03
CHAPTER

39V432

Introduction to Database Management System

LEARNING OBJECTIVES

To understand database concepts, components and its functions.

- **To know about relational model of data**
- **To understand Query languages for databases.**
- **Enables to write SQL commands and query processing**
- **To enhance the programming skills and Techniques using MySQL**

3.1 Introduction to Database Management System

DBMS stands for Database Management System, so let us break down the words what they really mean. A database is a place where we store, retrieve and manage data. So what's a data then? Meaningful information like your name, your favorite color etc to complex data like astronomical data that scientist handle, everything comes under database. The management system refers to a set of programs to manage the data, we have with various actions like storing, retrieving, filtering etc. Some of the popular Database Management System is MySQL, Oracle etc. Giving protection to data, user-friendly for users etc, are some of the notable features of good DBMS.

3.1.1 Introduction to DBMS

Definition: "A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data."

What type of data is stored in a database?

In a database, we would be grouping only related data together and storing them under one group name called table. This helps in identifying which data is stored where and under what name.

3.1.2 Evolution of DBMS

The concept of storing the data started before 40 years in various formats. In earlier days they have used punched card technology to store the data. Then files were used. The file systems were known as predecessor of database system. Various access methods in file system were indexed, random and sequential access. The file system had more limitations like

Data Duplication – Same data is used by multiple resources for processing, thus created multiple copies of same data wasting the spaces.

High Maintenance – Access control and verifying data consistency needs high maintenance cost.

Security – less security provided to the data.

So database systems became popular to overcome the above limitations of file system.

3.1.3 DBMS Concepts

There exist few standards that are applicable to all forms of database management Systems like Relational Database Management System (RDBMS) and Object Database Management System (ODBMS). All DBMS adheres to the following two basic concepts.

ACID Properties – The acronym stands for Atomicity, Consistency, Isolation and Durability. **A**tomicity follows the thumb rule "All or Nothing", while updating the data in database for the user performing the update operation. This update operation is called as transaction and it either commits (successful updating) or aborts (updating failure). **C**onsistency ensures that the changes in data value to be constant at any given instance. This property helps in the successful transaction. **I**solation property is needed during concurrent transaction. When multiple users do the transactions by accessing same object at the same time, the transaction is known as concurrent transaction. To prevent the conflict in database update, the transactions are isolated from other user and serialized. This is also known as Degree of Consistency. **D**urability is defined as the system's ability to recover all committed transactions during the failure of storage or the system.

Concurrency Control and Locking – It is the DBMSs mechanism used for data sharing. When the same data is shared among multiple users, proper access control is needed and privilege of changing the applications data item is controlled through Locking.

3.2 DBMS Database Models

The database technology came into existence in terms of models with relational and object-relational behavior. The major database models are listed below:

3.2.1 Hierarchical Database Model

The famous Hierarchical database model was IMS (Information Management

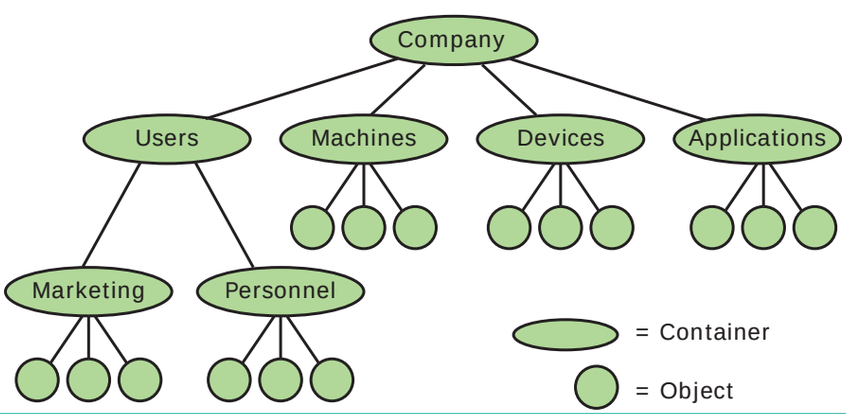


Figure: 3.1 Hierarchical database model

System), IBM's first DBMS. In this model each record has information in parent/ child relationship like a tree structure. The collection of records is called as record types, which are equivalent to tables in relational model. The individual records are equal to rows. See Figure 3.1

In the above model we have many advantages like less redundant data, efficient search, data integrity and security. This model also has few limitations like complex to implement and difficulty in handling many to many relationships.

