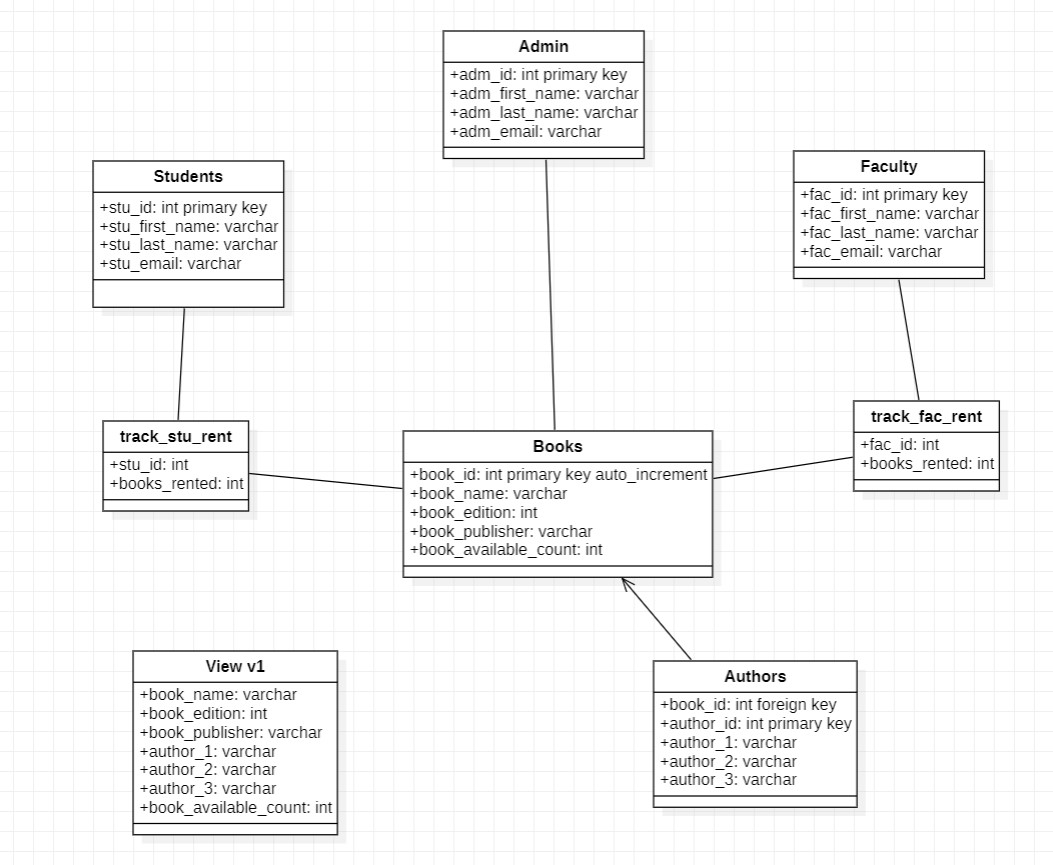
Schema - <https://youtu.be/pSRw7FzJN4s> (1:23 min)

Project Running - <https://youtu.be/w_5smSKmYVs> (1:26 min)

My project is implemented using Python, output can be seen on command prompt.

Tables/Views –

1. Students – Stores the Students’ information
2. Faculty – Stores the Faculty’s information
3. Admin – Stores the owner’s information
4. Track\_stu\_rent – tracks the number of books rented by a particular student
5. Track\_fac\_rent – tracks the number of books rented by a particular faculty
6. V1 – has joined information of books and authors.
7. Books – Has book related information.
8. Authors – Has author related information, for each book present in the Books table.



Delivered –

* I have 51 distinct books available in my database, but in reality, it stores 625 books (book\_available\_count column has the number of books present of each type).
* User can sign in/up.
* Unsigned in user (student or faculty) can display the books available (using view), get a count of number of books and get a count of distinct books.
* Signed in user (student or faculty) can rent book, check dues and checked the number of books rented.

Key SQL Commands used –

CREATE TABLE `lms`.`books` (

`book\_id` INT NOT NULL AUTO\_INCREMENT,

`book\_name` VARCHAR(45) NOT NULL,

`book\_edition` INT NOT NULL,

`book\_publisher` VARCHAR(45) NULL,

`book\_available\_count` INT NULL,

PRIMARY KEY (`book\_id`));

CREATE TABLE `lms`.`author` (

`book\_id` INT NULL,

`author\_id` INT NOT NULL,

`author\_1` VARCHAR(45) NULL,

`author\_2` VARCHAR(45) NULL,

`author\_3` VARCHAR(45) NULL,

PRIMARY KEY (`author\_id`),

INDEX `book\_id\_idx` (`book\_id` ASC) VISIBLE,

CONSTRAINT `book\_id`

FOREIGN KEY (`book\_id`)

REFERENCES `library`.`books` (`book\_id`)

ON DELETE CASCADE

ON UPDATE CASCADE);

CREATE TABLE `lms`.`students` (

`stu\_id` INT NOT NULL,

`stu\_first\_name` VARCHAR(45) NOT NULL,

`stu\_last\_name` VARCHAR(45) NOT NULL,

`stu\_email` VARCHAR(45) NOT NULL,

PRIMARY KEY (`stu\_id`));

CREATE TABLE `lms`.`faculty` (

`fac\_id` INT NOT NULL,

`fac\_first\_name` VARCHAR(45) NOT NULL,

`fac\_last\_name` VARCHAR(45) NOT NULL,

`fac\_email` VARCHAR(45) NOT NULL,

PRIMARY KEY (`fac\_id`));

CREATE TABLE `lms`.`track\_stu\_rent` (

`stu\_id` INT NULL,

`books\_rented` INT NULL);

CREATE TABLE `lms`.`track\_fac\_rent` (

`fac\_id` INT NULL,

`book\_rented` INT NULL);

#create view to display the details of books

CREATE VIEW v1 AS SELECT book\_name, book\_edition, book\_publisher, author\_1, author\_2, author\_3, book\_available\_count FROM books NATURAL JOIN author;

INSERT INTO library.books (book\_name, book\_edition, book\_publisher, book\_available\_count) VALUES ('Database System Concepts', 4, 'McGrawHill', 20);

INSERT INTO lms.author (book\_id, author\_1, author\_2, author\_3) VALUES (50, 'Daniel P. Friedman', 'Matthias Felleisen', 'NULL');

DELETE FROM books WHERE book\_name = 'oieur' and book\_edition = 32 and book\_publisher = 'kjdf';

#update the no. of books available, that is, rent a book

UPDATE books SET book\_available\_count = book\_available\_count - 1 where (book\_name = 'Database System Concepts' and book\_edition = 7);

#update number of books rented for students

UPDATE track\_stu\_rent SET books\_rented = books\_rented + 1 WHERE stu\_id = 15

#get count of the books

select sum(book\_available\_count) as count from books;

#get count of distinct books

select count(book\_id) as count from books;