

Visveswaraya Technological University Belagavi – 590018, Karnataka



A Mini Project Report on “APARTMENT MANAGEMENT DATABASE”

Mini Project Report submitted in partial fulfilment of the requirement for the
DBMS Laboratory with Mini Project [18CSL58]

**Bachelor of Engineering
In
Computer Science and Engineering**

**Submitted By
Aditya Krishnan
1JT19CS004
Abhishek Kumar
1JT19CS002**



**Department of Computer Science and Engineering
Jyothy Institute of Technology,
Tataguni, Bengaluru – 560082**

**Department of Computer Science and Engineering
Jyothy Institute of Technology,
Tataguni, Bengaluru – 560082**



CERTIFICATE

Certified that the mini project work entitled "**APARTMENT MANAGEMENT DATABASE**" carried out by **Aditya Krishnan [1JT19CS004]** and **Abhishek Kumar [1JT19CS002]** bonafide students of Jyothy Institute of Technology, in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** department of the **Visvesvaraya Technological University, Belagavi** during the year **2019-2020**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree

Mr. Arun Kumar N
Guide, Asst. Professor
Dept. of CSE

External Viva Examiner
1.
2.

Dr. Prabhanjan S
Professor & HOD
Dept. of CSE

Signature with Date:

ACKNOWLEDGEMENT

Firstly, we are very grateful to this esteemed institution “**Jyothy Institute of Technology**” for providing us an opportunity to complete our project.

We express our sincere thanks to our **Principal Dr. Gopalakrishna K** for providing us with adequate facilities to undertake this project.

We would like to thank **Dr. Prabhanjan S, Professor and Head of Computer Science and Engineering Department** for providing for his valuable support.

We would like to thank our guides **Mr. Arun Kumar N , Assistant Professor** for their keen interest and guidance in preparing this work.

Finally, we would thank all our friends who have helped us directly or indirectly in this project.

**Aditya Krishnan [1JT19CS004]
Abhishek Kumar [1JT19CS002]**

ABSTRACT

The aim of this project is to create a functional application to manage the daily operations of apartments and simplify various aspects of apartment management such as rent collection, booking an apartment, availability of apartments etc that are normally very tedious to maintain using traditional methods.

This software helps them to digitize their records, which in turn saves a lot of time and money. For a manager or a company that owns multiple apartments, keeping track of each one is very difficult. They need to note down and maintain every detail for every apartment, keep track of apartments that are already booked, yet to be booked, payment status, etc. Expansion also bring in many challenges as their existing data need to be modified to reflect any additions.

In the proposed system, each tenant has his/her details stored in the database. The monthly rent can be entered for each resident which can save time. With this application, enable property managers to keep track and maintain records of their entire property easily. Any important information such as available blocks, flats, details of residents, rent paid can be found with just a click of a button.

TABLE OF CONTENTS

Sl No	Description	Page No
1	INTRODUCTION	6
2	DESIGN	9
3	IMPLEMENTATION & SNAPSHOT	13
4	CONNECTION WITH PHP MYADMIN	28
5	QUERIES RELATED SNAPSHOTS	35
6	CONCLUSION	40

CHAPTER 1

INTRODUCTION

INTRODUCTION

1.1 Introduction to DBMS

A database is simply an organized collection of related data, typically stored on disk, and accessible by many concurrent users, it is a logically coherent collection of data with some inherent meaning, representing some aspect of real world and which is designed, built and populated with data for a specific purpose.

Databases are managed by a Database Management System(DBMS) which is a collection of programs that enables user to create and maintain a database.

Advantages of DBMS:

- Redundancy is controlled.
- Unauthorized access is restricted.
- Providing multiple user interfaces.
- Enforcing integrity constraints.
- Providing backup and recovery.

1.2 Introduction to SQL

Structured Query Language (SQL), is a language used to request data from a database which includes database creation, deletion, retrieval of required tables and even manipulation of data held in a relational database management system.

SQL is considered as a Non-Procedural or a High level language in which the expected result or operation is given without the specific details about how to accomplish the task. So, SQL is a declarative language.

Therefore, SQL is designed at a higher conceptual level of operation than procedural languages as procedural languages includes only the information about opening and closing tables, loading and searching indexes, or flushing buffers and writing data to file systems, but the lower level logical and physical operations are not specified in SQL.

1.3 Introduction to Apartment Management Database

An apartment is a residential building, consisting of many units. Each unit is owned by the apartment owner, who leases it out to tenants.

The “Apartment Database” is a database that has information about each tenant and his respective apartment. It manages booking, expenses calculation and rent collection through a simple interface.

1.4 Scope and importance of work

The scope of the project is to create a webpage than enables users to overcome the drawbacks of traditional record keeping systems.

The database stores details about the tenant – name, phone number and dependents.

The apartment leased to the tenant is also recorded, along with the type of apartment and the respective block. Apartments have expenses such maintenance charges, repair charges and security charges, which are stored in the database for easy access. Rent due for each month is set by the owner.

CHAPTER 2

DESIGN

Theory of ER Diagram

The Entity–Relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An Entity Relationship Diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of database. ER diagrams are used to sketch out the design of a database.

ENTITIES

An entity is an ‘object’ in the real world with an independent existence and an entity type defines a collection (or set) of entities that have the same attributes. Each entity type in the database is described by its name and attributes. An entity type is represented in ER diagrams as a rectangular box enclosing the entity type name.

RELATIONSHIPS

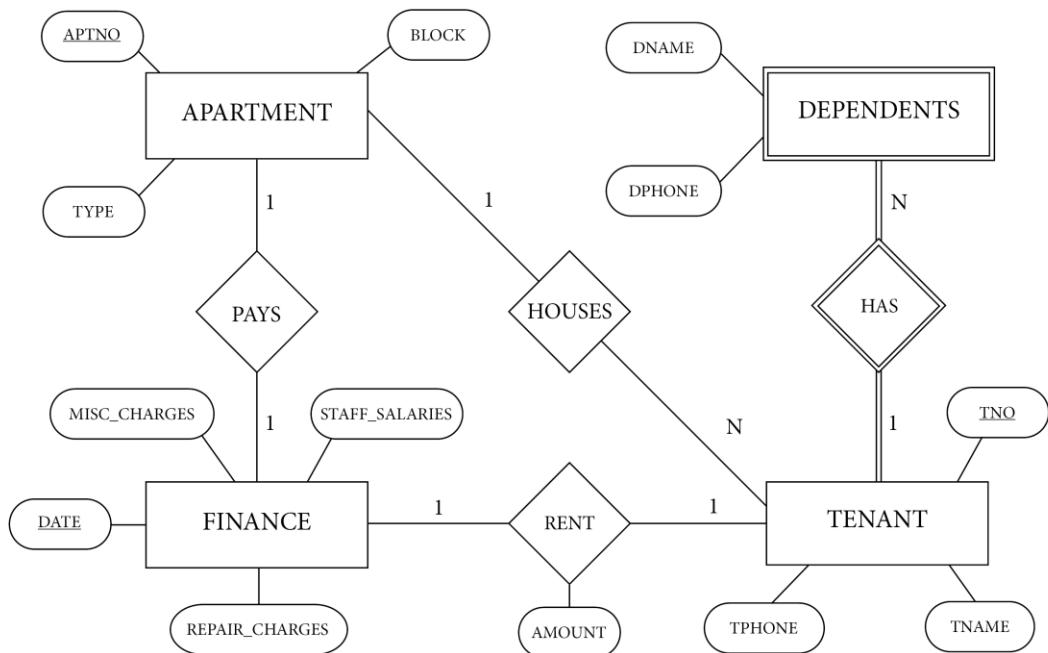
A relationship among two or more entities represents an association among the entities and whenever an attribute of one entity refers to another entity, there exists a relationship between the two entities.

