HOSTEL MANAGEMENT SYSTEM

A Project Report submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

SRAVANTHI.T	**121810307010**
NANDA VARDHAN.R	**121810307026**
CHANDANA.P	**121810307029**
SAI RUCHITH.R	**121810307060**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM INSTITUTE OF TECHNOLOGY

GITAM (Deemed to be University)

VISAKHAPATNAM

DECEMBER 2020

INSTITUTE OF TECHNOLOGY

GITAM (Deemed to be University)

VISHAKAPTNAM

DECLARATION

I, hereby declare that the project report entitled “HOSTEL MANAGEMENT SYSTEM” is an original work done through XAAMP and provide this opportunity by the Department of Computer Science and Engineering, GITAM Institute of Technology, GITAM (Deemed to be University), Visakhapatnam submitted in partial fulfillment of the requirements for the award of the degree of B.Tech. in Computer Science and Engineering. The work has not been submitted to any other college or University for the award of any degree or diploma.

Registration No	Name of the Student	Student Signature
121810307010	SRAVANTHI .T	
121810307026	NANDA VARDHAN .R	
121810307029	CHANDANA .P	
121810307060	SAI RUCHITH .R	

INSTITUTE OF TECHNOLOGY

GITAM (Deemed to be University)

VISHAKAPATNAM

CERTIFICATE

This is to certify that the project report entitled “HOSTEL MANAGEMENT SYSTEM” is a Bonafide record of work carried out by “121810307010-Sravanthi , 121810307026-Nanda Vardhan.R ,121810307029-Chandana.P , 121810307060-Sai Ruchith” submitted in partial fulfillment of requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering.

INTERNSHIP GUIDE **Prof.K.Thammi Reddy**

INTERNSHIP REVIEWER-1 **Yashwanth Amanapu**

INTERNSHIP REVIEWER-2 **Komal Kashyap**

TABLE OF CONTENTS

ABSTRACT

As the name specifies “HOSTEL MANAGEMENT SYSTEM” is a software developed for managing various activities in the hostel. For the past few years the number of educational institutions are increasing rapidly. Thereby the number of hostels are also increasing for the accom-modation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software’s are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually.

Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system Which is more user friendly and more GUI oriented. We can improve the efficiency of the system, thus overcome the drawbacks of the existing system.

- Less human error
- Strength and strain of manual labour can be reduced
- High security
- Data redundancy can be avoided to some extent
- Data consistency
- Easy to handle
- Easy data updating
- Easy record keeping

INTRODUCTION

Problem definition

We have got many hostels in our university. All these hostels are presently managed manually by the hostel office. The Registration form verification to the different data processing are done manually.

Thus there are a lot of repetitions which can be easily avoided. And hence there is a lot of strain on the person who are running the hostel and software's are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually

Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented. We can improve the efficiency of the system, thus overcome the drawbacks of the existing system.

1.Admin Panel

1.Admin Login

Admin can login through login form.

2. Admin Profile

Admin can manage his own profile. Admin can also change his password

3.Courses

Admin can create add course, edit courses and also delete the course

4. Rooms

Admin can create rooms and allots seater to particular rooms and assign the fees.

5. Registration

Admin can create student profile and allot the rooms

6.Manage the Registration

Admin can manage the all the student Profile. Take a print out of all profiles.

7.Forgot Password

Admin can also retrieve the password if admin forgot the password

2.User Panel

1. **User Registration** User can register through user registration form
2. **User Login--** User can login through login form
3. **Forgot Password—**user can retrieve password through forgot password link
4. User Dashboard
5. **User Profile—**User can manage own profile

6. **Book Hostel** – User can book hostel
7. **Room Details-** Booked Room Details
8. **Change Password-** User Can change own password
9. **User access log-** User can watch last login details

SYSTEM ENVIRONMENT

Software Configuration

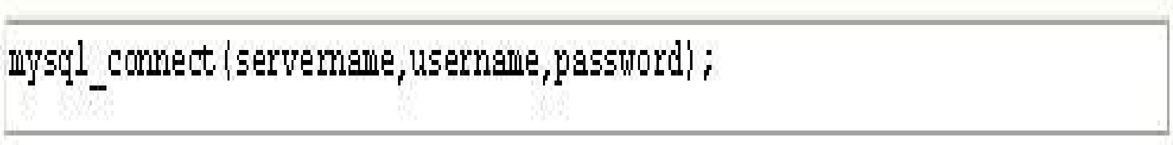
1.OS : Windows XP

2.PHP Triad (PHP5.6, MySQL, Apache, and PHPMyAdmin)

Create a connection to a database

Before you can access data in a database, you must create a connection to the database.In PHP, this is done with the mysqli_connect() function.

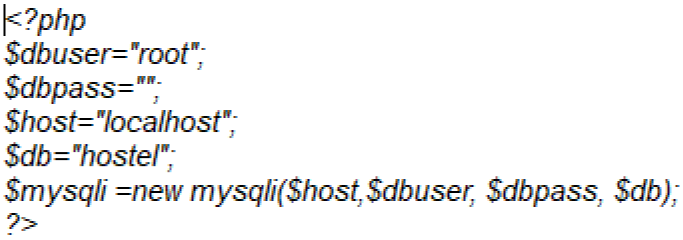
Syntax



```
mysql_connect(servername,username,password);
```

DATABASE CONNECTION

In the following php code we store the server name localhost in host variable, by default the user name is root which we store in dbuser, and database password is “” by default which we store in dbpass and database name is hostel which we store in db.



```
<?php
$dbuser="root";
$dbpass="";
$host="localhost";
$db="hostel";
$mysqli =new mysqli($host,$dbuser, $dbpass, $db);
?>
```

ER Diagram

An Entity Relation(ER) Diagram is a specialized graphics that illustrates the interrelationship between entities in a database. ER diagrams often use symbols to represent 3 different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes.

An Entity Relationship Model (ERM), in software engineering is an abstract and conceptual representation of data. Entity Relationship modeling is a relational schema database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relation database, and its requirements in a top-down fashion

Entity:

Entity is the thing which we want to store information. It is an elementary basic building block of storing information about business process. An entity represents an object defined within the information system about which you want to store information. Entities are distinct things in the enterprise.

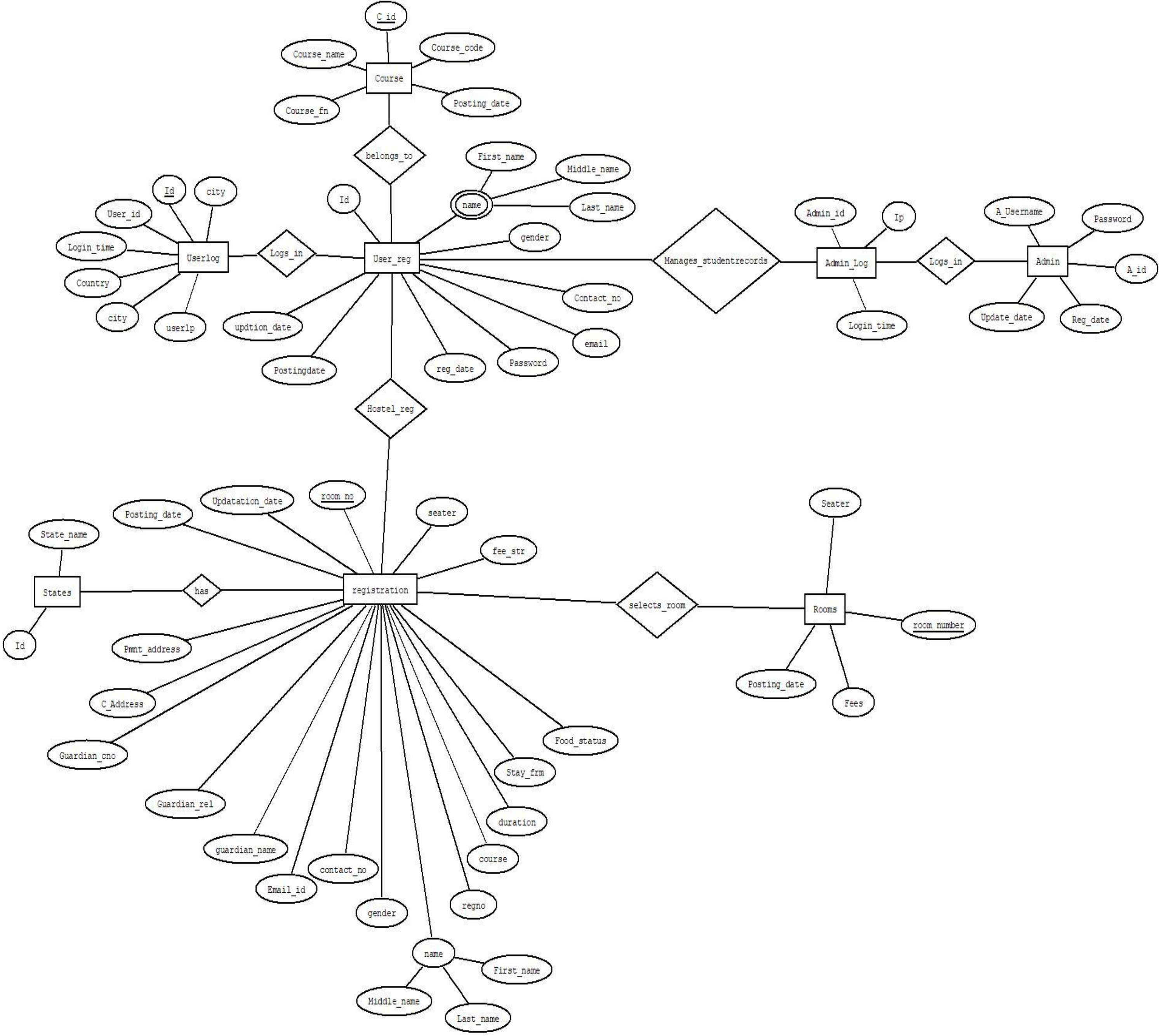
Relationships:

A relationship is a named collection or association between entities or used to relate two or more entities with some common attributes or meaningful interaction between the objects

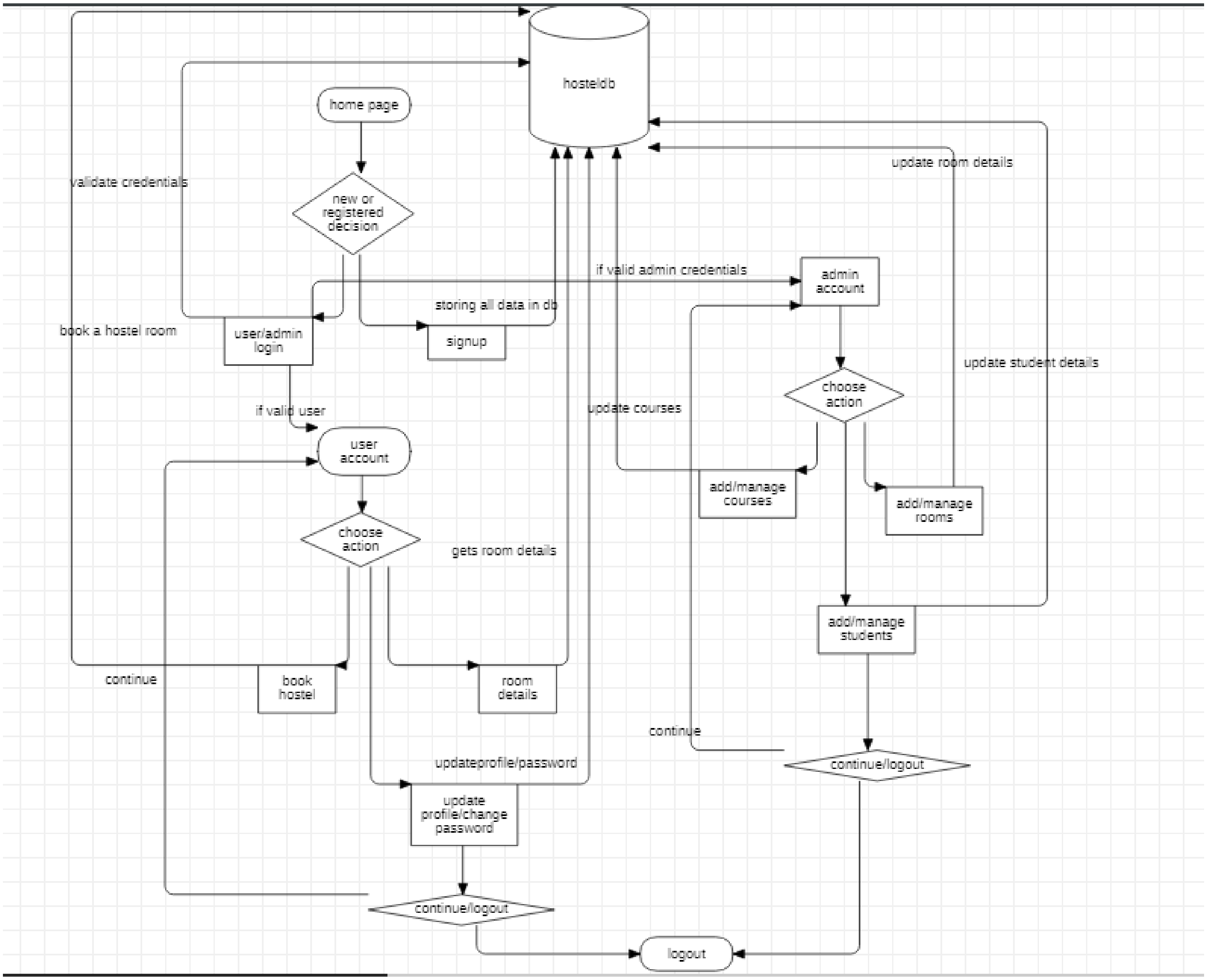
Attributes:

Attributes are the properties of the entities and relationship, Descriptor of the entity. Attributes are elementary pieces of information attached to an entity.

