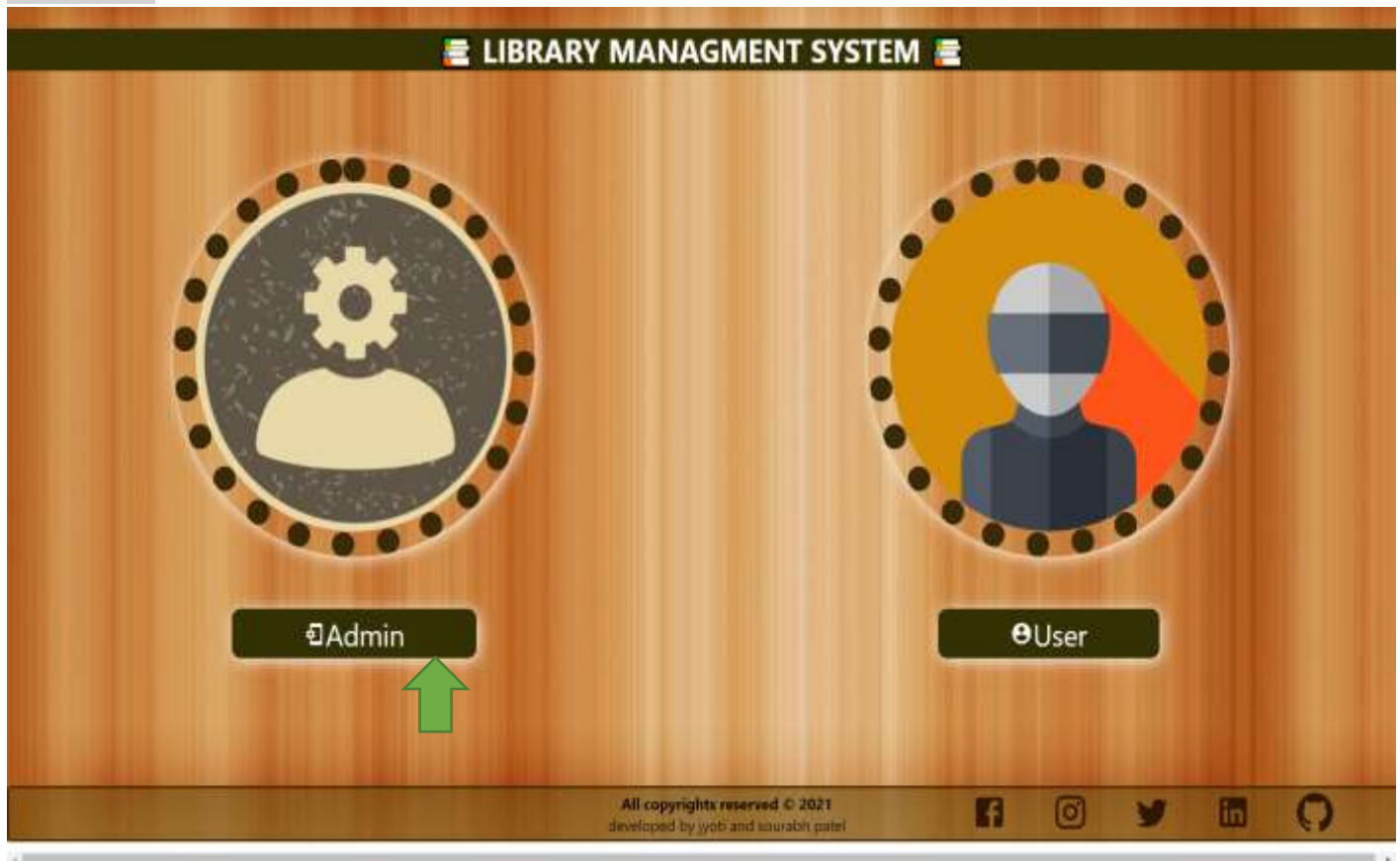
This procedure is used for updating entries in vaccinates table whenever a patient is given a dose of vaccine.

```
CREATE PROCEDURE update_reports(IN return_date DATE,IN new_return_date DATE,IN bookID
varchar)
BEGIN
update reports SET return_date = new_return_date where
bookid=bookID;
END
```