3.2.2 Network model

The first developed network data model was IDS (Integrated Data Store) at Honeywell. Network model is similar to Hierarchical model except that in this model each member can have more than one owner. The many to many relationships are handled in a better way. This model identified the three database components Network schema, Sub schema and Language for data management. See Figure 3.2

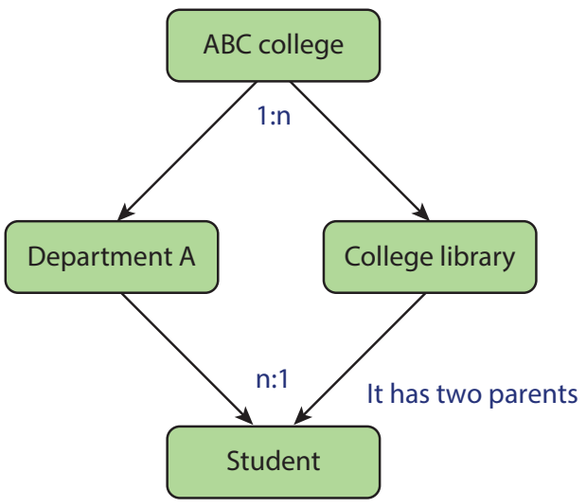


Figure: 3.2 Network Model

Network schema – schema defines all about the structure of the database.

Sub schema – control on views of the database for the user

Language – basic procede for accessing the database.

The major advantage of this model is the ability to handle more relationship types, easy data access, data integrity and independence. The limitation of network model is difficulty in design and maintenance.

3.2.3 Relational model

Oracle and DB2 are few commercial relational models in use. Relational model is defined with two terminologies Instance and Schema. See Figure 3.3

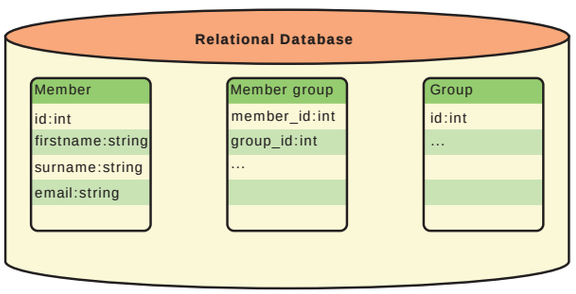


Figure: 3.3 Relational database model

Instance – A table consisting of rows and columns

Schema – Specifies the structure including name and type of each column.

A relation (table) consists of unique attributes (columns) and tuples (rows).

3.2.4 Object-oriented database model

This model incorporates the combination of Object Oriented Programming(OOP's) concepts and database technologies. Practically, this model serves as the base of Relational model. Object oriented model