Hostel management system-Er diagram



Flow Chart



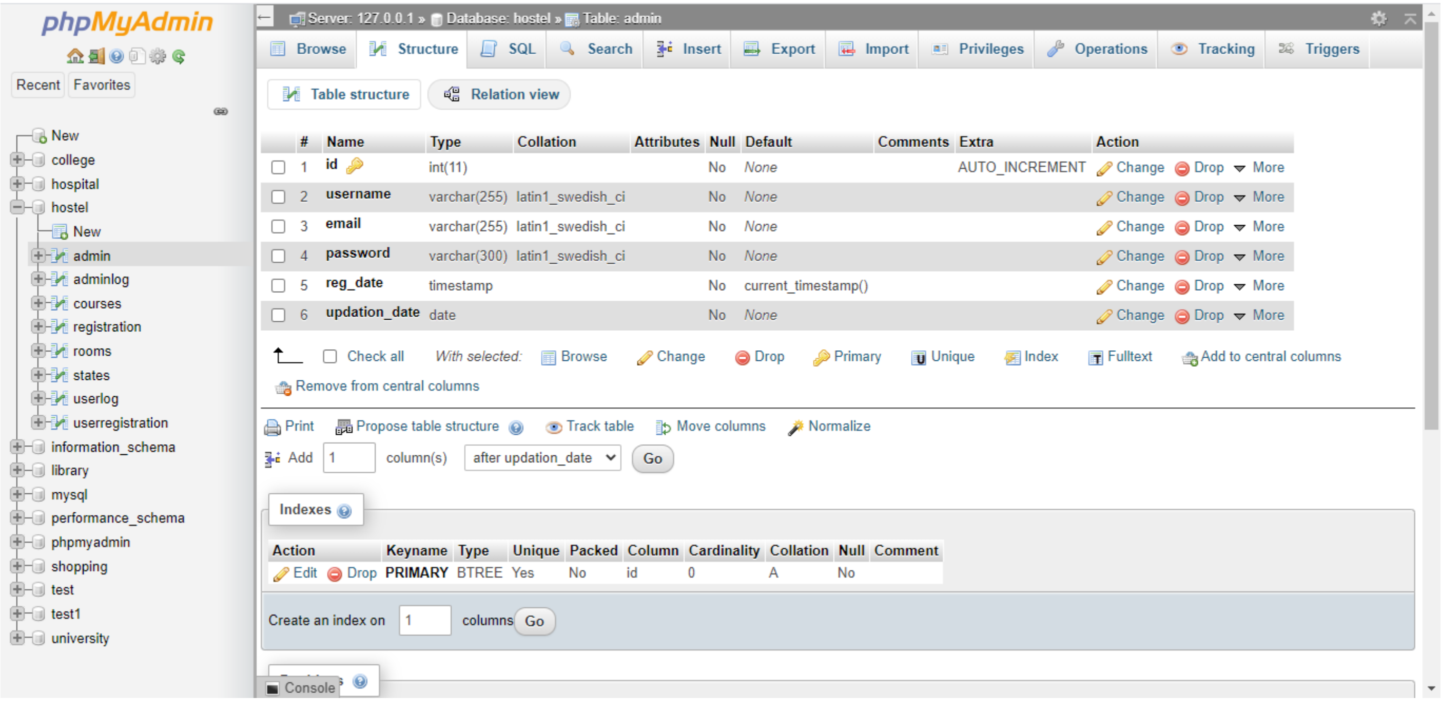
Database Design

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MS Access database has been chosen for developing the relevant databases.

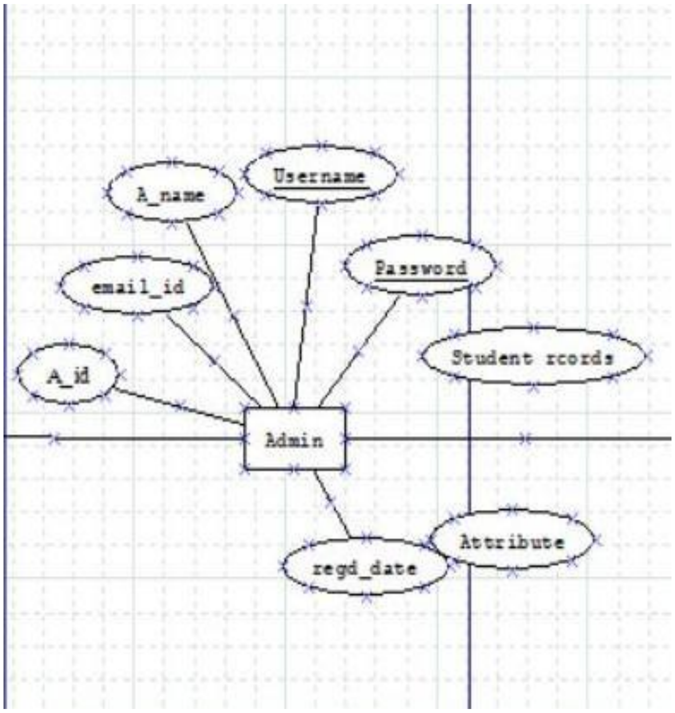
Admin Table Structure

Admin table stores the information regarding admin login and will be used at validating admin login .Here the entity is Admin and attributes are id,username,email,password,reg_date,updation_date.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	username	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
3	email	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
4	password	varchar(300)	latin1_swedish_ci		No	None			Change Drop More
5	reg_date	timestamp			No	current_timestamp()			Change Drop More
6	updation_date	date			No	None			Change Drop More

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	0	A	No	



-- Table structure for table `admin`--

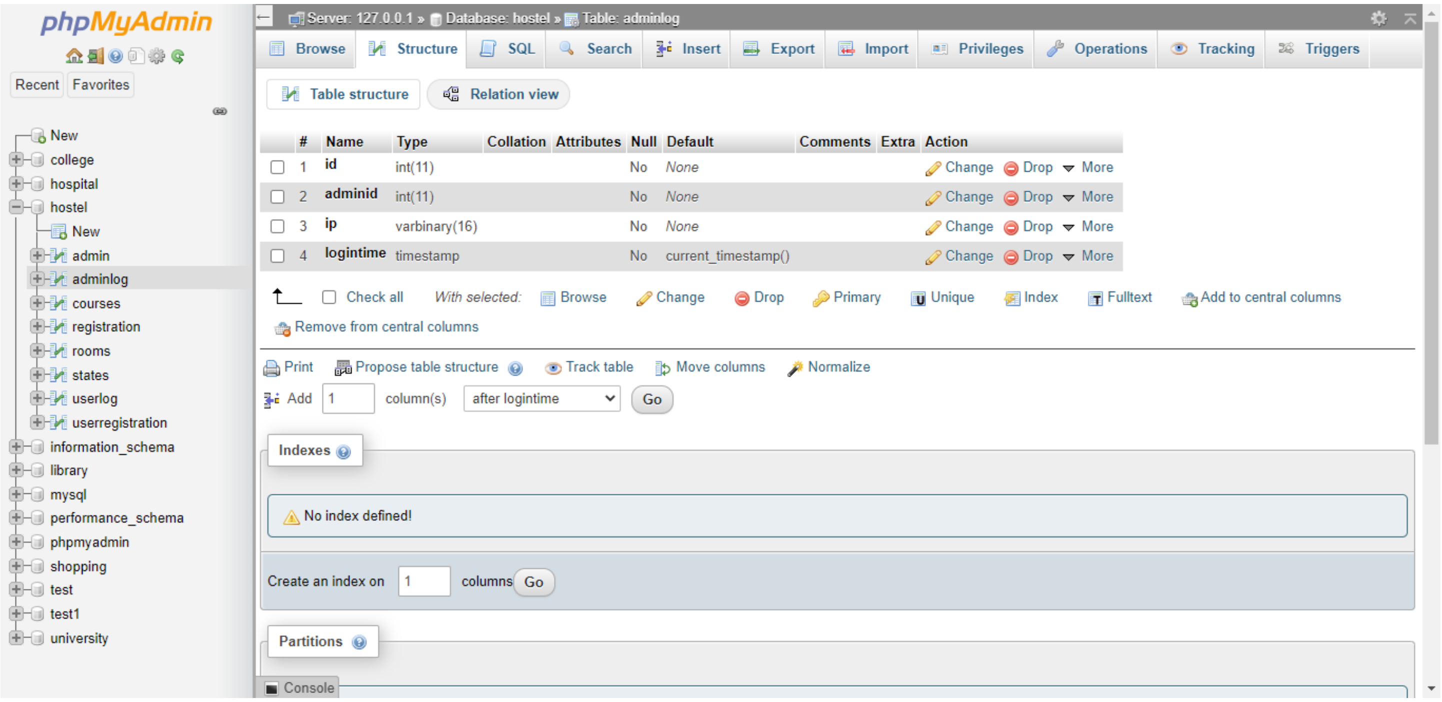
```
CREATE TABLE `admin` (
`id` int(11) NOT NULL,
`username` varchar(255) NOT NULL,
`email` varchar(255) NOT NULL,
`password` varchar(300) NOT NULL,
`reg_date` timestamp NOT NULL DEFAULT
current_timestamp(),
`updation_date` date NOT NULL
);
```

--Insering values into admin table----

```
INSERT INTO `admin` (`id`, `username`,
`email`, `password`, `reg_date`,
`updation_date`) VALUES
(1, 'admin',
'sravanthithukivakam@gmail.com',
'Test@1234', '2020-12-12 20:31:45', '2020-
12-12');
```

Adminlog Table Structure

This table stores the data of admin log ,which contains attributes such as id,adminid,ip,logintime.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	*None*			Change Drop More
2	adminid	int(11)			No	*None*			Change Drop More
3	ip	varbinary(16)			No	*None*			Change Drop More
4	logintime	timestamp			No	current_timestamp()			Change Drop More

No index defined!