In a relationship, a foreign key of one table refers the primary key of the other table and it is represented by diamond shape in ER diagram.

ATTRIBUTES

An attribute represents some property of interest that further describes an entity and the column header of the table shows the attributes. Each attribute in a table has a certain domain which allows it to accept a certain ‘set of values’ only. The attribute values, of each entity, will define its characteristics in the table and is represented by oval in the ER diagram.

ENTITY RELATIONSHIP DIAGRAM



SCHEMA DIAGRAM

LIST OF TABLES

APARTMENTS:

- APTN
- TYPE
- BLOCK

TENANT :

- TNO
- TNAME
- TPHONE
- APTN

DEPENDENTS :

- TNO
- DNAME
- DPHONE

RENT :

- DATE
- TNO
- AMOUNT

FINANCE :

- DATE
- REPAIR_CHARGES
- STAFF_SALARIES
- MISC_CHARGES
- APTNO

CHAPTER 3

IMPLEMENTATION &

SNAPSHOT

CREATION:

- 1 CREATE TABLE tenant (TNO INTEGER(5) PRIMARY KEY, TNAME VARCHAR(40), TPHONE INTEGER(10));
- 2 CREATE TABLE apartment (APTNO INTEGER(4) PRIMARY KEY, TNO INTEGER(5), FOREIGN KEY(TNO) REFERENCES TENANT(TNO) ON DELETE CASCADE, TYPE VARCHAR(20), BLOCK INTEGER(2));
- 3 CREATE TABLE dependent (TNO INTEGER(5), FOREIGN KEY(TNO) REFERENCES TENANT(TNO) ON DELETE CASCADE, DNAME VARCHAR(20), DPHONE INTEGER(10));
- 4 CREATE TABLE finance (DATE DATE PRIMARY KEY, REPAIR_CHARGES INTEGER(11), STAFF_SALARIES INTEGER(11), MISC_CHARGES INTEGER(11), APTNO INTEGER(5), FOREIGN KEY(TNO) REFERENCES APRTMENT(APTNO) ON DELETE CASCADE);
- 5 CREATE TABLE rent (DATE DATE, FOREIGN KEY(DATE) REFERENCES FINANCE(TNO) ON DELETE CASCADE, TNO INTEGER(5), FOREIGN KEY(TNO) REFERENCES TENANT(TNO) ON DELETE CASCADE);

INSERTION:

```
INSERT INTO apartment(`APTNO`, `TNO`, `TYPE`, `BLOCK`) VALUES  
(1001, 1, '9 bhk', 9),  
(1002, 2, '2BHK', 1),  
(1003, 3, '3BHK', 3),  
(1051, 5, '2 BHK', 2),  
(1078, 4, '4BHK', 1);
```

```
INSERT INTO dependent(`TNO`, `DNAME`, `DPHONE`) VALUES  
(1, 'harsh soni', 2147483647),
```

(2, 'M SUMESH', 2147483647),
(3, 'R NAMBIAR', 70787889),
(4, 'DHIRU M', 860394489),
(5, 'M SINGH', 60987867);

INSERT INTO finance (`DATE`, `REPAIR_CHARGES`,
 `STAFF_SALARIES`, `MISC_CHARGES`, `APTNO`) VALUES
 ('0000-00-00', 340, 120, 250, 1002),
 ('2021-01-12', 1200, 500, 270, 1001),
 ('2022-01-27', 1208, 1201, 1110, 1003),
 ('2022-01-29', 1500, 1234, 1000, 1078),
 ('2022-01-31', 1000, 1110, 1200, 1051);

INSERT INTO rent (`DATE`, `TNO`, `AMOUNT`) VALUES
 ('2021-01-12', 1, 20000),
 ('0000-00-00', 2, 23000),
 ('2022-01-27', 3, 25000),
 ('2022-01-29', 4, 35000),
 ('2022-01-31', 5, 20000);

INSERT INTO tenant (`TNO`, `TNAME`, `TPHONE`) VALUES
 (1, 'poorvi soni', 898765544),
 (2, 'Ajin Sumesh', 2147483647),
 (3, 'ADITYA KRISHNAN', 787971756),
 (4, 'VINAY M', 89876574),
 (5, 'BHUPINDER SINGH', 98887665);

localhost / 127.0.0.1 / apt_mgmt | Apartment Management System | Apartment Management System | +

localhost/phpmyadmin/index.php?route=table/structure&server=1&db=apt_mgmt&table=tenant

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	TNO	int(5)	utf8mb4_general_ci	No	None				Change Drop More
2	TNAME	varchar(40)	utf8mb4_general_ci	Yes	NULL				Change Drop More
3	TPHONE	int(10)		Yes	NULL				Change Drop More

Check all With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

Add 1 column(s) after TPHONE [Go](#)

Indexes

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

Create an index on 1 columns [Go](#)

Partitions

No partitioning defined!

Partition table

Information

Data	16.0 KIB	Format	dynamic
Index	0 B	Collation	utf8mb4_general_ci
Overhead	8.0 MIB	Next autoindex	0
Effective	-8,372,224 B	Creation	Jan 07, 2022 at 03:34 PM
Total	16.0 KIB	Last update	Jan 28, 2022 at 10:15 PM
Optimize table		Last check	Jan 28, 2022 at 05:54 PM
Space usage			Row statistics

[Console](#)

localhost / 127.0.0.1 / apt_mgmt | Apartment Management System | Apartment Management System | +

localhost/phpmyadmin/index.php?route=table/structure&server=1&db=apt_mgmt&table=apartment

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	APTNO	int(4)	utf8mb4_general_ci	No	None				Change Drop More
2	TNO	int(5)	utf8mb4_general_ci	Yes	NULL				Change Drop More
3	TYPE	varchar(20)	utf8mb4_general_ci	Yes	NULL				Change Drop More
4	BLOCK	int(2)		Yes	NULL				Change Drop More

Check all With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

Add 1 column(s) after BLOCK [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	APTNO	5	A	No	
Edit Rename Drop	TNO	BTREE	No	No	TNO	5	A	Yes	

Create an index on 1 columns [Go](#)

Partitions

No partitioning defined!