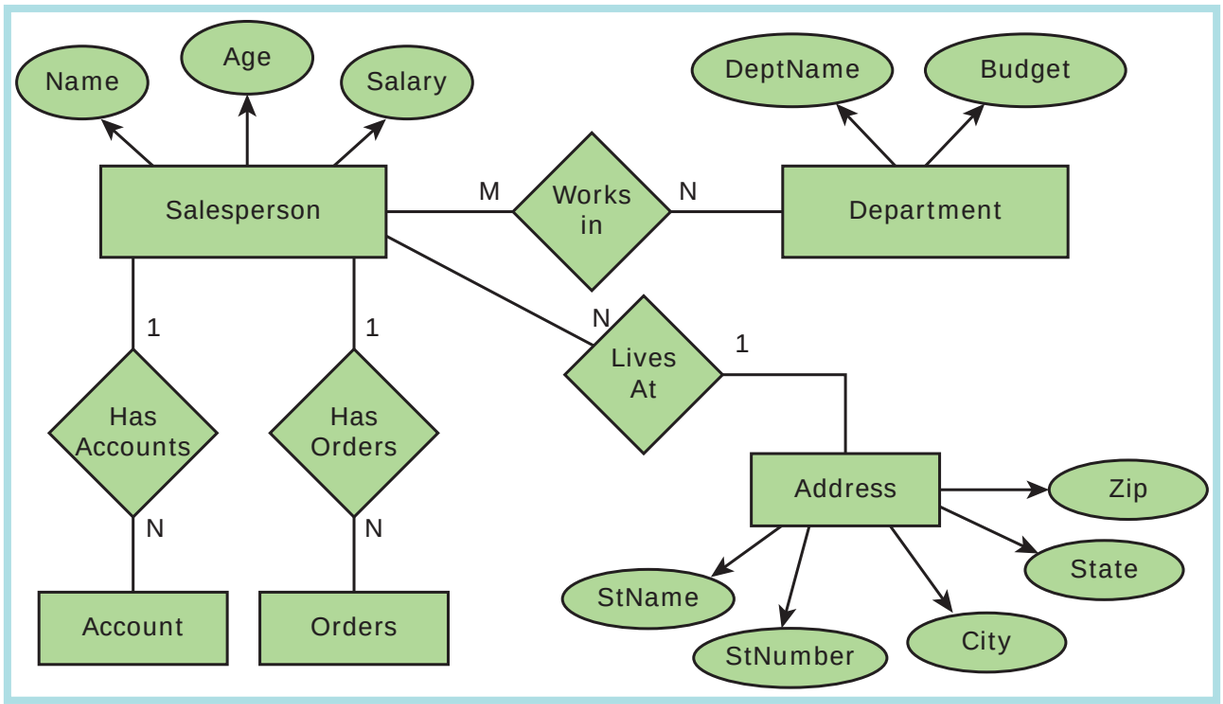


Figure: 3.4 **Object-oriented database model**

uses small, reusable software known as Objects. These are stored in object oriented database. This model efficiently manages large number of different data types. Moreover complex behaviors are handled efficiently using OOP's concepts. See Figure 3.4

3.3 Relational Database Management System

Basic RDBMS concepts

Any database whose logical organization is based on relational data model is known as Relational Database. A DBMS that manages the relational database is known as Relational Data Base Management System. RDBMS is basis for SQL and for all modern database systems like MySQL, oracle and Microsoft Access. The basic RDBMS concept includes Database, Tables, Tuple, Attribute, Schema and Key which are discussed in RDBMS Jargons.

3.4 RDBMS Jargons

3.4.1 Database

The most popular Relational Database is MySQL. It is an open source SQL database supporting different platforms like Windows, Linux and MAC Operating Systems. The other relational databases available are Oracle, MS SQL Server and MS Access. The features of RDBMS are

- High Availability
- High Performance
- Robust Transactions and support
- Ease of management
- Less cost

3.4.2 Table

In relational database model, table is defined as the collection of data organized in terms of rows and columns. Table is the simple representation of relations. The true relations cannot have duplicate rows

where as the table can have. The example of Employee table is shown below in Table 3.1.

Table 3.1 Table Structure

ID	NAME	AGE	SALARY
1	Alex	26	22,000
2	Divya	25	20,000
3	Tulsi	28	30,000

3.4.3 Column

The table consists of several rows and columns. Table can be divided into smaller parts, in terms of columns. Each column is known as attributes. In the Employee table four attributes are available namely Id, Name, Age and Salary. The attribute is defined in a table to hold values of same type. This is known as Attribute Domain. In the Employee table, the Name field will hold only characters not the numbers in it.The vertical entity in a table is known as Attribute or Column.

3.4.4 Row

A single entry in a table is called as Row or Record or Tuple. Set of related data's are represented in a row or tuple. The horizontal entity in a table is known as Record or row. See Table 3.2

Table 3.2 Row Structure

ID	NAME	AGE	SALARY
1	Alex	26	22,000

3.4.5 Key

The candidate key that is chosen to perform the identification task is called the primary key and any others are Alternate keys. Every tuple must have, by definition, a unique value for its primary key. A primary key which is a combination of more than one attribute is called a composite primary key.

3.4.6 Foreign Key

A foreign key is a "copy" of a primary key that has been exported from one relation into another to represent the existence of a relationship between them. A foreign key is a copy of the whole of its parent primary key i.e if the primary key is composite, then so is the foreign key. Foreign key values do not (usually) have to be unique. Foreign keys can also be null. A composite foreign key cannot have some attribute(s) null and others non-null.

3.4.7 Super Key

An attribute or group of attributes, which is sufficient to distinguish every tuple in the relation from every other one is known as Super Key. Each super key is called a candidate key. A candidate key is selected from the set of Super Key. While selecting candidate key, redundant attributes should not be taken. The candidate key is also known as minimal super keys.