-- Table structure for table `adminlog`--

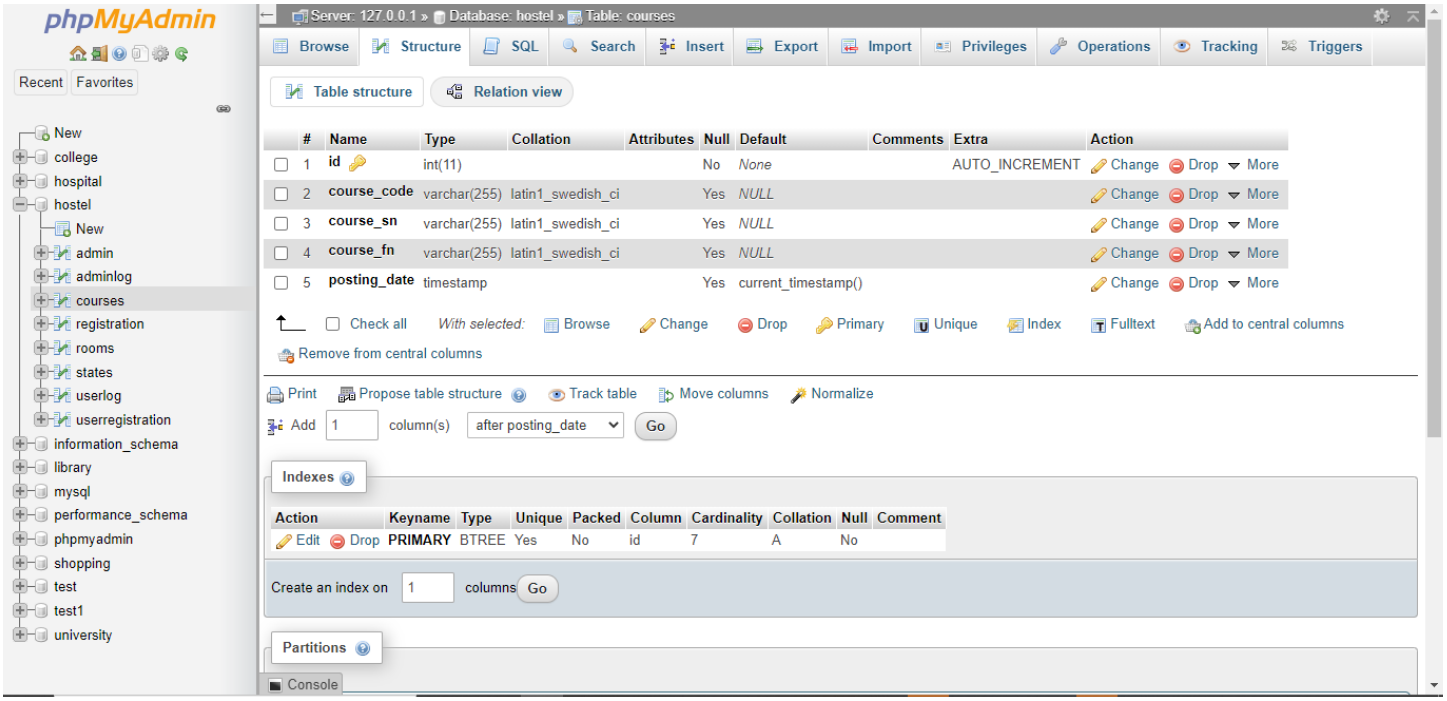
```
CREATE TABLE `adminlog` (
  `id` int(11) NOT NULL,
  `adminid` int(11) NOT NULL,
  `ip` varbinary(16) NOT NULL,
  `logintime` timestamp NOT NULL DEFAULT current_timestamp()
);
```

--Insering values into adminlog table----

```
INSERT INTO `adminlog` (`id`, `adminid`, `ip`, `logintime) VALUES
(1, '1', '234', '20:31:45');
```

Courses Table Structure

Courses table stores the information regarding student courses in which they are studying and will be used at hostel registration time.



Server: 127.0.0.1 » Database: hostel » Table: courses

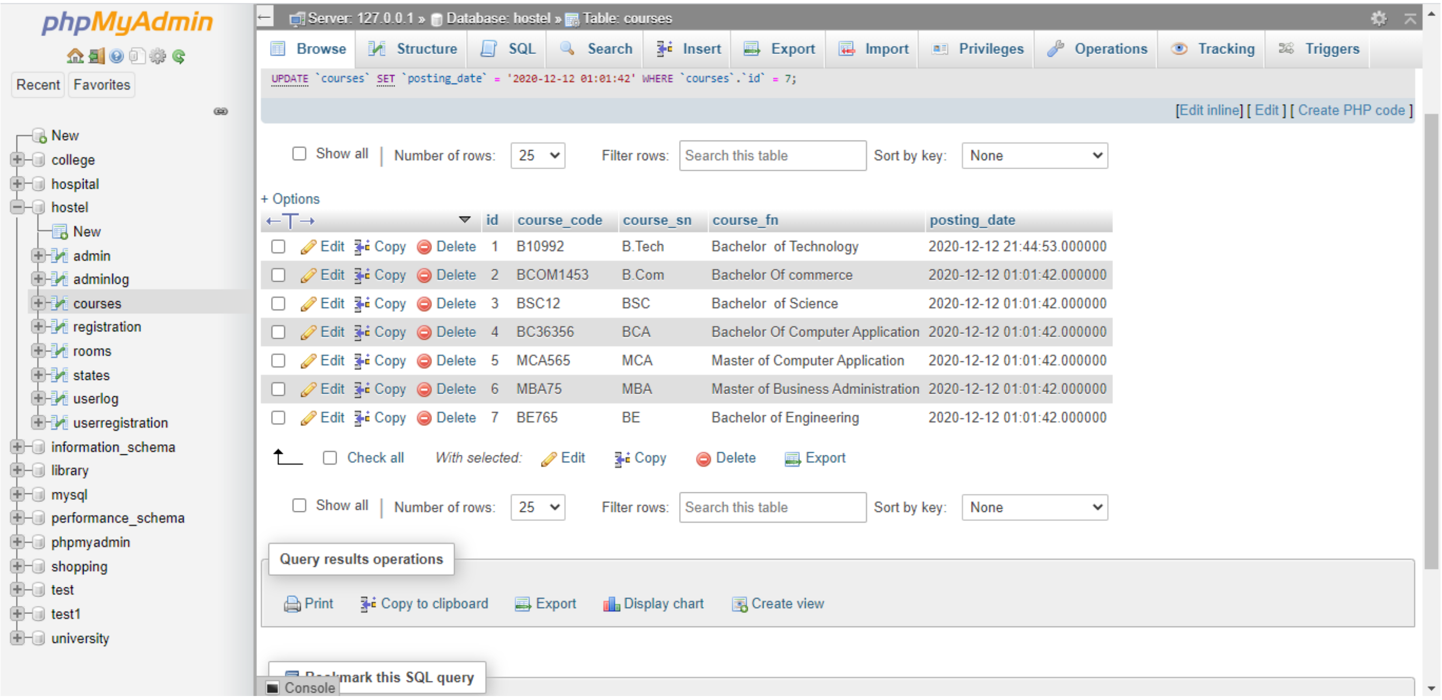
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	course_code	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	course_sn	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop More
4	course_fn	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop More
5	posting_date	timestamp			Yes	current_timestamp()			Change Drop More

Add 1 column(s) after posting_date

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	7	A	No	

Create an index on 1 columns

```
-- Table structure for table `courses`

CREATE TABLE `courses` (
  `id` int(11) NOT NULL,
  `course_code` varchar(255) DEFAULT NULL,
  `course_sn` varchar(255) DEFAULT NULL,
  `course_fn` varchar(255) DEFAULT NULL,
  `posting_date` timestamp NULL DEFAULT current_timestamp()
) ;

-- Inserting values into table `courses`

INSERT INTO `courses` (`id`, `course_code`, `course_sn`, `course_fn`,
`posting_date`) VALUES
(1, 'B10992', 'B.Tech', 'Bachelor  of Technology', '2020-12-12 19:31:42'),
(2, 'BCOM1453', 'B.Com', 'Bachelor Of commerce ', '2020-12-12 19:31:42'),
(3, 'BSC12', 'BSC', 'Bachelor  of Science', '2020-12-12 19:31:42'),
(4, 'BC36356', 'BCA', 'Bachelor Of Computer Application', '2020-12-12 19:31:42'),
(5, 'MCA565', 'MCA', 'Master of Computer Application', '2020-12-12 19:31:42'),
(6, 'MBA75', 'MBA', 'Master of Business Administration', '2020-12-12 19:31:42'),
(7, 'BE765', 'BE', 'Bachelor of Engineering', '2020-12-12 19:31:42');
```

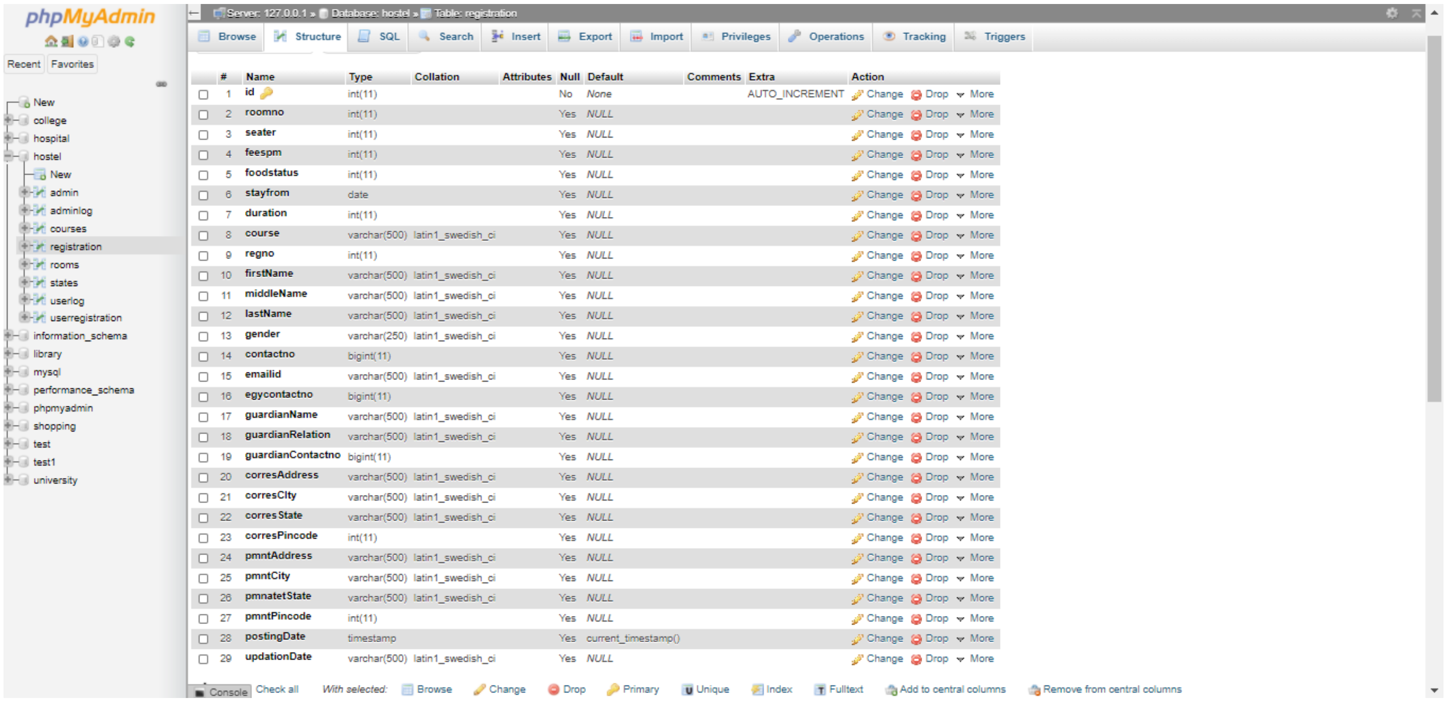


```
UPDATE `courses` SET `posting_date` = '2020-12-12 01:01:42' WHERE `courses`.`id` = 7;
```

	id	course_code	course_sn	course_fn	posting_date
□ Edit Copy Delete	1	B10992	B.Tech	Bachelor of Technology	2020-12-12 21:44:53.000000
□ Edit Copy Delete	2	BCOM1453	B.Com	Bachelor Of commerce	2020-12-12 01:01:42.000000
□ Edit Copy Delete	3	BSC12	BSC	Bachelor of Science	2020-12-12 01:01:42.000000
□ Edit Copy Delete	4	BC36356	BCA	Bachelor Of Computer Application	2020-12-12 01:01:42.000000
□ Edit Copy Delete	5	MCA565	MCA	Master of Computer Application	2020-12-12 01:01:42.000000
□ Edit Copy Delete	6	MBA75	MBA	Master of Business Administration	2020-12-12 01:01:42.000000
□ Edit Copy Delete	7	BE765	BE	Bachelor of Engineering	2020-12-12 01:01:42.000000

Registeration Table Structure

Registeration table stores the information regarding student courses,regno,all personal information along with room details and guardian details



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
□	2	roomno	int(11)			Yes	NULL			Change Drop More
□	3	seater	int(11)			Yes	NULL			Change Drop More
□	4	feespm	int(11)			Yes	NULL			Change Drop More
□	5	foodstatus	int(11)			Yes	NULL			Change Drop More
□	6	stayfrom	date			Yes	NULL			Change Drop More
□	7	duration	int(11)			Yes	NULL			Change Drop More
□	8	course	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	9	regno	int(11)			Yes	NULL			Change Drop More
□	10	firstName	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	11	middleName	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	12	lastName	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	13	gender	varchar(250)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	14	contactno	bigint(11)			Yes	NULL			Change Drop More
□	15	emailid	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	16	egycontactno	bigint(11)			Yes	NULL			Change Drop More
□	17	guardianName	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	18	guardianRelation	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	19	guardianContactno	bigint(11)			Yes	NULL			Change Drop More
□	20	corresAddress	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	21	corresCIty	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	22	corresState	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	23	corresPincode	int(11)			Yes	NULL			Change Drop More
□	24	pmntAddress	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	25	pmntCity	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	26	pmnatetState	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More
□	27	pmntPincode	int(11)			Yes	NULL			Change Drop More
□	28	postingDate	timestamp			Yes	current_timestamp()			Change Drop More
□	29	updationDate	varchar(500)	latin1_swedish_ci		Yes	NULL			Change Drop More