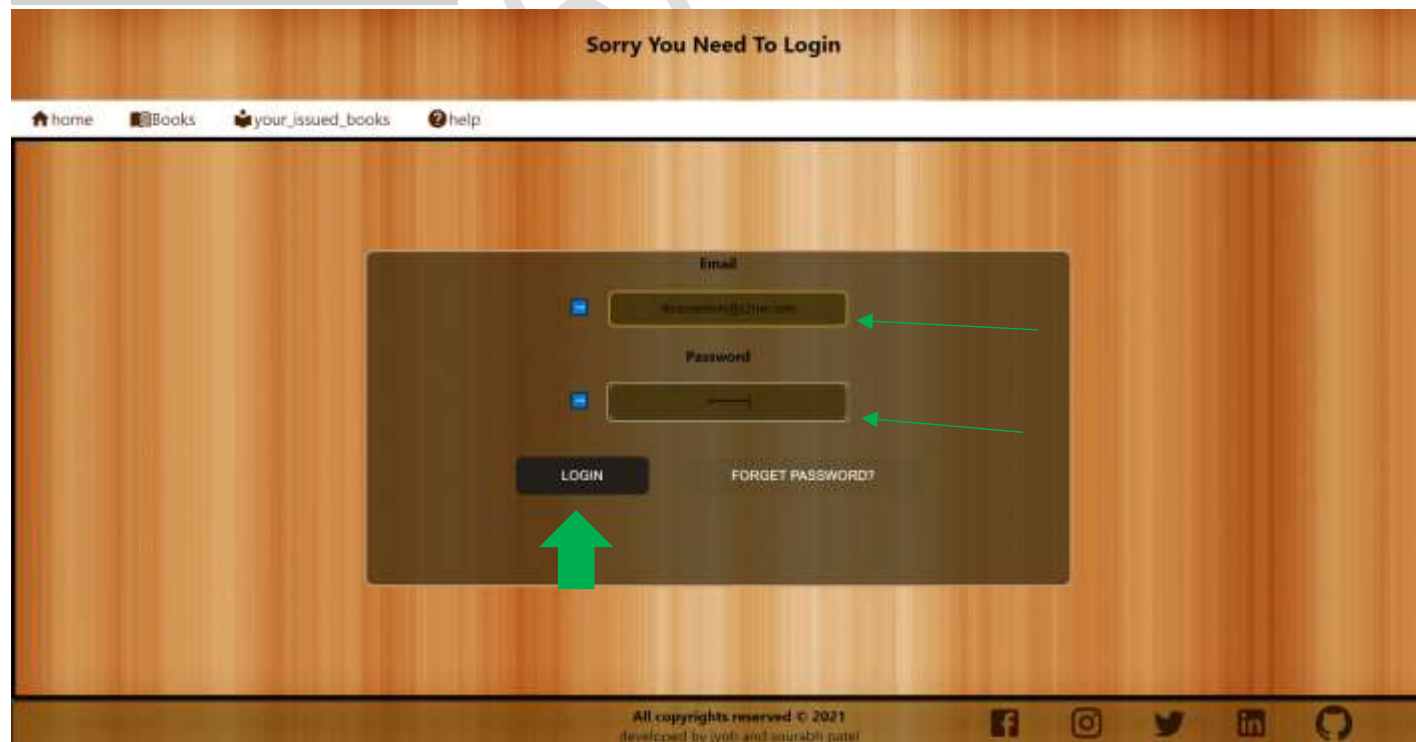
VIEWS

```
CREATE VIEW book_data AS SELECT bookid, bookname, bookcategory, edition, title,
price, publisher_email,quantity,language FROM book;
```

HOME:



ADMIN LOGIN PAGE:



IF PASSWORD WRONG:



ADMIN HOME PAGE:



ADMIN BOOKS PAGE:

All books in library | [add new book](#) | 

home Books Report add_newbook help ADMIN: libraryadmin@c2me.com

book poster	book id	book name	book author	revenue title	publishing date	available	remove book	update book
	1	Plane Trigonometry	299 RS	maths	2021-07-01T18:30:00.000Z	yes		
	10	introduction to dbms	1199 RS	dbms	2021-05-08T18:30:00.000Z	no		
	11	Differential Calculus	599 RS	maths	2019-07-01T18:30:00.000Z	yes		
	12	41 Years Paper	599 RS	maths	2018-07-01T18:30:00.000Z	yes		
	13	higher mathematics	699 RS	maths	2013-07-01T18:30:00.000Z	yes		
	14	concept of physq	599 RS	physq	2014-07-01T18:30:00.000Z	yes		