3.4.8 Composite Key

A key with more than one attribute to identify rows uniquely in a table is called Composite key. This is also known as Compound Key.

3.5 ER Model

Generally we use an ER model to know the concept of database design and this model consists of a collection of entities(real world objects)where each of these entities will be interconnected with each other with conditions and dependencies(i.e. one entity is dependent on another).

3.5.1 ER Modeling basic concepts

The basic concepts of ER model consists of

1. Entity or Entity type
2. Attributes
3. Relationship

These are the general concepts which help to create an ER diagram and produce an ER model. With the help of these any database design can be created and viewed to know the concept in that database design.

3.5.2 Entity or Entity type

An Entity can be anything a real-world object or animation which is easily identifiable by anyone even by a common man.

Eg: In a company's database Employee, HR, Manager are considered as entities, where each of these entity will be having their own attributes. An entity is represented by a rectangular box. See Figure 3.5.

Figure 3.5 Entity

Types of Entity:

1. Strong Entity
2. Weak Entity
3. Entity Instance

Strong Entity

A Strong entity is the one which doesn't depend on any other entity on the schema or database and a strong entity will have a primary key with it (i.e. a unique id which other entities will not have in their attributes).It is represented by one rectangle.In the above example it is a strong entity because it has a primary key(a unique id) as the roll no because for every one roll no varies and it will not be same.

Weak Entity

A weak entity is dependent on other entities and it doesn't have any primary key like the Strong entity. It is represented by double rectangle.

For Example:

Here the marks is the weak entity and there are no unique id or primary key for that entity. So they are dependent on the existence of the other entity. See Table 3.3

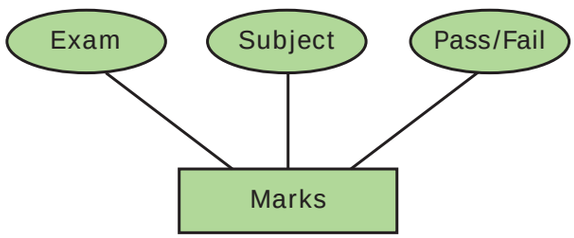


Figure: 3.6 Weak Entity

3.5.3 Entity Instance

Instances are the values for the entity if we consider animals as the entity their instances will be dog, cat, cow… Etc. So an Entity Instance denotes the category values for the given entity.

Table: 3.3 Entity Instances

Entity	Instances
Human	Male , Female
Animals	Dog, cats, lion
Jobs	Engineer, Doctor, Lawyer
Actors	Ajith, Vijay, Vikram
Electronics	Laptop, Mobile

3.5.4 Attributes

An attribute is the information about that entity and it will describe, quantify, qualify, classify, and specify an entity. An attribute will always have a single value, that value can be a number or character or string.

Types of attributes:

1. Key Attribute
2. Simple Attributes
3. Composite Attributes
4. Single Valued Attribute
5. Multi Valued Attribute

3.5.4.1 Key Attribute

Generally a key attribute describes a unique characteristic of an entity.

3.5.4.2 Simple Attribute

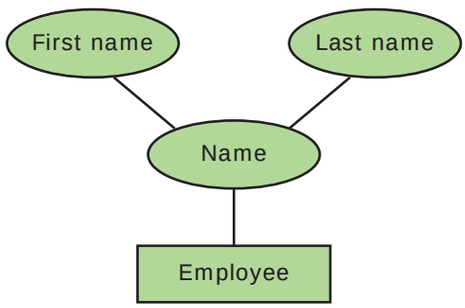


Figure 3.7 Simple Attribute

The simple attributes cannot be separated. It will have a single value for that entity. For Example: Let us consider the name as the attribute for the entity employee and here the value for that attribute is a single value. See Figure 3.7

3.5.4.3 Composite Attributes

The composite attributes can be sub-divided into simple attributes without change in the meaning of that attribute. For Example: In the above diagram the employee is the entity with the composite attribute Name which are sub-divided into two simple attributes first and last name.