Partition table

Information

Data	16.0 KIB	Format	dynamic
Index	16.0 KIB	Collation	utf8mb4_general_ci
Overhead	0 B	Next autoindex	0
Effective	32.0 KIB	Creation	Jan 07, 2022 at 03:34 PM
Total	32.0 KIB	Last update	Jan 28, 2022 at 10:15 PM
Optimize table		Last check	Jan 28, 2022 at 05:54 PM
Space usage			Row statistics

[Console](#)

Screenshot of phpMyAdmin interface showing the structure of the 'dependent' table in the 'apt_mgmt' database.

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	TNO	int(5)	utf8mb4_general_ci	Yes	NULL				Change Drop More
2	DNAME	varchar(40)	utf8mb4_general_ci	Yes	NULL				Change Drop Primary Unique Index Spatial Fulltext
3	DPHONE	int(10)	utf8mb4_general_ci	Yes	NULL				Change Drop More

Add: Add 1 column(s) after DPHONE [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	TNO	BTREE	No		TNO	5	A	Yes	

Create an index on 1 columns [Go](#)

Partitions

No partitioning defined!

Partition table

Information

Data	16.0 KB	Format	dynamic
Index	16.0 KB	Collation	utf8mb4_general_ci
Overhead	0 B	Next autoindex	0
Effective	32.0 KB	Creation	Jan 07, 2022 at 03:34 PM
Total	32.0 KB	Last update	Jan 28, 2022 at 10:15 PM
Optimize table			
Last check Jan 28, 2022 at 05:55 PM			
Space usage Row statistics			

Screenshot of phpMyAdmin interface showing the structure of the 'finance' table in the 'apt_mgmt' database.

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	DATE	date	utf8mb4_general_ci	No	None				Change Drop More
2	REPAIR_CHARGES	int(11)	utf8mb4_general_ci	Yes	NULL				Change Drop More
3	STAFF_SALARIES	int(11)	utf8mb4_general_ci	Yes	NULL				Change Drop More
4	MISC_CHARGES	int(11)	utf8mb4_general_ci	Yes	NULL				Change Drop More
5	APTNNO	int(4)	utf8mb4_general_ci	Yes	NULL				Change Drop More

Add: Add 1 column(s) after APTNNO [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	DATE	5	A	No	
Edit Rename Drop	APTNNO	BTREE	No	No	APTNNO	5	A	Yes	

Create an index on 1 columns [Go](#)

Partitions

No partitioning defined!

Partition table

Information

Data	16.0 KB	Format	dynamic
Index	16.0 KB	Collation	utf8mb4_general_ci
Overhead	0 B	Next autoindex	0
Effective	32.0 KB	Creation	Jan 07, 2022 at 03:34 PM
Total	32.0 KB	Last update	Jan 28, 2022 at 10:16 PM
Optimize table			
Last check Jan 28, 2022 at 05:55 PM			
Space usage Row statistics			

localhost / 127.0.0.1 / apt_mgmt | Apartment Management System | Apartment Management System | +

localhost/phpmyadmin/index.php?route=/table/structure&server=1&db=apt_mgmt&table=rent

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	DATE	date	utf8mb4_general_ci		Yes	NULL			Change Drop More
2	TNO	int(5)	utf8mb4_general_ci		Yes	NULL			Change Drop More
3	AMOUNT	int(11)	utf8mb4_general_ci		Yes	NULL			Change Drop Primary Index Unique Spatial Fulltext

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) 1 column(s) after AMOUNT [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	DATE	BTREE	No	No	DATE	5	A	Yes	
Edit Rename Drop	TNO	BTREE	No	No	TNO	5	A	Yes	

Create an index on 1 columns [Go](#)

Partitions

No partitioning defined

Partition table

Information

Data	16.0 KB	Format	dynamic
Index	32.0 KB	Collation	utf8mb4_general_ci
Overhead	0 B		
Effective	48.0 KB	Creation	Jan 07, 2022 at 03:34 PM
Total	48.0 KB	Last update	Jan 28, 2022 at 10:16 PM
Optimize table			
Last check: Jan 28, 2022 at 05:56 PM			
Space usage			
Row statistics			

[Console](#)

localhost / 127.0.0.1 / apt_mgmt | Apartment Management System | Apartment Management System | +

localhost/phpmyadmin/index.php?route=/table/structure&server=1&db=apt_mgmt&table=contact_us

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Name	text	utf8mb4_general_ci		No	None			Change Drop More
2	Email	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
3	Subject	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
4	Message	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) 1 column(s) after Message [Go](#)

Indexes

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

Create an index on 1 columns [Go](#)