-- Table structure for table `registration`

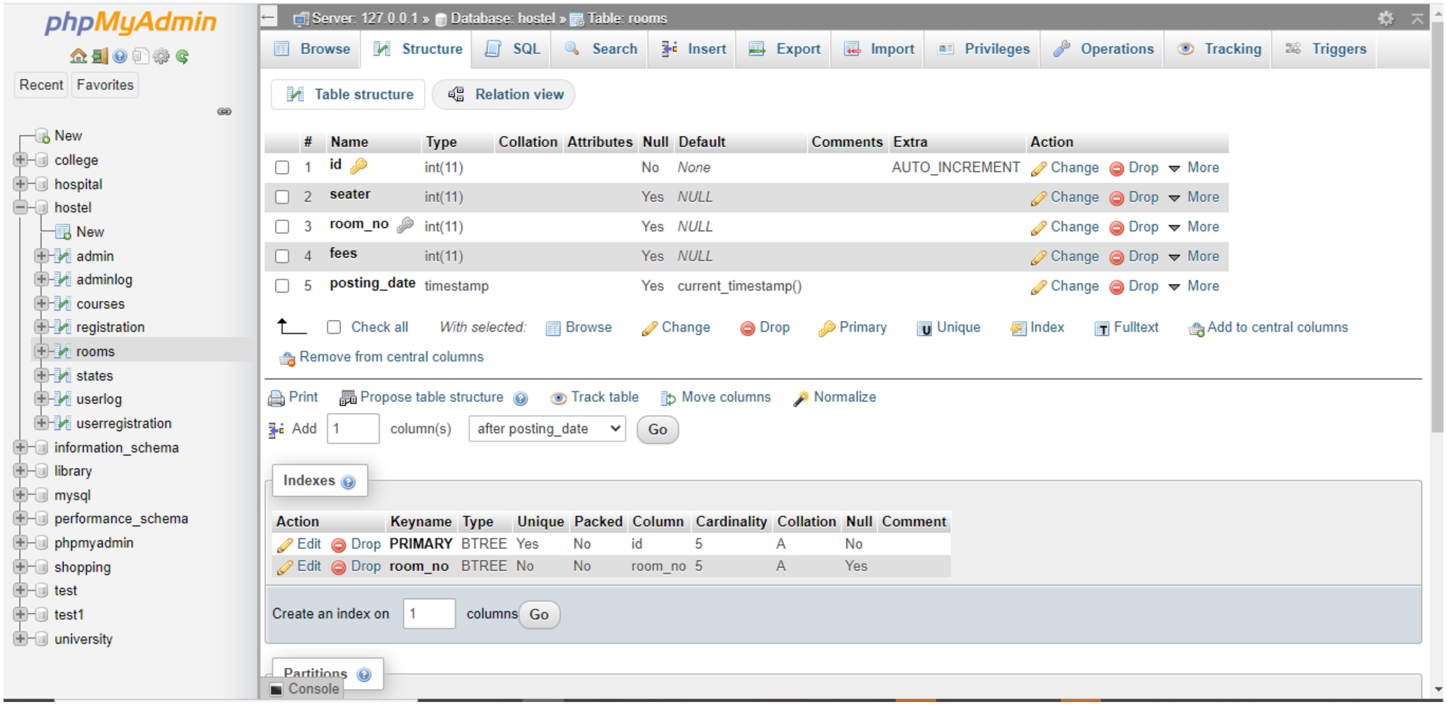
```
CREATE TABLE `registration` (
  `id` int(11) NOT NULL,
  `roomno` int(11) DEFAULT NULL,
  `seater` int(11) DEFAULT NULL,
  `feespm` int(11) DEFAULT NULL,
  `foodstatus` int(11) DEFAULT NULL,
  `stayfrom` date DEFAULT NULL,
  `duration` int(11) DEFAULT NULL,
  `course` varchar(500) DEFAULT NULL,
  `regno` int(11) DEFAULT NULL,
  `firstName` varchar(500) DEFAULT NULL,
  `middleName` varchar(500) DEFAULT NULL,
  `lastName` varchar(500) DEFAULT NULL,
  `gender` varchar(250) DEFAULT NULL,
  `contactno` bigint(11) DEFAULT NULL,
  `emailid` varchar(500) DEFAULT NULL,
  `egycontactno` bigint(11) DEFAULT NULL,
  `guardianName` varchar(500) DEFAULT NULL,
  `guardianRelation` varchar(500) DEFAULT NULL,
  `guardianContactno` bigint(11) DEFAULT NULL,
  `corresAddress` varchar(500) DEFAULT NULL,
  `corresCIty` varchar(500) DEFAULT NULL,
  `corresState` varchar(500) DEFAULT NULL,
  `corresPincode` int(11) DEFAULT NULL,
  `pmntAddress` varchar(500) DEFAULT NULL,
  `pmntCity` varchar(500) DEFAULT NULL,
  `pmnatetState` varchar(500) DEFAULT NULL,
  `pmntPincode` int(11) DEFAULT NULL,
  `postingDate` timestamp NULL DEFAULT current_timestamp(),
  `updationDate` varchar(500) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

-- Inserting values into table `registration`

```
INSERT INTO `registration` (`id`, `roomno`, `seater`, `feespm`, `foodstatus`,
`stayfrom`, `duration`, `course`, `regno`, `firstName`, `middleName`, `lastName`,
`gender`, `contactno`, `emailid`, `egycontactno`, `guardianName`,
`guardianRelation`, `guardianContactno`, `corresAddress`, `corresCIty`,
`corresState`, `corresPincode`, `pmntAddress`, `pmntCity`, `pmnatetState`,
`pmntPincode`, `postingDate`, `updationDate`) VALUES
(3, 100, 5, 8000, 0, '2020-12-06', 3, 'Bachelor  of Technology', 12154,
'sravanthi', 'thukivakam', 'reddy', 'female', 367262, 'srav@gmail.com', 45644,
'abc', 'uncle', 5644, 'vizag', 'vizag', 'Andhra pradesh',517501, 'tirupati',
'chittor', 'Andhra pradesh', 517501, '2020-12-05 14:58:26', NULL);
```

Rooms Table Structure

Rooms table store information regarding the booked rooms by student and the fee it costs for the stay.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
2	seater	int(11)			Yes	*NULL*			Change Drop More
3	room_no	int(11)			Yes	*NULL*			Change Drop More
4	fees	int(11)			Yes	*NULL*			Change Drop More
5	posting_date	timestamp			Yes	current_timestamp()			Change Drop More

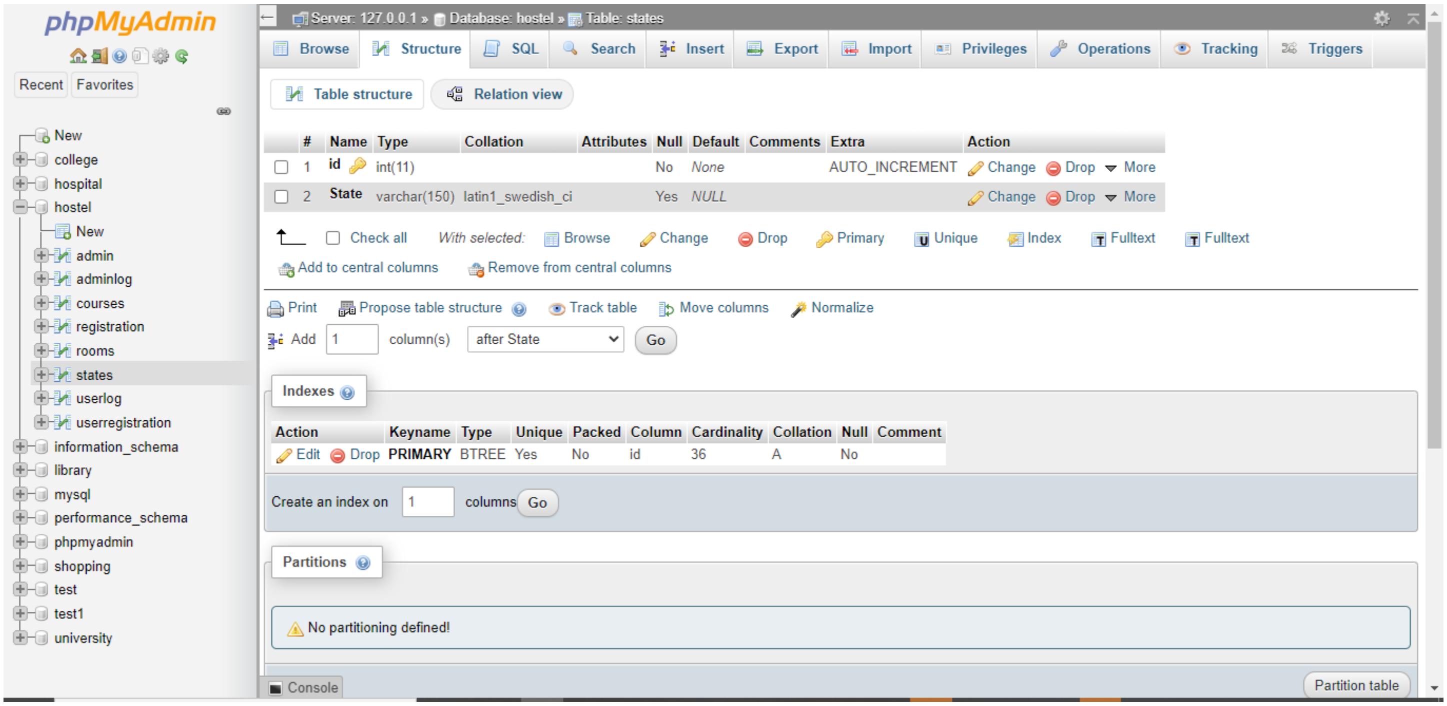
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	5	A	No	
Edit Drop	room_no	BTREE	No	No	room_no	5	A	Yes	

```
-- Table structure for table `rooms`

CREATE TABLE `rooms` (
  `id` int(11) NOT NULL,
  `seater` int(11) DEFAULT NULL,
  `room_no` int(11) DEFAULT NULL,
  `fees` int(11) DEFAULT NULL,
  `posting_date` timestamp NULL DEFAULT current_timestamp()
);
```