3.5.4.4 Single Valued Attributes:

A single valued attribute contains only one value for the attribute and they don't have multiple number of values. For Example:Age- It is a single value for a person as we cannot give 'n' number of ages for a single person,therefore it is a single valued attribute. See Table 3.4

Table: 3.4 Single Valued Attributes

Attribute	Values
Age	3
Roll no	85

In the above table are the some examples for single valued attributes. See Figure 3.8

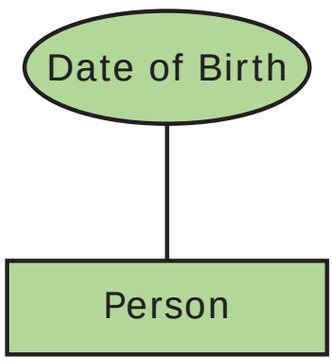


Figure 3.8 Single Valued Attributes

3.5.4.5 Multi Valued Attributes:

A multi valued attribute has more than one value for that particular attribute.For Example:Degree - A person can hold n number of degrees so it is a multi-valued attribute.

In Table 3.5 are some examples for Multi valued attributes.

Table: 3.5 Attributes and Values

Attribute	Values
Degree	B.Tech, MBA
Bank_Account	SBI, HDFC

3.5.5 Relationship Type

In ER Model, relationship exists between two entities. Three types of relationships are available and the Entity-Relationship(ER) diagram is based on the three types listed below.

One-to-One relationship: Consider two entities A and B. one-to-one (1:1) relationship is said to exist in a relational database design, if 0 or 1 instance of entity A is associated with 0 or 1 instance of entity B, and 0 or 1 instance of entity B is associated with 0 or 1 instance of entity A.

One-to-Many relationship: Consider two entities A and B. one-to-many (1:N) relationship is said to exist in a relational database design, for 1 instance of entity A there exists 0 or 1 or many instances of entity B, but for 1 instance of entity B there exists 0 or 1 instance of entity A.

Many-to-Many relationship:Consider two entities A and B. many-to-many (M:N) relationship is said to exist in a relational database design, for 1 instance of entity A there exists 0 or 1 or many instances of entity B, and for 1 instance of entity B there exists 0 or 1 or many instance of entity A.

In reality one-to-one are in less usage, where as one-to-many and many-to-many are commonly used. However in relational databases, many-to-many are converted into one-to-many relationships.

3.5.5.1 Relationship instance

It is a finite set of liples in the RDBMS systems relation instances never have duplicate . E.g if **Works-for** is the relationship between the Employee entity and the department entity, then Ram works for Comp.sc department, shyam works for electrical department ..etc are relationship instances of the relationship, works for.

3.5.5.2 Degree of a relationship

The number of entity types involved is known as Degree of relationship. One – Unary, Two – Binary, Three – Ternary.E.g An employee of an organization acts as manager of few other employees. It also connects one entity to itself as a loop. so **manager-of** is unary. Similarly employee **works-for**department, connects two entities and is binary. If a customer purchase an item, it involves shop keeper also and is a ternary relationship.

3.5.5.3 Cardinality

It is defined as the number of items that must be included in a relationship.ie number of entities in one set mapped with the number of entities of another set via the relationship. The three classifications in Cardinality are one-to-one, one-to-many and Many-to-Many. See Figure 3.9-3.11

Figure 3.9 Cardinality

In the above example we have two entities Person and Vehicle. If we consider a person driving vwchicle, then we have one-to-one relationship between Person and Vehicle. See Figure 3.10

Figure 3.10 CardinalityRelation 1 to n

In the above example, Customer places the Order is a one-to-many relationship.

Here the customer can place multiple orders and the order is related to only one customer. See Figure 3.11

Figure: 3.11 Cardinality Relation n to n

The example of many-to-many relationship is Students registering the Courses. A student can register more than one courses and A course can be registered by many students. Hence it is many-to-many.

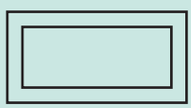
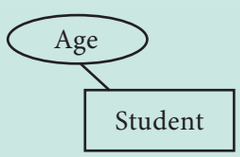
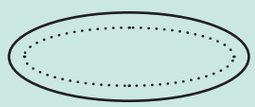
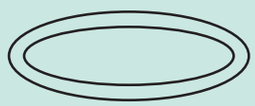
3.6 ER-Diagram

ER Diagram presents data visually on how they are related to each other. This model follows separate notations for representing data into entities, attributes and relationship between the entities.

3.6.1 ER-Modeling Diagram Notations

Entities, Attributes and Relationship forms the components of ER Diagram and the defined symbols and shapes are summarized below in Table 3.6.