Partitions

No partitioning defined

Partition table

Information

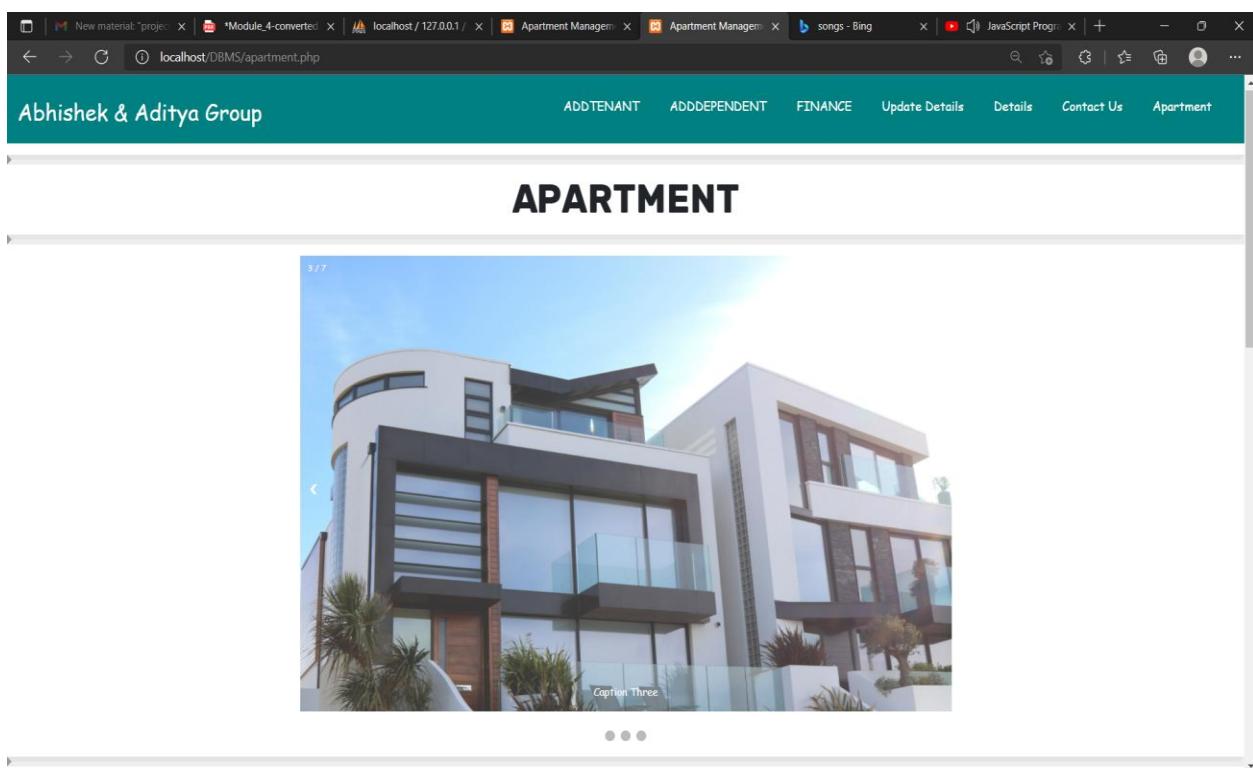
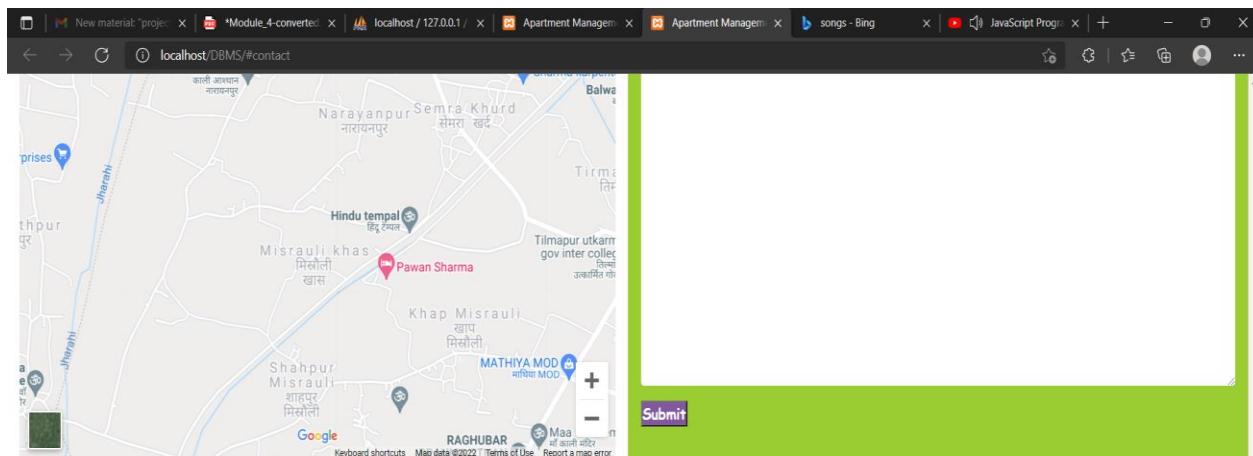
Data	16.0 KB	Format	dynamic
Index	0 B	Collation	utf8mb4_general_ci
Overhead	0 B		
Effective	16.0 KB	Creation	Jan 08, 2022 at 01:35 PM
Total	16.0 KB	Last update	Jan 28, 2022 at 05:56 PM
Optimize table			
Last check: Jan 28, 2022 at 05:56 PM			
Space usage			
Row statistics			

[Console](#)

HOME PAGE

The screenshot shows a web browser window with multiple tabs open. The main content area displays a large grid of apartment photographs. The top navigation bar includes links for 'ADD TENANT', 'ADD DEPENDENT', 'FINANCE', 'Update Details', 'Details', 'Contact Us', and 'Apartment'.

The screenshot shows a web browser window with multiple tabs open. The main content area features a 'Contact Us' heading. To the left is a map of Khalwa, Bihar, showing various landmarks and locations. To the right is a green-themed contact form with fields for 'Name', 'Email', 'Subject', and 'Message'.



ADD TENANT

The screenshot shows a web browser window with a teal header bar. The header contains the text "Abhishek & Aditya Group" on the left and navigation links "ADDTENANT", "ADDEPENDENT", "FINANCE", "Update Details", "Details", "Contact Us", and "Apartment" on the right. Below the header, the main content area has a title "Add New Tenant" and a sub-section titled "Enter New Tenant Details". The form fields are filled with the following data:

Name	RAMKUWAR SINGH
Apartment Number	1006
Phone	8603944898
Type	3BHK
Block	3

A "Submit" button is located at the bottom of the form. Below the form, there is a "Contact Us" section.

DATA INSERTED SUCCESSFULLY

The screenshot shows a web browser window with a teal header bar. The header contains the text "Abhishek & Aditya Group" on the left and navigation links "ADDTENANT", "ADDEPENDENT", "FINANCE", "Update Details", "Details", "Contact Us", and "Apartment" on the right. Below the header, the main content area has a title "Add New Tenant" and a sub-section titled "Enter New Tenant Details". All the form fields are currently empty, showing placeholder text such as "Enter the Tenant Name", "Enter The Apartment Number", "Enter The Phone Number", "Enter The Type", and "Enter The Block". A "Submit" button is located at the bottom of the form. Below the form, there is a "Contact Us" section.

Screenshot of phpMyAdmin showing the tenant table in the apt_mgmt database.

Table: tenant

Structure:

```

CREATE TABLE `tenant` (
  `TNO` int(11) NOT NULL,
  `TNAME` varchar(50) NOT NULL,
  `TPHONE` varchar(15) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Data:

TNO	TNAME	TPHONE
1	poonil soni	898765544
2	Ajin Sumesh	2147483647
3	arun kumar n	87676543
4	RAMKUWAR SINGH	2147483647

Screenshot of phpMyAdmin showing the apartment table in the apt_mgmt database.

Table: apartment

Structure:

```

CREATE TABLE `apartment` (
  `APTNO` int(11) NOT NULL,
  `TNO` int(11) NOT NULL,
  `TYPE` varchar(50) NOT NULL,
  `BLOCK` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Data:

APTNO	TNO	TYPE	BLOCK
4	3	3bhk	2
1001	1	9 bhk	9
1002	2	2BHk	1
1006	4	3BHK	3

ADD DEPENDENT

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "localhost/DBMS/adddependent.php". The page header "Abhishek & Aditya Group" is visible. Below the header, there are navigation links: ADDTENANT, ADDEPENDENT, FINANCE, Update Details, Details, Contact Us, and Apartment. The main content area has a title "Add Dependent" and a sub-section "Enter Dependent Details". Inside this section, there are three input fields: "Tenant Number" (containing "4"), "Dependent Name" (containing "SAHDEO SINGH"), and "Dependent Phone Number" (containing "78767574732"). A "Submit" button is at the bottom. Below this section, there is a "Contact Us" heading followed by a map and a green "Let us know" button.