```
-- Inserting values into table `rooms`

INSERT INTO `rooms` (`id`, `seater`, `room_no`, `fees`, `posting_date`) VALUES
(1, 5, 100, 8000, '2020-04-11 22:45:43'),
(2, 2, 201, 6000, '2020-04-12 01:30:47'),
(3, 2, 200, 6000, '2020-04-12 01:30:58'),
(4, 3, 112, 4000, '2020-04-12 01:31:07'),
(5, 5, 132, 2000, '2020-04-12 01:31:15');
```

States Table Structure

Stable store information regarding the states in which the student belongs to.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	State	varchar(150)	latin1_swedish_ci		Yes	NULL			Change Drop More

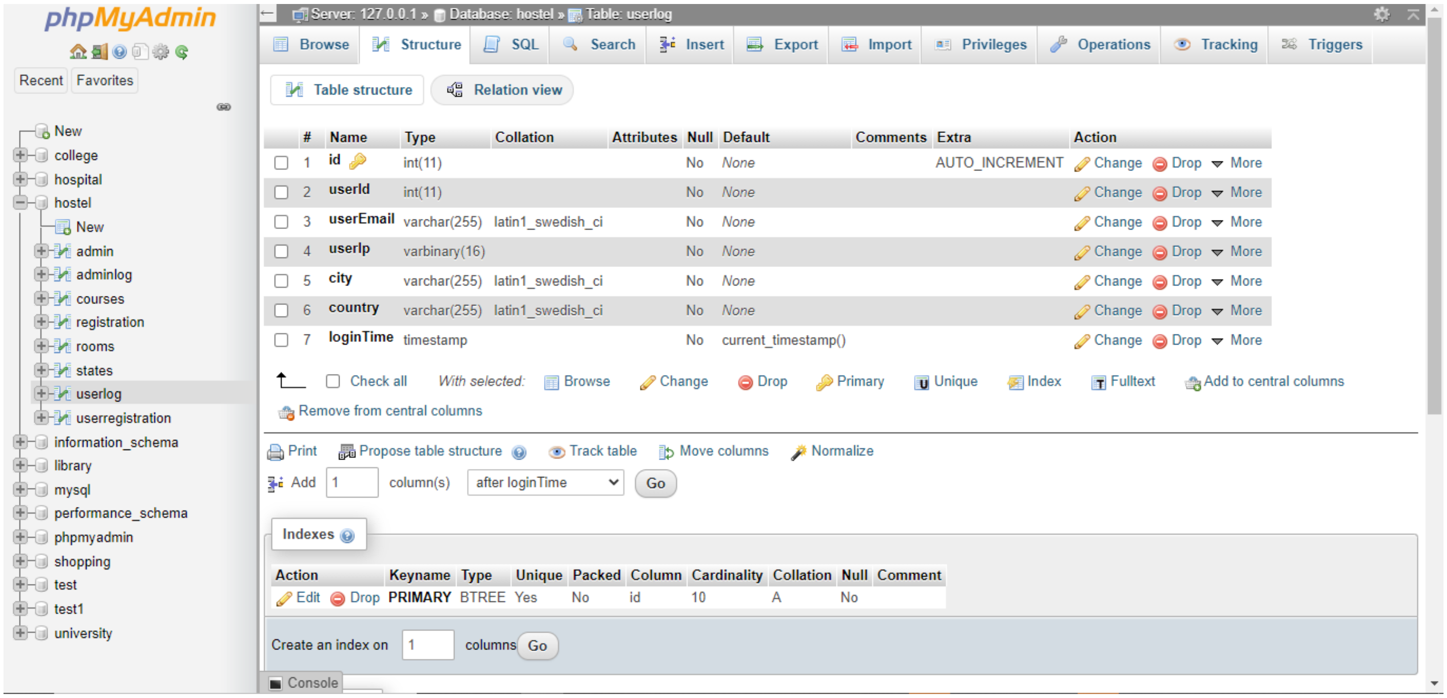
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	36	A	No	

No partitioning defined!

```
-- Table structure for table `states`

CREATE TABLE `states` (
  `id` int(11) NOT NULL,
  `State` varchar(150) DEFAULT NULL
);



-- Inserting values into table `states`

INSERT INTO `states` (`id`, `State`) VALUES
(1, 'Andaman and Nicobar Island (UT)'),
(2, 'Andhra Pradesh'),
(3, 'Arunachal Pradesh'),
(4, 'Assam'),
(5, 'Bihar'),
(6, 'Chandigarh (UT)'),
(7, 'Chhattisgarh'),
(8, 'Dadra and Nagar Haveli (UT)'),
(9, 'Daman and Diu (UT)'),
(10, 'Delhi (NCT)'),
(11, 'Goa'),
(12, 'Gujarat'),
(13, 'Haryana'),
(14, 'Himachal Pradesh'),
(15, 'Jammu and Kashmir'),
(16, 'Jharkhand'),
(17, 'Karnataka'),
(18, 'Kerala'),
(19, 'Lakshadweep (UT)'),
(20, 'Madhya Pradesh'),
(21, 'Maharashtra'),
(22, 'Manipur'),
(23, 'Meghalaya'),
(24, 'Mizoram'),
(25, 'Nagaland'),
(26, 'Odisha'),
(27, 'Puducherry (UT)'),
(28, 'Punjab'),
(29, 'Rajastha'),
(30, 'Sikkim'),
(31, 'Tamil Nadu'),
(32, 'Telangana'),
(33, 'Tripura'),
(34, 'Uttarakhand'),
(35, 'Uttar Pradesh'),
(36, 'West Bengal');
```

Userlog Table Structure

Userlog store information regarding the user login details.



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	userId	int(11)			No	*None*			Change Drop More
□	3	userEmail	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	4	userIp	varbinary(16)			No	*None*			Change Drop More
□	5	city	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	6	country	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	7	loginTime	timestamp			No	current_timestamp()			Change Drop More

□ Check all *With selected:* Browse Change Drop Primary Unique Index Fulltext Add to central columns

Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after loginTime Go

Indexes

Action		Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	Drop	PRIMARY	BTREE	Yes	No	id	10	A	No	

Create an index on 1 columns Go

-- Table structure for table `userlog`

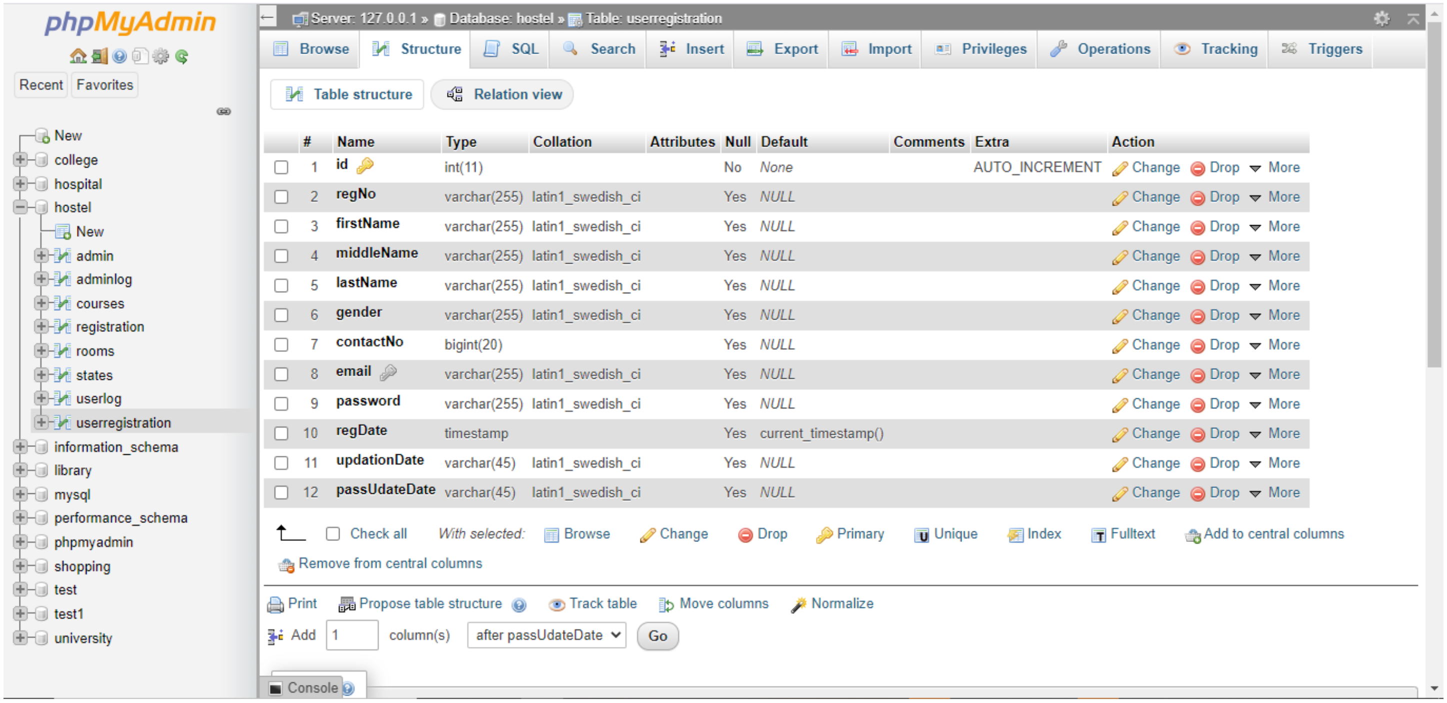
```
  CREATE TABLE `userlog` (
  `id` int(11) NOT NULL,
  `userId` int(11) NOT NULL,
  `userEmail` varchar(255) NOT NULL,
  `userIp` varbinary(16) NOT NULL,
  `city` varchar(255) NOT NULL,
  `country` varchar(255) NOT NULL,
  `loginTime` timestamp NOT NULL DEFAULT current_timestamp()
);
```

-- Inserting values into table `userlog`

```
INSERT INTO `userlog` (`id`, `userId`, `userEmail`, `userIp`, `city`, `country`,
`loginTime`) VALUES
(6, 3, '10806121', 0x3a3a31, '', '', '2020-07-20 14:56:45');
```

Userregistration Table Structure

Userregistration table stores information regarding student register details and are used when vaidating the user login credentials.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
2	regNo	varchar(255)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
3	firstName	varchar(255)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
4	middleName	varchar(255)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
5	lastName	varchar(255)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
6	gender	varchar(255)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
7	contactNo	bigint(20)			Yes	*NULL*			Change Drop More
8	email	varchar(255)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
9	password	varchar(255)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
10	regDate	timestamp			Yes	current_timestamp()			Change Drop More
11	updationDate	varchar(45)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
12	passUdateDate	varchar(45)	latin1_swedish_ci		Yes	*NULL*			Change Drop More

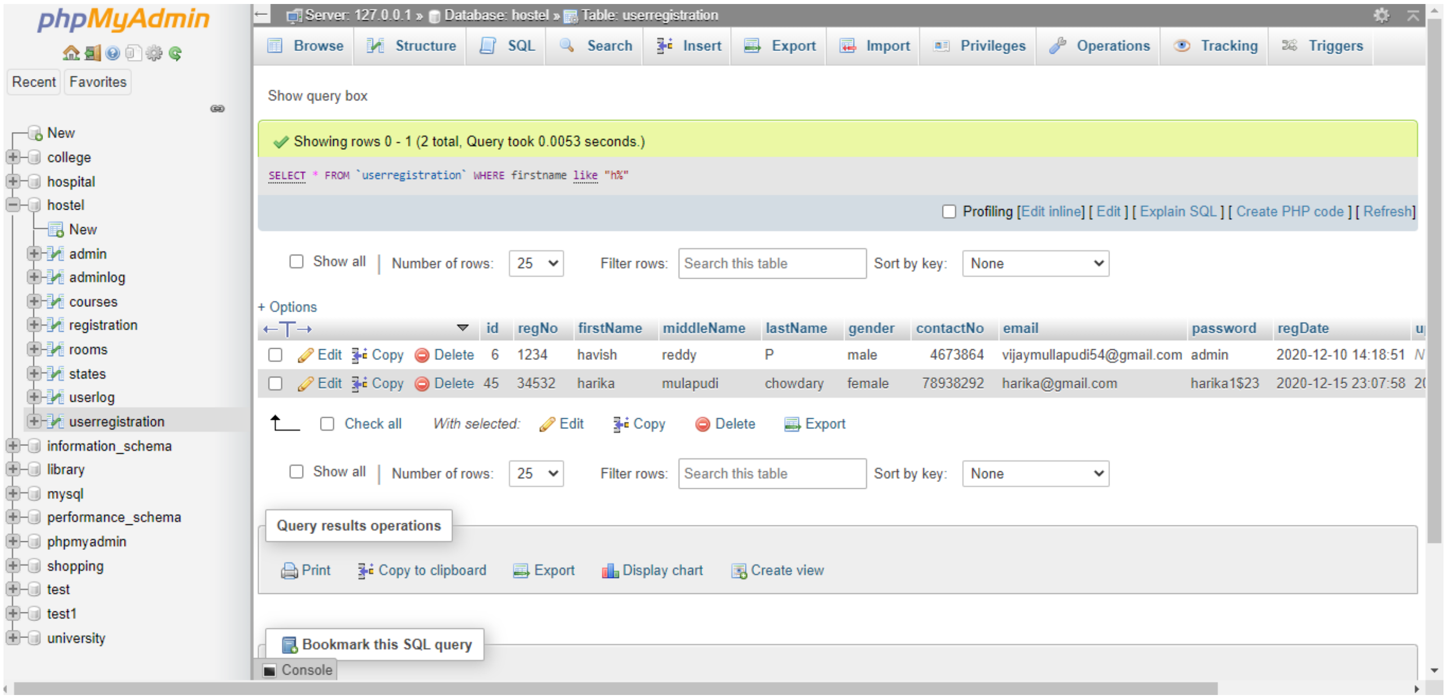
```
-- Table structure for table `userregistration`

CREATE TABLE `userregistration` (
  `id` int(11) NOT NULL,
  `regNo` varchar(255) DEFAULT NULL,
  `firstName` varchar(255) DEFAULT NULL,
  `middleName` varchar(255) DEFAULT NULL,
  `lastName` varchar(255) DEFAULT NULL,
  `gender` varchar(255) DEFAULT NULL,
  `contactNo` bigint(20) DEFAULT NULL,
  `email` varchar(255) DEFAULT NULL,
  `password` varchar(255) DEFAULT NULL,
  `regDate` timestamp NULL DEFAULT current_timestamp(),
  `updationDate` varchar(45) DEFAULT NULL,
  `passUdateDate` varchar(45) DEFAULT NULL
);
```

```
-- Inserting values into table `userregistration`

INSERT INTO `userregistration` (`id`, `regNo`, `firstName`, `middleName`,
`lastName`, `gender`, `contactNo`, `email`, `password`, `regDate`,
`updationDate`, `passUdateDate`) VALUES
(3, '10806121', 'Anuj', '', 'kumar', 'male', 1234567890, 'test@gmail.com',
'Test@123', '2020-07-20 14:56:18', NULL, NULL);
```

SQL QUERIES

1)To view the students who have name starting with h in userregistration table we run the sql query.

SELECT * FROM `userregistration` WHERE firstname like "h%";



✓ Showing rows 0 - 1 (2 total, Query took 0.0053 seconds.)