Table 3.6 ER diagram Notations

ER Component	Description (how it is represented)	Notation
Entity - Strong	Simple rectangular box	
Entity – Weak	Double rectangular boxes	
Relationships	Rhombus symbol - Strong	
between Entities	Rhombus within rhombus – Weak	
Attributes	Ellipse Symbol connected to the entity	
Key Attribute for Entity	Underline the attribute name inside Ellipse	
Derived Attribute for Entity	Dotted ellipse inside main ellipse	
Multivalued Attribute for Entity	Double Ellipse	

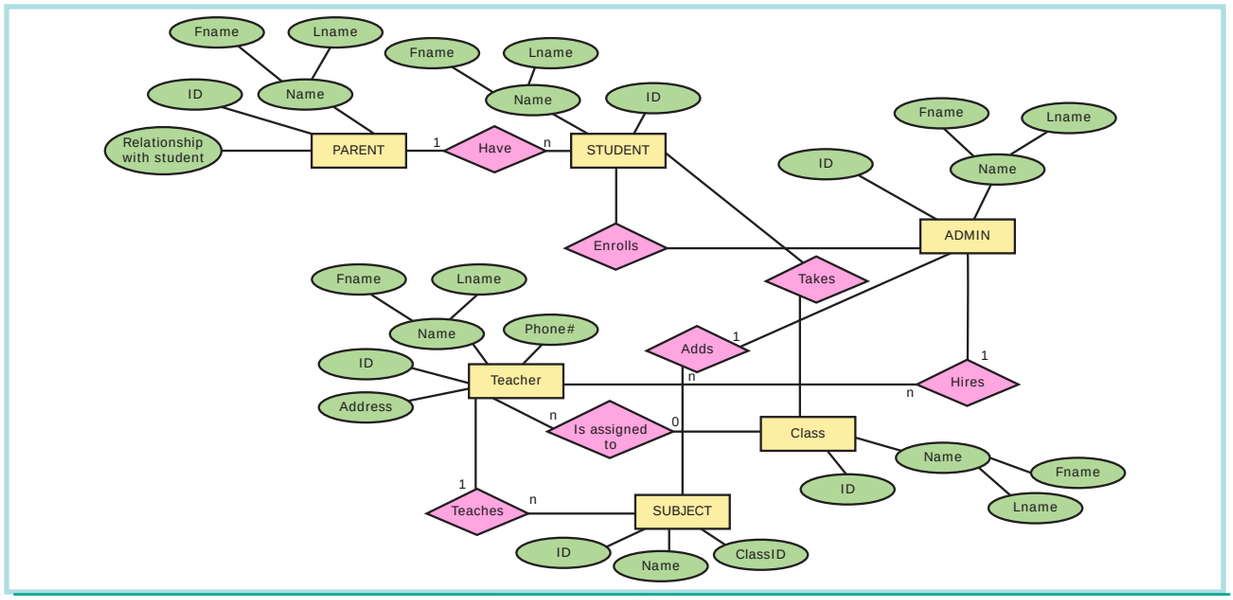


Figure: 3.12 ER-Diagram

3.6.2 Example

Fig 3.12 is an example of ER Diagram for the scenario of School Management System. We have many entities like Parent, student, Admin, Teacher, class and subjects. The attributes of few entity is given below:

Parent – Name, Id, Fname, Lname.
Student – Id, Name, Fname, Lname.
Teacher – Name, Id, Phone#, Address.

The relationships that exist between the entities are as follows:

- Parent **HAVE** Student
- Admin **ENROLLS** Student
- Admin **HIRES** Teacher
- Admin **ADDS** Subject
- Teacher **TEACHES** Subject
- Teacher **IS ASSIGNED TO** Class

Key Attributes are listed with underline.

3.7 Introduction to MYSQL

3.7.1 About MYSQL

MySQL is an open source relational database management system. Its name is a combination of "My" the name of the founder Monty Widenius's daughter and "SQL". A clear definition of database and SQL is mandatory to understand MySQL. In simple, a database is defined as the structured collection of data. Ex. Photo gallery is a database which has collection of photos (data). SQL - structured query language is not a database. It is a standardized language used to access the database and the data's are processed to turn into efficient information. The SQL standard always refers to the current version and the current version is 2003. MySQL is open source software that allows managing relational databases. It also provides the flexibility of changing the source code as per the needs. It runs on multiple platforms like Windows, Linux and is scalable, reliable and fast.