DATA INSERTED SUCCESSFULLY

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "localhost/DBMS/adddependent.php". The page header "Abhishek & Aditya Group" is visible. Below the header, there are navigation links: ADDTENANT, ADDEPENDENT, FINANCE, Update Details, Details, Contact Us, and Apartment. A message "Data Inserted Successfully" is displayed above the "Add Dependent" section. The "Add Dependent" section and its "Enter Dependent Details" form are identical to the previous screenshot, showing the same input fields and layout. Below this, there is a "Contact Us" heading followed by a map and a green "Let us know" button.

phpMyAdmin

Database: apt_mgmt > Table: dependent

Showing rows 0 - 2 (3 total). Query took 0.0005 seconds.

SELECT * FROM `dependent`

TNO	DNAME	DPHONE
1	harsh soni	2147483647
2	M SUMESH	2147483647
4	SAHdeo SINGH	2147483647

ADD FINANCE

Abhishek & Aditya Group

ADDTENANT ADDDEPENDENT FINANCE Update Details Details Contact Us Apartment

Update Finance

Tenant Number: 4

Apartment Number: 1006

Staff Salary: 1201

Miscellaneous Charges: 1501

Repair Charges: 1250

Rent: 30000

Date (YYYY-MM-DD): 2022-01-24

Submit

DATA INSERTED SUCCESSFULLY

The screenshot shows a web browser window with multiple tabs open. The active tab displays a success message: "Data Inserted Successfully". Below this, there is a navigation bar with links: ADDTENANT, ADDDEPENDENT, FINANCE, Update Details, Details, Contact Us, and Apartment. The main content area features a title "Update Finance" and a form titled "Update Finance". The form contains fields for Tenant Number, Apartment Number, Staff Salary, Miscellaneous Charges, Repair Charges, Rent, and Date (YYYY-MM-DD). Each field has a placeholder text describing its purpose.

The screenshot shows the phpMyAdmin interface connected to a MySQL database named "apt_mgmt". The left sidebar shows the database structure with tables like "New", "admin", "apartment", "contact_us", "dependent", "finance", "rent", and "tenant". The right panel displays the "rent" table. A SQL query is shown at the top: "SELECT * FROM `rent`". The results table shows three rows of data:

DATE	TNO	AMOUNT
2021-01-12	1	20000
0000-00-00	2	23000
2022-01-24	4	30000

TENANT DETAILS

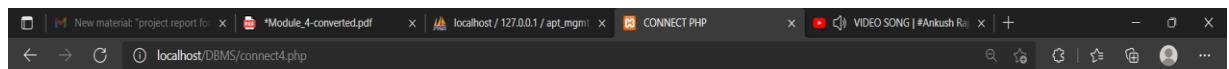
Tenant Details

Enter the Tenant Number
Tenant Number: 4
Submit

Delete tenant

Enter the Tenant Number
Enter The Tenant Number
Submit

Contact Us



TENANT NUMBER	TENANT NAME	TENANT PHONE
4	RAMKUWAR SINGH	2147483647

APARTMENT NUMBER	APARTMENT TYPE	APARTMENT BLOCK
1006	3BHK	3

DEPENDENT NAME	DEPENDENT PHONE
SAHDEO SINGH	2147483647

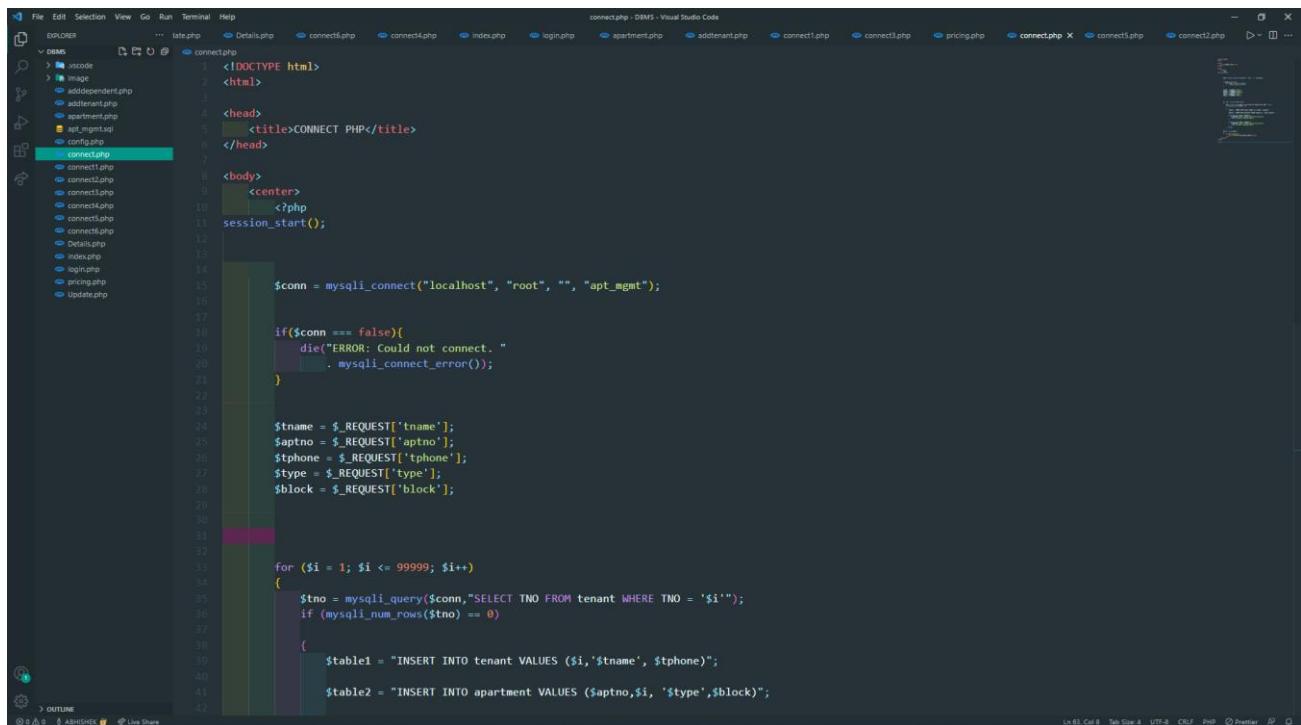
PAYMENT DATE	TOTAL AMOUNT
2022-01-24	30000