```
SELECT * FROM `userregistration` WHERE firstname like "h%"
```

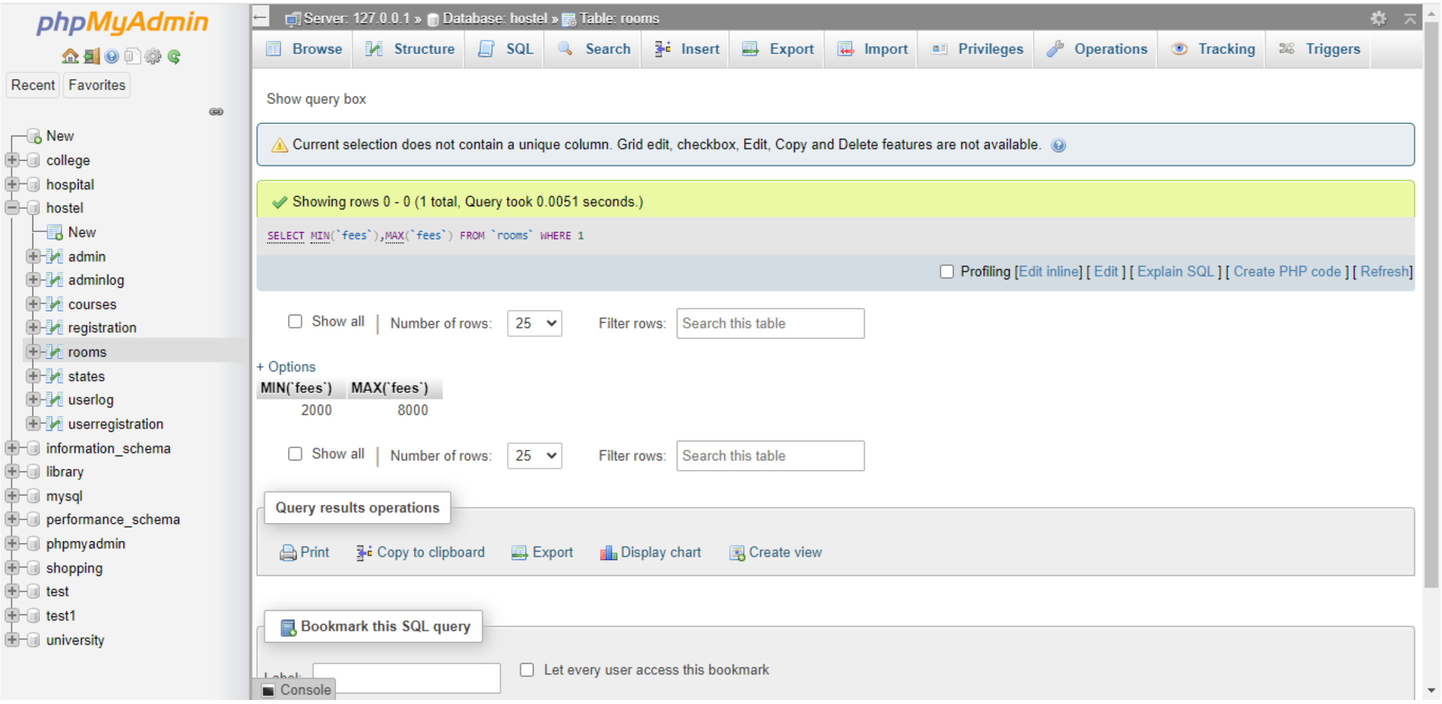
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
+ Options

←T→	id	regNo	firstName	middleName	lastName	gender	contactNo	email	password	regDate	u
□ Edit Copy Delete	6	1234	havish	reddy	P	male	4673864	vijaymullapudi54@gmail.com	admin	2020-12-10 14:18:51	N
□ Edit Copy Delete	45	34532	harika	mulapudi	chowdary	female	78938292	harika@gmail.com	harika1$23	2020-12-15 23:07:58	2

□ Check all With selected: Edit Copy Delete Export
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Query results operations
Print
Copy to clipboard
Export
Display chart
Create view
Bookmark this SQL query
Console

2)To view the students who have highest room rent and lowest rent in room table we run the sql query.

SELECT MIN(`fees`),MAX(`fees`) FROM `rooms` WHERE 1



⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 0 (1 total, Query took 0.0051 seconds.)

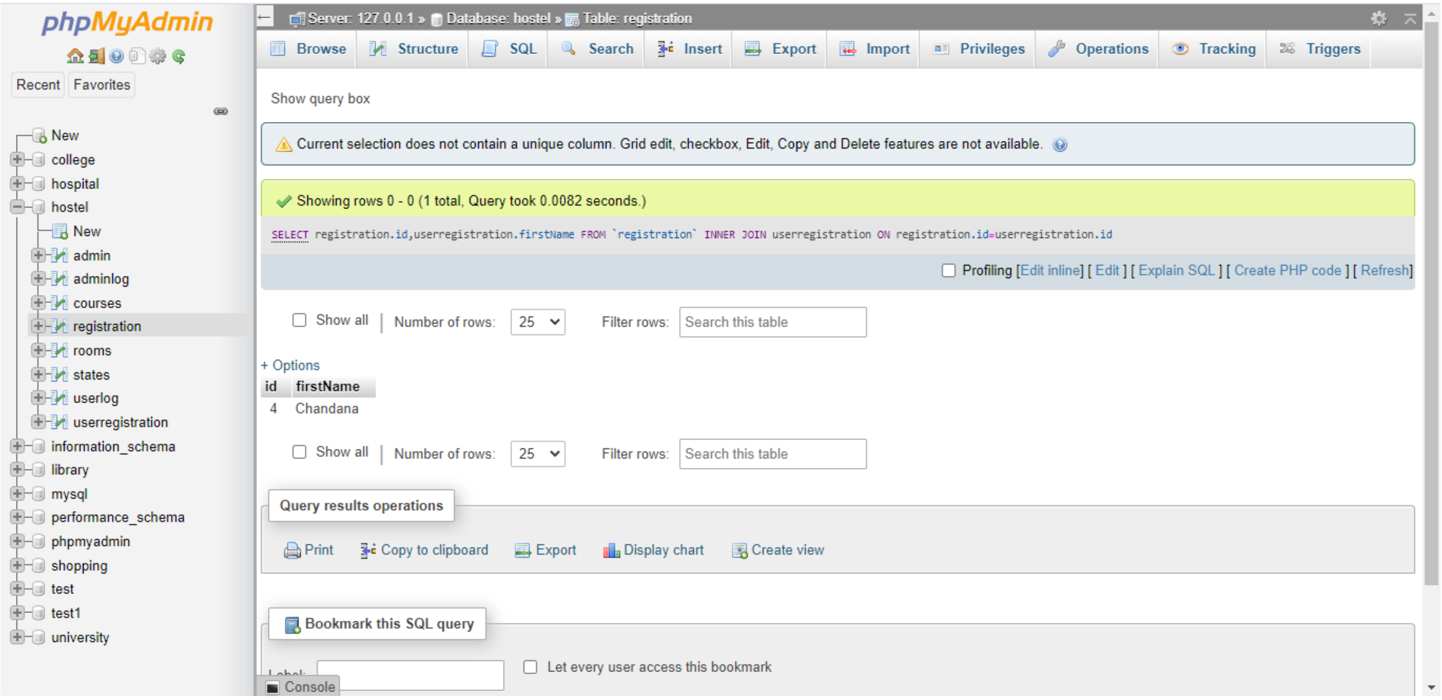
```
SELECT MIN(`fees`),MAX(`fees`) FROM `rooms` WHERE 1
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table
+ Options

MIN(`fees`)	MAX(`fees`)
2000	8000

3)To view the students who have user account and registered as well in hostel we run the sql query.

SELECT registration.id,userregistration.firstName FROM `registration` INNER JOIN userregistration ON registration.id=userregistration.id



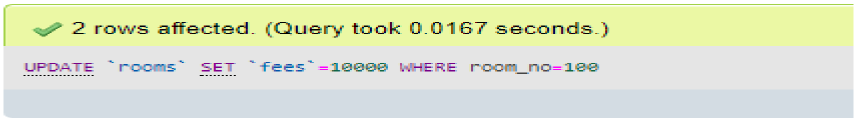
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 0 (1 total, Query took 0.0082 seconds.)

```
SELECT registration.id,userregistration.firstName FROM `registration` INNER JOIN userregistration ON registration.id=userregistration.id
```

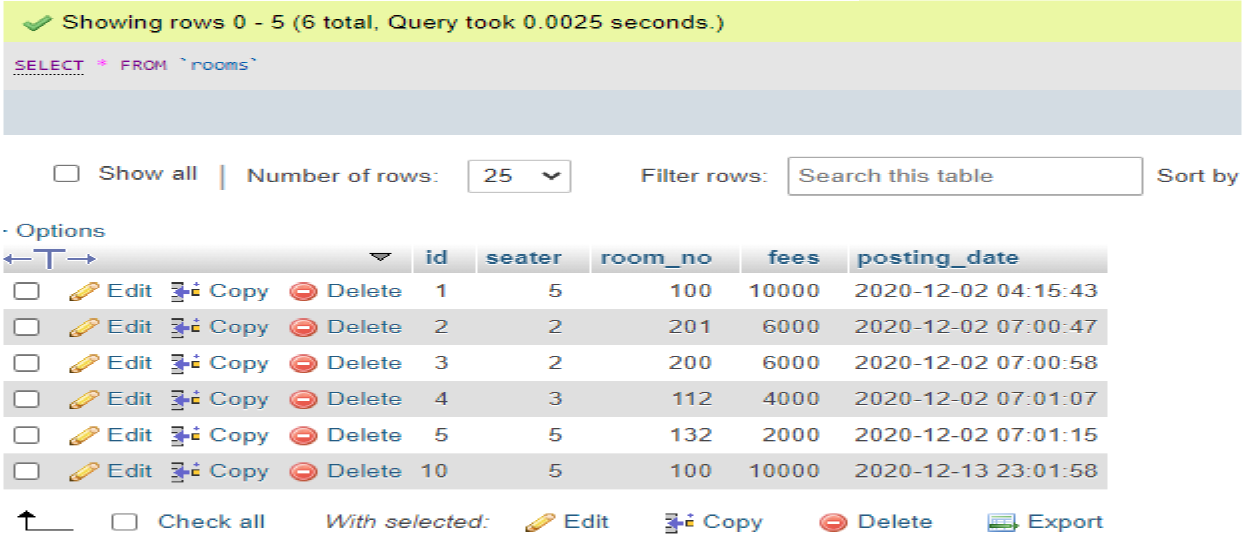
id	firstName
4	Chandana

4)For the admin to update the fees of a particular room we run the sql query



✓ 2 rows affected. (Query took 0.0167 seconds.)

```
UPDATE `rooms` SET `fees`=10000 WHERE room_no=100
```