3.7.2 Overview of Web Database

Many databases are in existence to meet out the needs of the application. These databases are broadly divided into Heavy and Light databases. Heavy databases support all the desktop applications whereas the web applications are

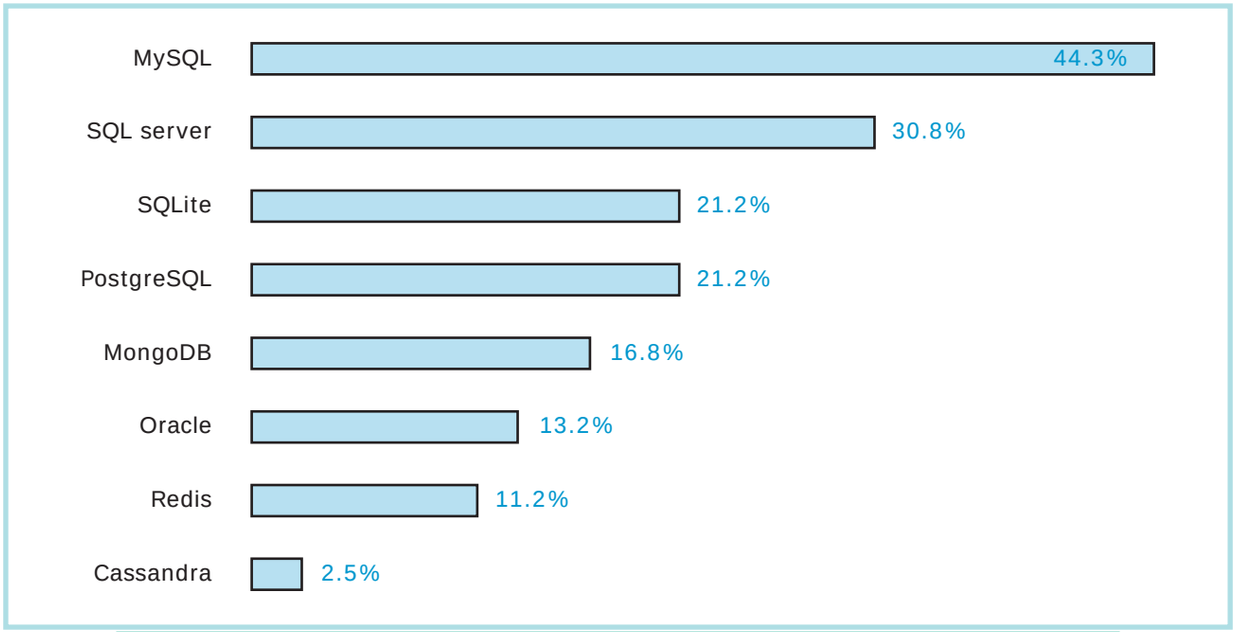


Figure: 3.13 Most Popular databases - Statistics

supported by Light databases. Below are the lists of commonly used databases.

- DB2
- MySQL
- Oracle
- PostgreSQL
- SQLite
- SQL Server
- Sybase

Due to the rapid growth in the web domain, most of the desktop applications are being converted into web applications. These transformations to the web results in the availability of many web applications in the network. Heavy databases did not meet out the network issues efficiently. Light databases were able to handle all the issues raised by the network. So all the light databases that supports the web applications are also known as Web Databases. See Figure 3.13

MySQL is the most commonly used database in the world due to its ease of use.

3.8 MySQL - Administration

3.8.1 MySQL – Administration Responsibilities

In general there exists a role known as Database Administrators (DBA's) who takes care of configuration, installation, performance, security and data backup. DBA's posses the skills on database design, database queries, RDMS, SQL and networking. The primary task is the creation of new user and providing them with access rights.

Creating New User Account to MySQL

In **MySQL** database, there exists a table named **user**. The newly created account must have an entry in this **user** table. Consider the admin creates an account with username and password. The user account is activated with various access rights like INSERT, SELECT and UPDATE. Consider the **user** table has the following fields host, name, password, select_priv, insert_priv and update_priv.