ADMIN UPDATE BOOK PAGE:

UPDATE BOOK DETAILS

home Books Report add_newbook help ADMIN: libraryadmin@c2me.com

bookid

 enter bookid

category

 enter book category

book name

 enter book name

price

 enter book price

title

 enter book title

publishing date

 dd-mm-yyyy 

edition

 enter book edition

publishing email

 enter publisher email

quantity

 enter quantity of books

UPDATE BOOK DETAILS

All copyrights reserved © 2021
developed by urvi and sourabh Patel

ADD NEW BOOK BY ADMIN PAGE:

ADD NEW BOOK

home Books Report add_newbook help ADMIN: libraryadmin@c2me.com

book id	quantity
<input type="text" value="enter book id"/>	<input type="text" value="enter quantity of books"/>
book name	category
<input type="text" value="enter book name"/>	<input type="text" value="enter book category"/>
title	price
<input type="text" value="enter book title"/>	<input type="text" value="enter book price"/>
edition	publishing date
<input type="text" value="enter book edition"/>	<input type="text" value="enter book publishing date"/>

ADD BOOK

ADMIN REPORTS PAGE:

All Books that are issued by user

home Books Report add_newbook help ADMIN: libraryadmin@c2me.com

poster	reg_no	user_email	issue_date	return_date	bookid
	2	exmp@gmail.com	2021-04-18T18:30:00.000Z	2021-04-22T18:30:00.000Z	3
	3	exmp@gmail.com	2021-04-18T18:30:00.000Z	2021-04-22T18:30:00.000Z	3
	4	exmp@gmail.com	2021-04-18T18:30:00.000Z	2021-04-22T18:30:00.000Z	3
	5	exmp@gmail.com	2021-04-18T18:30:00.000Z	2021-04-22T18:30:00.000Z	3
	11	gujarsourabh0802@gmail.com	2021-04-01T18:30:00.000Z	2021-05-01T18:30:00.000Z	7

ADMIN LATEST BOOKS :

ADMIN OF LIBRARY MANAGMENT SYSTEM Logout

home Books Report add_newbook help ADMIN: libraryadmin@c2me.com

Latestbooks category Search books

Books detials

book_poster	book_id	book_name	book_title	book_edition	book_category	book_price	quantity	publishing_date	avilable	delete book	update book detials
	1	Plane Trigonometry	maths	2	JEE MAINS	299 RS	6	2021-07-01T18:30:00.000Z	yes		

Concept of physics newcls introduction to ds introduction to ds introduction to ds

bk jeevni 12th Physics sspatel

ALGO ANALYSIS NCERT

SEARCH

All copyrights reserved © 2021

FILTER BY CATEGORY:

ADMIN OF LIBRARY MANAGMENT SYSTEM Logout

home Books Report add_newbook help ADMIN: libraryadmin@c2me.com

Latestbooks category Search books

All books with specified category in library

book_poster	book_id	book_name	book_title	book_edition	book_category	book_price	quantity	publishing_date	avilable	delete book	update book
	11	Differential Calculus	maths	1	JEE ADVANCED	599 RS	9	2019-07-01T18:30:00.000Z	yes		
	12	41 Years Paper	maths	1	JEE ADVANCED	599 RS	9	2018-07-01T18:30:00.000Z	yes		
	13	higher mathematics	maths	1	JEE ADVANCED	699 RS	9	2013-07-01T18:30:00.000Z	yes		

SEARCHED BOOK:

The screenshot displays the 'ADMIN OF LIBRARY MANAGMENT SYSTEM' interface. The top navigation bar includes links for home, Books, Report, add_newbook, and help, along with a user profile dropdown showing 'ADMIN: libraryadmin@c2me.com' and a Logout button. The main content area features a 'Latestbooks' section and a search bar. Below the search bar, a modal titled 'your Search books in library' is open, displaying a table of search results.

book_poster	book_id	book_name	book_title	book_edition	book_category	book_price	quantity	publishing_date	avilable	delete book	update book details
	11	Differential Calculus	maths	1	JEE ADVANCED	599 RS	9	2019-07-01T18:30:00.000Z	yes		
	12	41 Years Paper	maths	1	JEE ADVANCED	599 RS	9	2018-07-01T18:30:00.000Z	yes		
	13	higher mathematics	maths	1	JEE ADVANCED	699 RS	9	2013-07-01T18:30:00.000Z	yes		

The footer contains copyright information: 'All copyrights reserved © 2021 developed by jyoti and sourabh patel' and social media icons for Facebook, Instagram, Twitter, LinkedIn, and GitHub.