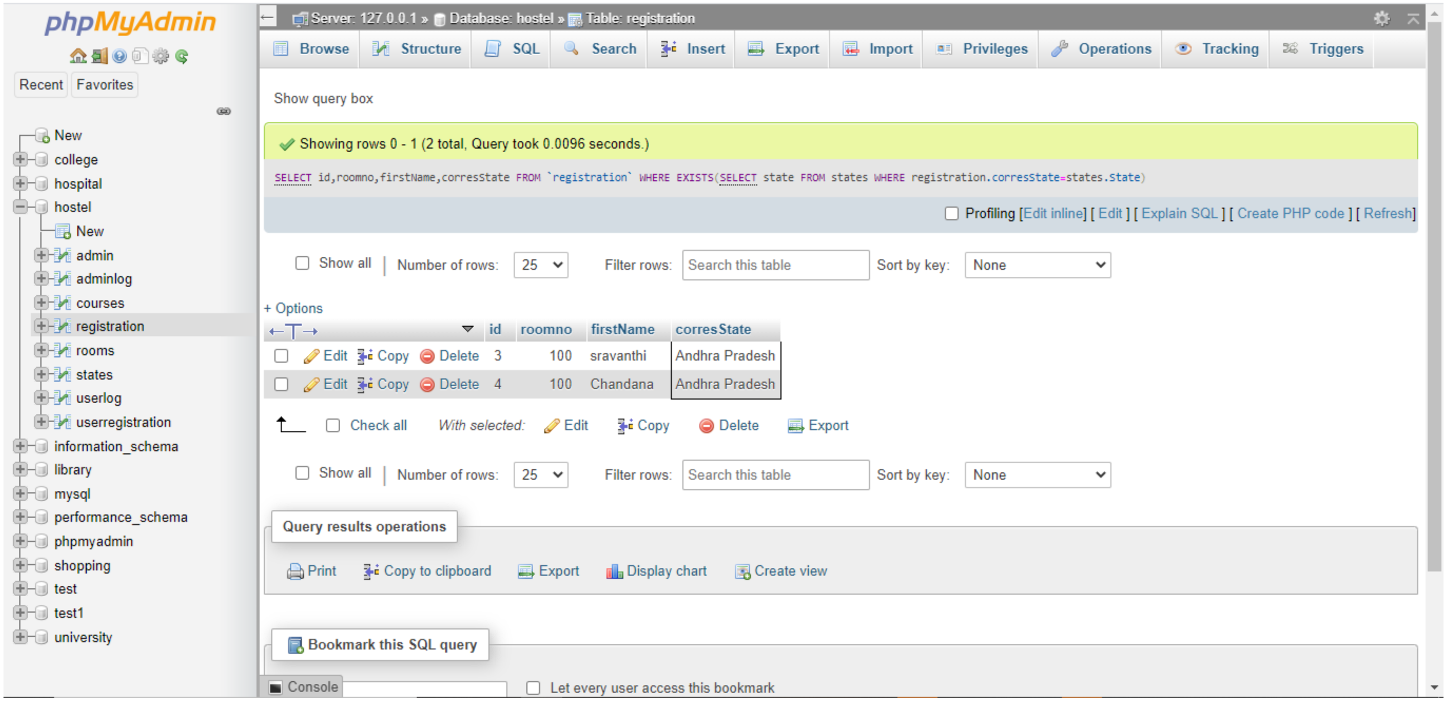
✓ Showing rows 0 - 5 (6 total, Query took 0.0025 seconds.)

```
SELECT * FROM `rooms`
```

	id	seater	room_no	fees	posting_date
□ Edit Copy Delete	1	5	100	10000	2020-12-02 04:15:43
□ Edit Copy Delete	2	2	201	6000	2020-12-02 07:00:47
□ Edit Copy Delete	3	2	200	6000	2020-12-02 07:00:58
□ Edit Copy Delete	4	3	112	4000	2020-12-02 07:01:07
□ Edit Copy Delete	5	5	132	2000	2020-12-02 07:01:15
□ Edit Copy Delete	10	5	100	10000	2020-12-13 23:01:58

5)To view the students who have exists in the states of India we run the sql query.

SELECT id,roomno,firstName,corresState FROM `registration` WHERE EXISTS(SELECT state FROM states WHERE registration.corresState=states.State)



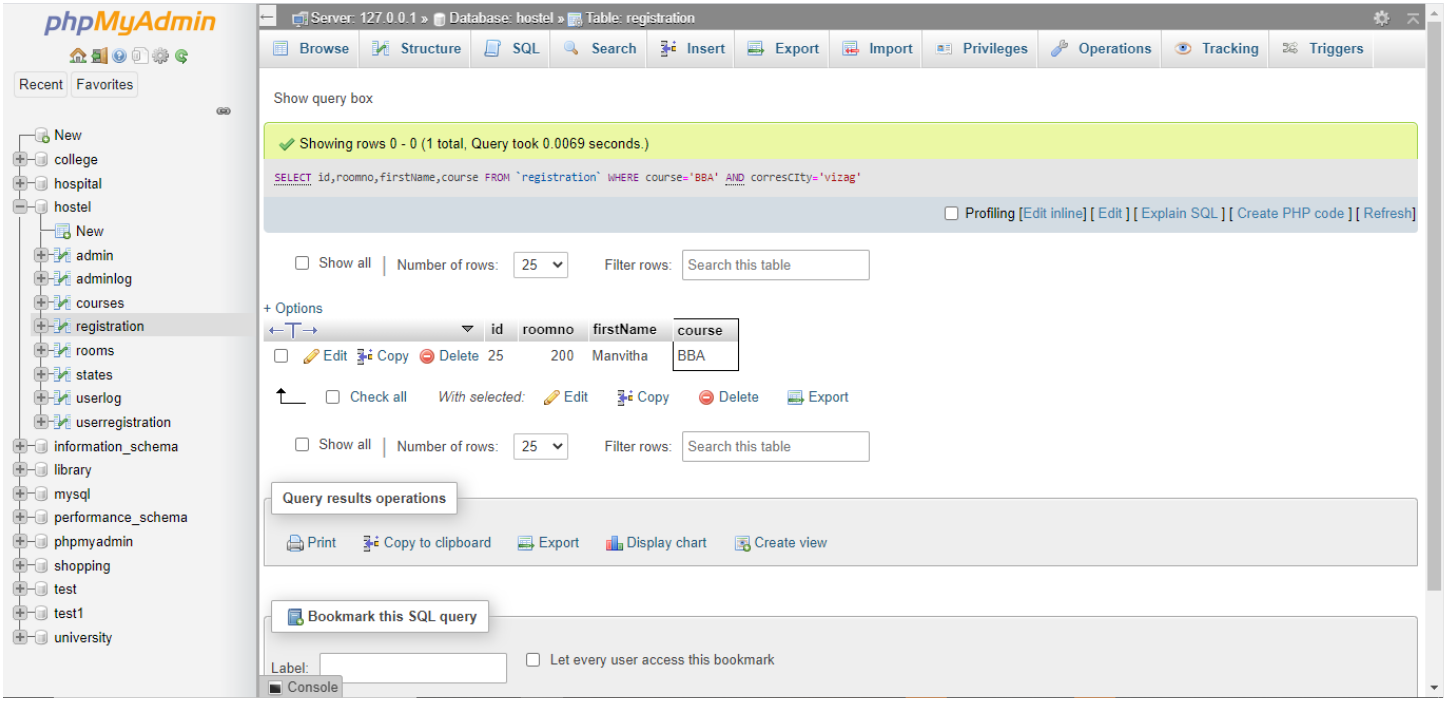
✓ Showing rows 0 - 1 (2 total, Query took 0.0096 seconds.)

```
SELECT id,roomno,firstName,corresState FROM `registration` WHERE EXISTS(SELECT state FROM states WHERE registration.corresState=states.State)
```

	id	roomno	firstName	corresState
□ Edit Copy Delete	3	100	sravanthi	Andhra Pradesh
□ Edit Copy Delete	4	100	Chandana	Andhra Pradesh

6)To view the students who study BBAwe run the sql query.

SELECT id,roomno,firstName,course FROM `registration` WHERE course='BBA' AND corresCIty='vizag'



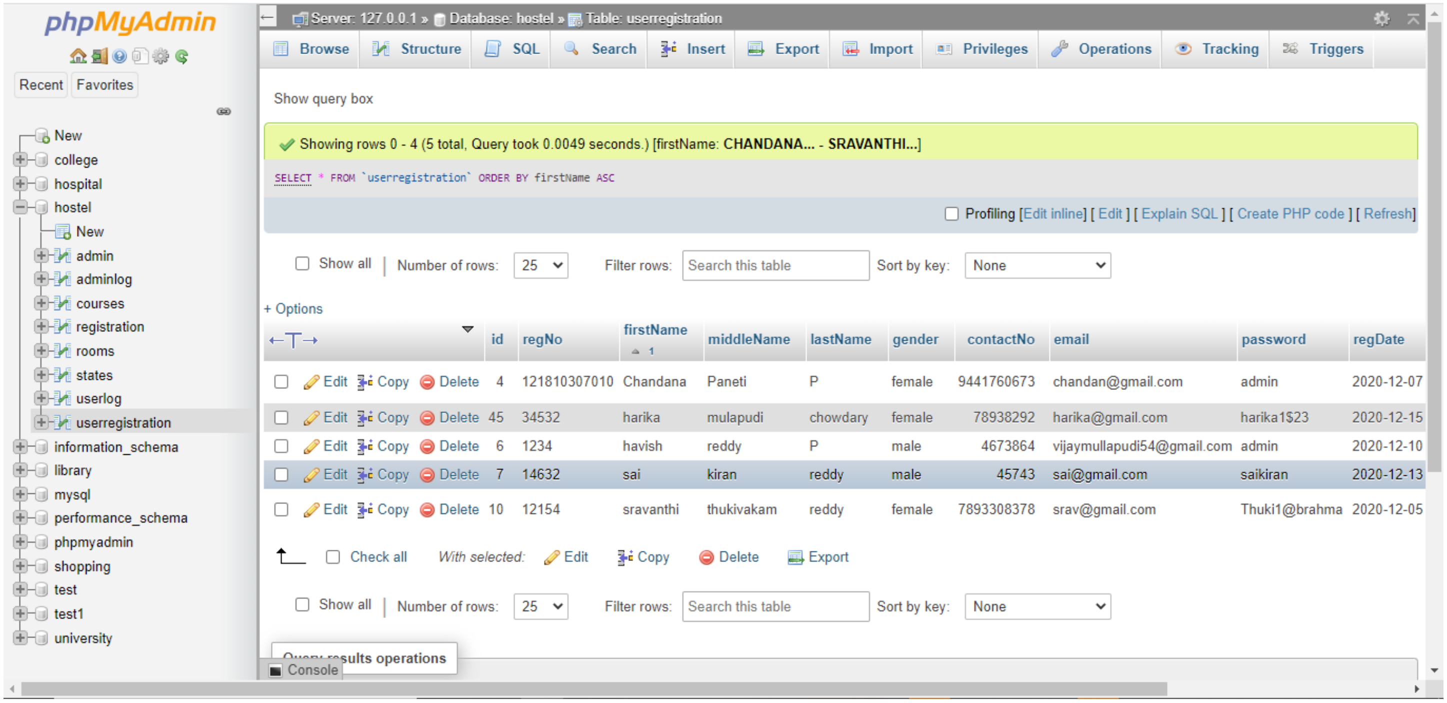
✓ Showing rows 0 - 0 (1 total, Query took 0.0069 seconds.)

```
SELECT id,roomno,firstName,course FROM `registration` WHERE course='BBA' AND corresCIty='vizag'
```

	id	roomno	firstName	course
□ Edit Copy Delete	25	200	Manvitha	BBA

7)To view the students by ascending name order we run the sql query.