ADMIN

CHAPTER 4

CONNECTION WITH PHP

MYADMIN



```
<!DOCTYPE html>
<html>
<head>
<title>CONNECT PHP</title>
</head>
<body>
<center>
<?php
session_start();

$conn = mysqli_connect("localhost", "root", "", "apt_mgmt");

if($conn === false){
die("ERROR: Could not connect. "
. mysqli_connect_error());
}

$tno = $_REQUEST['tname'];
$aptno = $_REQUEST['aptno'];
$tphone = $_REQUEST['tphone'];
$type = $_REQUEST['type'];
$block = $_REQUEST['block'];

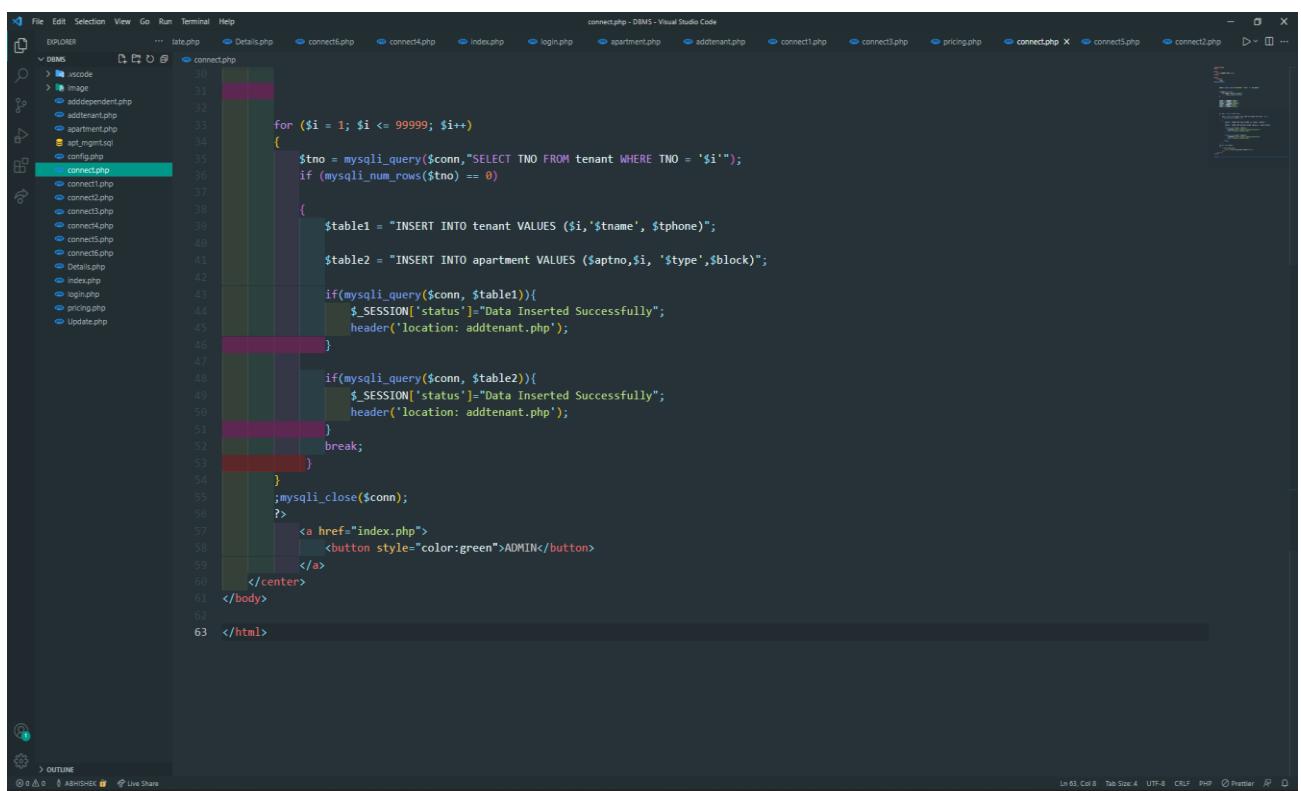
for ($i = 1; $i <= 99999; $i++)
{
    $tno = mysqli_query($conn,"SELECT TNO FROM tenant WHERE TNO = '$i'");
    if (mysqli_num_rows($tno) == 0)

    {
        $table1 = "INSERT INTO tenant VALUES ($i,$tname, $tphone)";

        $table2 = "INSERT INTO apartment VALUES ($aptno,$i, '$type',$block)";

        if(mysqli_query($conn, $table1)){
            $_SESSION['status']="Data Inserted Successfully";
            header('location: addtenant.php');
        }

        if(mysqli_query($conn, $table2)){
            $_SESSION['status']="Data Inserted Successfully";
            header('location: addtenant.php');
        }
        break;
    }
}
;mysqli_close($conn);
?>
<a href="index.php">
<button style="color:green">ADMIN</button>
</a>
</center>
</body>
</html>
```



```
<!DOCTYPE html>
<html>
<head>
<title>CONNECT PHP</title>
</head>
<body>
<center>
<?php
session_start();

for ($i = 1; $i <= 99999; $i++)
{
    $tno = mysqli_query($conn,"SELECT TNO FROM tenant WHERE TNO = '$i'");
    if (mysqli_num_rows($tno) == 0)

    {
        $table1 = "INSERT INTO tenant VALUES ($i,$tname, $tphone)";

        $table2 = "INSERT INTO apartment VALUES ($aptno,$i, '$type',$block)";

        if(mysqli_query($conn, $table1)){
            $_SESSION['status']="Data Inserted Successfully";
            header('location: addtenant.php');
        }

        if(mysqli_query($conn, $table2)){
            $_SESSION['status']="Data Inserted Successfully";
            header('location: addtenant.php');
        }
        break;
    }
}
;mysqli_close($conn);
?>
<a href="index.php">
<button style="color:green">ADMIN</button>
</a>
</center>
</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
    <title>CONNECT PHP</title>
</head>
<body>
    <center>
        <?php
        session_start();
        $conn = mysqli_connect("localhost", "root", "", "apt_mgmt");
        if($conn === false){
            die("ERROR: Could not connect. " . mysqli_connect_error());
        }
        $tno = $_REQUEST['tno'];
        $dname = $_REQUEST['dname'];
        $dphone = $_REQUEST['dphone'];
        $table1 = "INSERT INTO dependent VALUES ($tno, '$dname', '$dphone')";

        if(mysqli_query($conn, $table1)){
            $_SESSION['status']="Data Inserted Successfully";
            header('location: adddependent.php');
        } else{
            echo "ERROR: Hush! Sorry $sql. " . mysqli_error($conn);
        }

        mysqli_close($conn);
        <a href="index.php">
            <button style="color:black; background-color:red;">ADMIN</button>
        </a>
    </center>
</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
    <title>CONNECT PHP</title>
</head>
<body>
    <center>
        <?php
        session_start();
        $conn = mysqli_connect("localhost", "root", "", "apt_mgmt");

        if($conn === false){
            die("ERROR: Could not connect. " . mysqli_connect_error());
        }

        $tno = $_REQUEST['tno'];
        $aptno = $_REQUEST['aptno'];
        $staff_sal = $_REQUEST['staff_sal'];

        $misc_charges = $_REQUEST['misc_charges'];
        $repaircharges = $_REQUEST['repaircharges'];
        $rent = $_REQUEST['rent'];
        $date = $_REQUEST['date'];

        $table1 = "INSERT INTO finance VALUES ('$date', $repaircharges, $staff_sal, $misc_charges, $aptno)";

        $table2 = "INSERT INTO rent VALUES ('$date', $tno, $rent)";

        if(mysqli_query($conn, $table1)){
            $_SESSION['status']="Data Inserted Successfully";
            header('location: pricing.php');
        }

        if(mysqli_query($conn, $table2)){
            $_SESSION['status']="Data Inserted Successfully";
        }
    </center>
</body>
</html>
```