USER SIGNUP PAGE:

The screenshot shows the 'Signup' page of the library management system. The page has a navigation bar with links for home, Books, your_issued_books, and help. The main content area features a large, dark-themed signup form with the following fields:

- First name
- Last name
- mobile number
- Address
- Email
- Password
- Confirm password

A 'SIGNUP' button is located at the bottom right of the form, and a link for 'already have account? login' is positioned below it. The footer includes the same copyright notice and social media icons as the previous page.

USER LOGIN PAGE:

Sorry You Need To Login

home Books your_issued_books help

Email

Password

LOGIN FORGET PASSWORD?

new user? sign up

All copyrights reserved © 2021
developed by jyoti and sourabh patel

f o t i n

USER HOME PAGE:

LIBRARY MANAGMENT SYSTEM Logout

home Books your_issued_books help USER: psourabh080299@gmail.com

Latestbooks

Plane Trigonometry introduction to dams Differential Calculus 41 Years Paper higher mathematics concept of psql

Concept of physics newcfs introduction to ds introduction to ds introduction to ds

bk jeeval 12th Physics ssputel

category

JEE MAINS
btech
JEE ADVANCED
DATA STRUCTURE
DS
ALGO ANALYSIS
NCERT

Search books

Select Category

Enter book category

Select Author

Enter book author

Select Language

Enter book language

Select Title

Enter book title

SEARCH

All copyrights reserved © 2021
developed by jyoti and sourabh patel

f o t i n

ALL BOOKS PAGE:

All books in library										
home	Books	your_issued_books	help	USER: psourabh080299@gmail.com						
book poster	book id	book name	book title	edition	bookcategory	Price	quantity	publishing date	avilable	add to issue book
	1	Plane Trigonometry	maths	2	JEE MAINS	299 RS	6	2021-07-01T18:30:00.000Z	yes	
	10	introduction to dbms	dbms	1	itech	1199 RS	0	2021-05-08T18:30:00.000Z	no	
	11	Differential Calculus	maths	1	JEE ADVANCED	599 RS	9	2019-07-01T18:30:00.000Z	yes	
	12	41 Years Paper	maths	1	JEE ADVANCED	599 RS	9	2018-07-01T18:30:00.000Z	yes	
										





USER ISSUED BOOK PAGE:

Your Issued books										
home	Books	your_issued_books	help	USER: psourabh080299@gmail.com						
book poster	book name	bookid	bookcategory	bookprice	reg. no	title	book due date	book issue date	renew book	submit book
	introduction to dbms	10	itech	1199 RS	14	dbms	2021-05-01T18:30:00.000Z	2021-04-14T18:30:00.000Z		
	introduction to ds	4	DS	999 RS	15	data structure	2021-04-26T18:30:00.000Z	2021-04-06T18:30:00.000Z		
	Differential Calculus	11	JEE ADVANCED	599 RS	16	maths	2021-05-08T18:30:00.000Z	2021-04-27T18:30:00.000Z		
	concept of plsql	14	itech	399 RS	17	plsql	2021-05-06T18:30:00.000Z	2021-04-25T18:30:00.000Z		

EXTEND SUBMIT DATE:

Your Issued books

home Books your_issued_books help USER: psourabh080299@gmail.com

book poster	book name	bookid	bookcategory	bookprice	reg.no	title	book due date	book issue date	renew book	submit book
	introduction to dlims	10	itech	1199 RS	14	dlims	2021-05-01T18:30:00.000Z	2021-04-14T18:30:00.000Z	RENEW	SUBMIT
	introduction to its	4	itech	799 RS	15	its	2021-05-01T18:30:00.000Z	2021-04-14T18:30:00.000Z	RENEW	SUBMIT
	Differential Calculus	11	itech	799 RS	16	calculus	2021-05-01T18:30:00.000Z	2021-04-14T18:30:00.000Z	RENEW	SUBMIT
	concept of plsql	14	itech	599 RS	17	plsql	2021-05-06T18:30:00.000Z	2021-04-25T18:30:00.000Z	RENEW	SUBMIT

All copyrights reserved © 2021
developed by joti and sourabh patel

U19CS067