SELECT * FROM `userregistration` ORDER BY firstName ASC



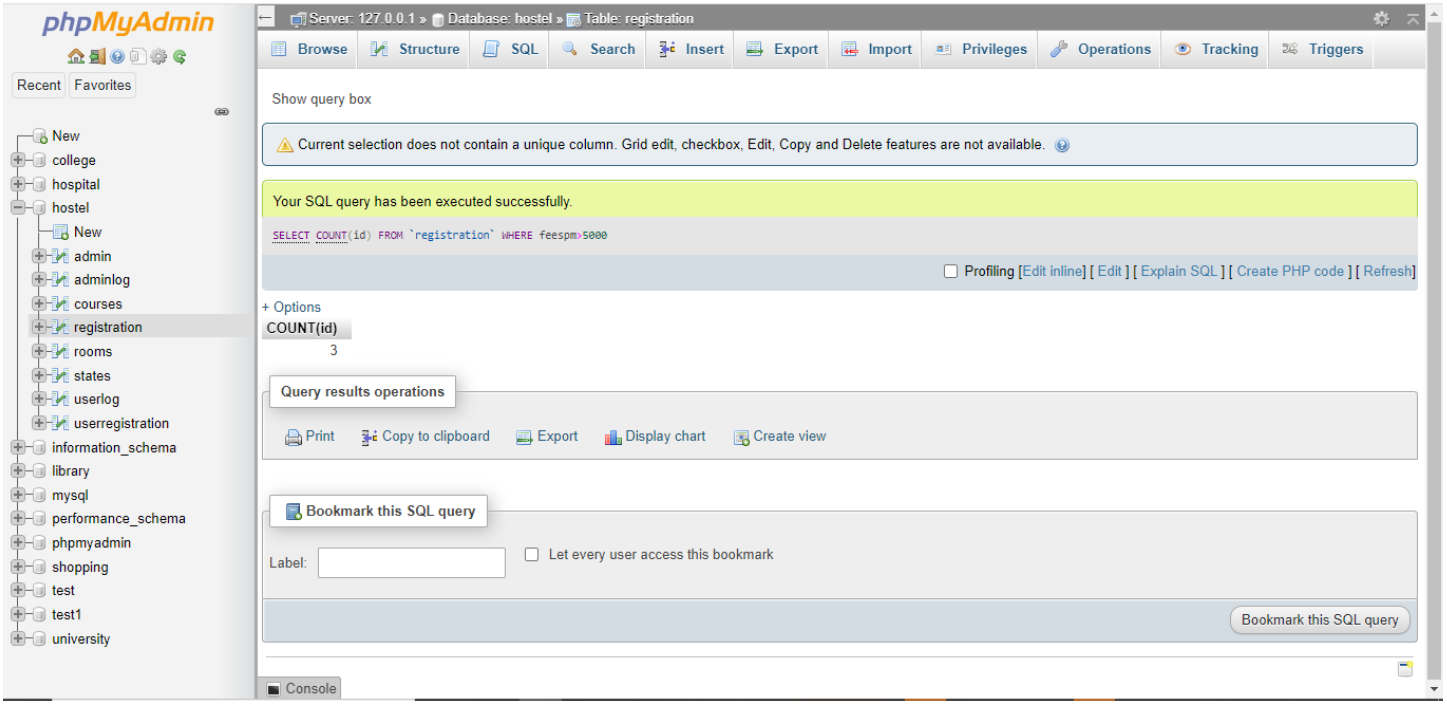
✓ Showing rows 0 - 4 (5 total, Query took 0.0049 seconds.) [firstName: CHANDANA... - SRAVANTHI...]

```
SELECT * FROM `userregistration` ORDER BY firstName ASC
```

	id	regNo	firstName	middleName	lastName	gender	contactNo	email	password	regDate
□ Edit Copy Delete	4	121810307010	Chandana	Paneti	P	female	9441760673	chandan@gmail.com	admin	2020-12-07
□ Edit Copy Delete	45	34532	harika	mulapudi	chowdary	female	78938292	harika@gmail.com	harika1$23	2020-12-15
□ Edit Copy Delete	6	1234	havish	reddy	P	male	4673864	vijaymullapudi54@gmail.com	admin	2020-12-10
□ Edit Copy Delete	7	14632	sai	kiran	reddy	male	45743	sai@gmail.com	saikiran	2020-12-13
□ Edit Copy Delete	10	12154	sravanthi	thukivakam	reddy	female	7893308378	srav@gmail.com	Thuki1@brahma	2020-12-05

8)To count the students who are having roomrent more than 5000 we run the sql query.

SELECT COUNT(id) FROM `registration` WHERE feespm>5000



⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Your SQL query has been executed successfully.

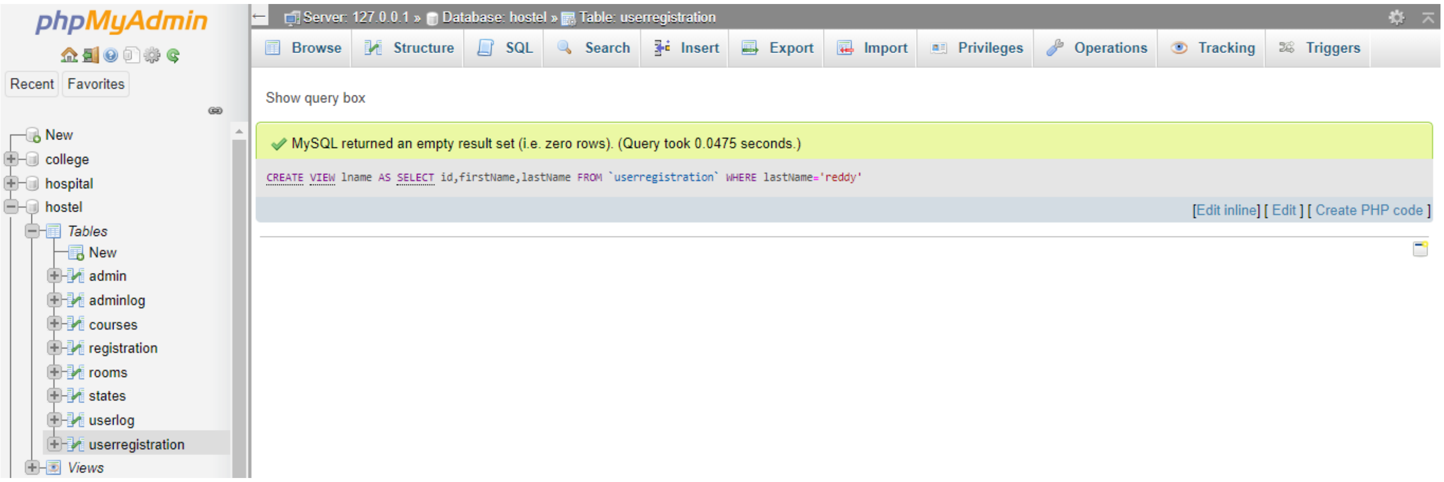
```
SELECT COUNT(id) FROM `registration` WHERE feespm>5000
```

COUNT(id)
3

9)To create a view of students whose lastname is reddy we run the sql query.

CREATING A VIEW

CREATE VIEW lname AS SELECT id,firstname,lastname FROM ‘userregistration’ WHERE lastname=’reddy’

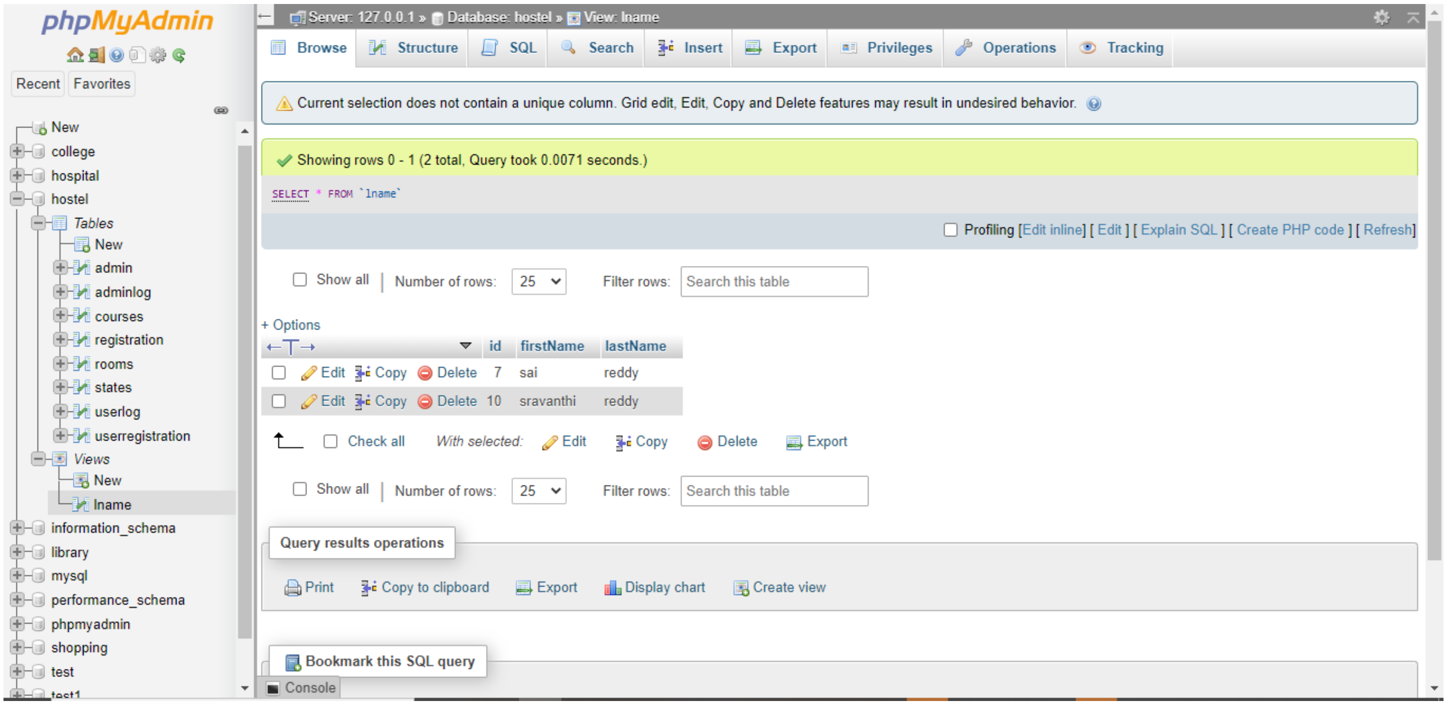


✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0475 seconds.)

```
CREATE VIEW lname AS SELECT id,firstName,lastName FROM `userregistration` WHERE lastName='reddy'
```

VIEWING THE VIEW LNAME

SELECT * FROM `lname`



⚠ Current selection does not contain a unique column. Grid edit, Edit, Copy and Delete features may result in undesired behavior.

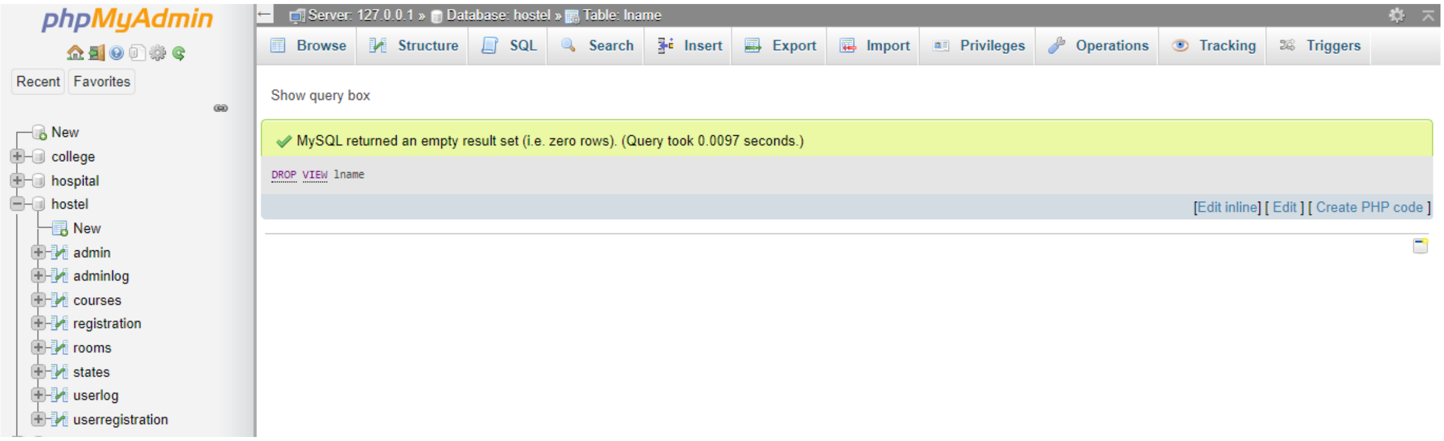
✓ Showing rows 0 - 1 (2 total, Query took 0.0071 seconds.)

```
SELECT * FROM `lname`
```

	id	firstName	lastName
□ Edit Copy Delete	7	sai	reddy
□ Edit Copy Delete	10	sravanthi	reddy

10)To drop a view we run the sql query.

DROP VIEW lname



MySQL returned an empty result set (i.e. zero rows). (Query took 0.0097 seconds.)

```
DROP VIEW lname
```

CONCLUSION

To conclude the description about the project : The project, developed using PHP and MySQL is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement.

The expanded functionality of today's software requires an appropriate approach towards software development. This hostel management software is designed for people who want to manage various activi- ties in the hostel. For the past few years the number of educational institutions are increasing rapidly.

Thereby the number of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software's are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually.

Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented.

BIBILIOGRAPHY

1. www.w3schools.com

*2. in.**php**.net*

3.** en.wikipedia.org/wiki/**PHP

4 . www.hotscripts.com/category/**php**/

5. www.**apache**.org/

6. www.**mysql**.com/click.php?e=35050