This screenshot shows a Visual Studio Code window with the title "connect2.php - DBMS - Visual Studio Code". The code is written in PHP and interacts with a MySQL database named "DBMS". The code performs two insert operations: one into the "finance" table and another into the "rent" table. It uses session variables to store data and then inserts it into the respective tables. Finally, it closes the database connection and provides links to "index.php" and an "ADMIN" button.

```
$tno = $_REQUEST['tno'];
$aptno = $_REQUEST['aptno'];
$staff_sal = $_REQUEST['staff_sal'];

$misc_charges = $_REQUEST['misc_charges'];
$repaircharges = $_REQUEST['repaircharges'];
$rent = $_REQUEST['rent'];
$date = $_REQUEST['date'];

$table1 = "INSERT INTO finance VALUES ('$date', $repaircharges, $staff_sal, $misc_charges, $aptno)";

$table2 = "INSERT INTO rent VALUES ('$date', $tno, $rent)";

if(mysqli_query($conn, $table1)){
    $_SESSION['status'] = "Data Inserted Successfully";
    header('location: pricing.php');
}

if(mysqli_query($conn, $table2)){
    $_SESSION['status'] = "Data Inserted Successfully";
    header('location: pricing.php');
}

;mysqli_close($conn);
?>
<a href="index.php">
    <button style="color:black; background-color:red;">ADMIN</button>
</a>
</center>
</body>
</html>
```

This screenshot shows a Visual Studio Code window with the title "connect4.php - DBMS - Visual Studio Code". The code connects to a MySQL database using the "apt_mgmt" user. It retrieves the value of the "tno" session variable and performs a query to select all rows from the "tenant" table where the "tno" matches the session variable. The results are then displayed in an HTML table.

```
<!DOCTYPE html>
<html>
<head>
    <title>CONNECT PHP</title>
</head>
<body>
    <center>
        ?>
        session_start();
        $conn = mysqli_connect("localhost", "root", "", "apt_mgmt");

        if($conn === false){
            die("ERROR: Could not connect. " . mysqli_connect_error());
        }

        $tno = $_REQUEST['tno'];
        $result = mysqli_query($conn, "SELECT * FROM tenant WHERE tno=$tno");
        while($row = mysqli_fetch_array($result)){
            echo "<table border='1'>
                <tr>
                    <th>TENANT NUMBER</th>
                    <th>TENANT NAME</th>
                    <th>TENANT PHONE</th>
                </tr>";
            echo "<tr>";
            echo "<td>" . $row['TNO'] . "</td>";
            echo "<td>" . $row['TNAME'] . "</td>";
            echo "<td>" . $row['TPHONE'] . "</td>";
            echo "</tr>";
            echo "<br/>";echo "<br/>";
        }
        echo "</table>";
    </center>
</body>
</html>
```



```

<!DOCTYPE html>
<html>
<head>
<title>CONNECT PHP</title>
</head>
<body>
<center>
<?php
session_start();

$conn = mysqli_connect("localhost", "root", "", "apt_mgmt");

if($conn === false){
die("ERROR: Could not connect.
. mysqli_connect_error());
}

$tname = $_REQUEST['tname'];
$aptno=$_REQUEST['aptno'];
$tmo = $_REQUEST['tmo'];
$tphone = $_REQUEST['tphone'];
$type = $_REQUEST['type'];
$block = $_REQUEST['block'];
$dname = $_REQUEST['dname'];
$dphone = $_REQUEST['dphone'];

$table1="update tenant set TNAME = '$tname' where TNO = '$tmo'";


```

$tbl1="update tenant set TNAME = '$tname' where TNO = '$tmo' ";
$tbl2="update tenant set TPHONE = '$tphone' where TNO = '$tmo' ";
$tbl3="update apartment set TYPE = '$type' where TNO = '$tmo' ";
$tbl4="update apartment set BLOCK = '$block' where TNO = '$tmo' ";

if(mysqli_query($conn, $tbl1)){
$_SESSION['status']="Data Updated Successfully";
header('location: Update.php');
}

if(mysqli_query($conn, $tbl2)){
$_SESSION['status']="Data Updated Successfully";
header('location: Update.php');
}

if(mysqli_query($conn, $tbl3)){
$_SESSION['status']="Data Updated Successfully";
header('location: Update.php');
}

if(mysqli_query($conn, $tbl4)){
$_SESSION['status']="Data Updated Successfully";
header('location: Update.php');
}

if(mysqli_query($conn, $tbl5)){
$_SESSION['status']="Data Updated Successfully";
header('location: Update.php');
}

if(mysqli_query($conn, $tbl6)){
$_SESSION['status']="Data Updated Successfully";
header('location: Update.php');
}

}

;mysqli_close($conn);
?>

```


```

```
<!DOCTYPE html>
<html>
<head>
<title>CONNECT PHP</title>
</head>
<body>
<center>
<?php
session_start();

$conn = mysqli_connect("localhost", "root", "", "apt_mgmt");

if($conn === false){
    die("ERROR: Could not connect. " .
        . mysqli_connect_error());
}

$tno = $_REQUEST['tno'];
$result1 = "DELETE FROM tenant
            where TNO = '$tno'";
$result2 = "DELETE FROM apartment
            where TNO = '$tno'";
$result3 = "DELETE FROM rent
            where TNO = '$tno'";
$result4="DELETE FROM dependent
            where TNO = '$tno'";

if(mysqli_query($conn, $result1)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

if(mysqli_query($conn, $result2)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

if(mysqli_query($conn, $result3)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

if(mysqli_query($conn, $result4)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

mysql_close($conn);?
<a href="Details.php">
    <button style="color:black; background-color:red;">ADMIN</button>
</a>
</center>
</body>
</html>
```

```
$tno = $_REQUEST['tno'];
$result1 = "DELETE FROM tenant
            where TNO = '$tno'";
$result2 = "DELETE FROM apartment
            where TNO = '$tno'";
$result3 = "DELETE FROM rent
            where TNO = '$tno'";
$result4="DELETE FROM dependent
            where TNO = '$tno'";

if(mysqli_query($conn, $result1)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

if(mysqli_query($conn, $result2)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

if(mysqli_query($conn, $result3)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

if(mysqli_query($conn, $result4)){
    $_SESSION['status']="Data Deleted Successfully";
    header('location: Details.php');
}

mysql_close($conn);?
<a href="Details.php">
    <button style="color:black; background-color:red;">ADMIN</button>
</a>
</center>
</body>
</html>
```

CHAPTER 5

QUERIES RELATED SNAPSHOTS

UPDATE FINANCE

The screenshot shows a web browser window with multiple tabs open. The active tab is titled 'Update Tenant'. The page has a light blue background with a form titled 'Update Tenant Details'. The form fields are as follows:

- Name: ABHISHEK KUMAR
- Tenant Number: 4
- Apartment Number: 1006
- Phone: 89867665
- Type: 6BHK
- Block: 6
- Dependent Name: RAMKUWAR SINGH
- Dependent Phone Number: 91999038
- Submit button

Below the form, there is a link labeled 'Contact Us'.

DATA UPDATED SUCCESSFULLY

The screenshot shows a web browser window with multiple tabs open. The active tab is titled 'Update Tenant'. The page has a light blue background with a form titled 'Update Tenant Details'. The form fields are all empty or placeholder text:

- Name: Enter the Tenant Name
- Tenant Number: Enter The Tenant Number
- Apartment Number: Enter The Apartment Number
- Phone: Enter The Phone Number
- Type: Enter The Type
- Block: Enter The Block
- Dependent Name: Enter The Dependent Name
- Dependent Phone Number: Enter The Dependent Phone Number

Screenshot of phpMyAdmin showing the 'tenant' table in the 'apt_mgmt' database.

Table: tenant

Showing rows 0 - 3 (4 total). Query took 0.0004 seconds.

SELECT * FROM `tenant`

+ Options + T -> ▾ TNO TNAME TPHONE

	TNO	TNAME	TPHONE			
<input type="checkbox"/>	Edit	Copy	Delete	1	poorni soni	98765544
<input type="checkbox"/>	Edit	Copy	Delete	2	Ajin Sumesh	2147483647
<input type="checkbox"/>	Edit	Copy	Delete	3	arun kumar n	87676543
<input type="checkbox"/>	Edit	Copy	Delete	4	ABHISHEK KUMAR	89867665

With selected: 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

Screenshot of phpMyAdmin showing the 'apartment' table in the 'apt_mgmt' database.

Table: apartment

Showing rows 0 - 3 (4 total). Query took 0.0004 seconds.

SELECT * FROM `apartment`

+ Options + T -> ▾ APPTNO TNO TYPE BLOCK

	APPTNO	TNO	TYPE	BLOCK			
<input type="checkbox"/>	Edit	Copy	Delete	4	3	3bhk	2
<input type="checkbox"/>	Edit	Copy	Delete	1001	1	9 bhk	9
<input type="checkbox"/>	Edit	Copy	Delete	1002	2	2BHK	1
<input type="checkbox"/>	Edit	Copy	Delete	1006	4	6BHK	6

With selected: 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

The screenshot shows the phpMyAdmin interface connected to the 'apt_mgmt' database. The 'dependent' table is selected, displaying the following data:

TNO	DNAME	DPHONE
1	harsh soni	2147483647
2	M SUMESH	2147483647
4	RAMKUWAR SINGH	91999038

DELETE TENANT

The screenshot shows a web page titled "Delete tenant". A form is displayed with the heading "Enter the Tenant Number" and a text input field containing the value "4". Below the form, there is a section titled "Contact Us" and a map of a local area with several landmarks labeled.

Delete tenant

Enter the Tenant Number

4

Submit

Contact Us

Creativeabhi13
26.305439,84.187254 ABHISHEK KUMAR S/O RAMKUWAR SINGH & BINOD KUMARI VILL-BHAIGWANPUR, POST: Sharur, Bihar 841243
4.6 ⭐⭐⭐⭐⭐ 19 reviews
View larger map

Directions

Madarsa

Khap Banakat

Durga Temple

Hanuman Temple

Kali mata temple

Chafwa Mathia

Khalwa Asaram

M/S Gangotri Group

Ankit Digital Service Center

Khalwa Village

Durga temple

Shrikant Singh

Smart Coaching Centre

Mishrauli

Let us know

Name
Enter Your Name

Email
Enter Your Email

Subject
Enter The Subject

DATA DELETED SUCCESSFULLY

The screenshot shows a web browser with three tabs open:

- New material "project report fo...": This tab is mostly blank.
- *Module 4-converted.pdf: This tab shows a PDF document.
- localhost / 127.0.0.1 / apt_mgmt: This tab displays the main interface of an Apartment Management System.

The main interface has a header with the Abhishek & Aditya Group logo and navigation links: ADDTENANT, ADDDEPENDENT, FINANCE, Update Details, Details, Contact Us, and Apartment.

The first section, "Tenant Details", contains a form with a placeholder "Enter The Tenant Number" and a "Submit" button.

The second section, "Delete tenant", also contains a similar form with the same placeholder and button.

The third section, "Contact Us", contains a form with the same placeholder and button.

The screenshot shows the phpMyAdmin interface connected to a MySQL database named "apt_mgmt". The left sidebar shows the database structure with tables like New, admin, apartment, contact_us, dependent, finance, rent, and tenant.

The right panel displays the "tenant" table with the following data:

TNO	TNAME	TPHONE
1	poorni soni	898765544
2	Ajin Sumesh	2147483647
3	arun kumar n	87676543

CONCLUSION

The use of this software to manage all the aspects of an apartment makes it very easy for owners and property managers to maintain records of their apartments and to quickly look up important details. It saves a significant amount of time when compared to the traditional way of pen and paper, where records tend to get lost or are hard to update and search. Data entry is much faster and less tedious. There can also be multiple copies of the database ensuring that if the main record database runs into issues and is not available, the data is safe elsewhere and the records are not lost.

REFERENCES

PHP MYADMIN

<https://www.phpmyadmin.net/docs/>

MYSQL

<https://dev.mysql.com/doc/>