A new user account is added with values to the **user** table using the following INSERT query in MySQL database. The Syntax for inserting record is INSERT INTO table name (Parameter1,Parameter2, Parameter3..) VALUES (Value1, Value2, Value3..); All the query is in SQL will terminate with semicolon(;).

```
mysql> INSERT INTO user (host, name, password, select_priv, insert_priv, update_priv)
VALUES ('localhost', 'guest', PASSWORD('guest123'), 'Y', 'Y', 'Y');
```

Query OK, 1 row affected (0.20 sec) – This statement implies that the query is executed successfully with the time in seconds.

```
mysql>FLUSH PRIVILEGES;
```

The above command is executed after every new account creation. This command is similar to rebooting the server so that newly created account and the access privilege are updated in the server. Manual server rebooting is avoided by this command.The inserted record is retrieved using SELECT query and the structure is shown below Table 3.7 & 3.8:

```
mysql>SELECT * FROM user WHERE name = 'guest';
```

Table 3.7 Example Table

host	name	password	select_ priv	insert_ priv	update_ priv
localhost	guest	j2gd6vxd1bj3k13g4	Y	Y	Y

Since MySQL is more secured, it provides function to encrypt the password. Thus the password 'guest123' is encrypted and stored as 'j2gd6vxd1bj3k13g4' using PASSWORD() function. The parameters select_priv, insert_priv and update_priv are few privileges set for the new user. If the flag is set as 'Y' then access is granted and if flag set as 'N' then access is denied.

Table 3.8 List of privileges available in MySQL

Privileges	Action Performed (If Granted)
Select_priv	User can select rows from database tables.
Insert_priv	User can insert rows into database tables.
Update_priv	User can update rows of database tables.
Delete_priv	User can delete rows of database tables.
Create_priv	User can create new tables in database.
Alter_priv	User can make changes to the database structure.

3.8.2 Administrative MySQL Command

The Database Administrator (DBA) frequently uses few commands to control the entire database. These commands are known as Administrative MySQL Commands. The following are few such important commands used frequently while working with MySQL.

1. **USE Database** – This command is used to select the database in MySQL for working. If there exists a database named **test**, it is used as working database using the below Syntax.

   Syntax:

```
mysql > use test;
   Database changed
mysql>
```

2. **SHOW Databases** – Lists all the databases available in the database server. See Table 3.9

Syntax:

```
mysql > show databases;
```

Table 3.9 Database List

Database
test
mysql
employee
students
parents

3. **SHOW Tables** – Lists all the tables available in the current database we are working in. See Table 3.10

Syntax:

```
mysql > show tables;
```

Table 3.10 Table List

Tables_in_test
school
salary
employee
library
sports

4. **SHOW COLUMNS FROM tablename** – Lists all the attributes, attribute type, Is Null value permitted, key information, default value and other information for the given table. The show columns for **sports** table is given below in Table 3.11.

Syntax:

```
mysql > show columns from sports;
```

Table 3.11 Column List

Field	Type	Null	Key	Default	Extra
Team_id	int(5)	yes		null	
Team_name	varchar (10)	yes		null	
Team_size	int(5)	yes		null	
Team_rank	int(5)	yes		null	

5. **SHOW INDEX FROM tablename** – The query shows all the indexes for the given table.

Syntax:

```
mysql > show indexes from sports;
```

6. **SHOW TABLE STATUS LIKE tablename\G** – This command provides with detailed report on the performance of the table.

3.8.3 MySQL Installation

Download and install XAMPP Server Software from Internet. Refer Figure 3.14 to 3.23.

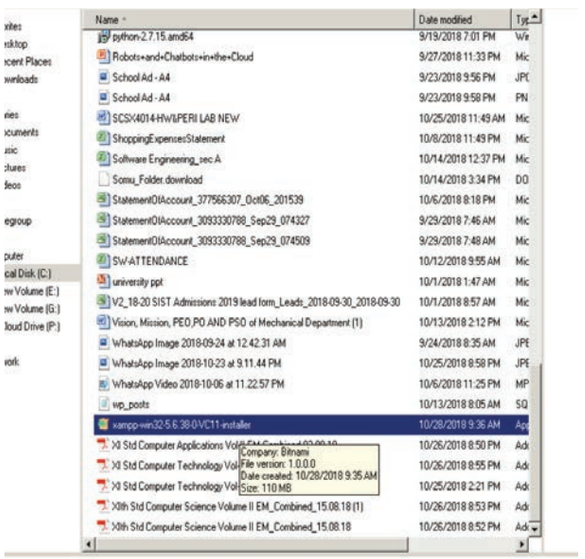


Figure: 3.14 XAMPP Server executable file

Click the Welcome Page Next Button

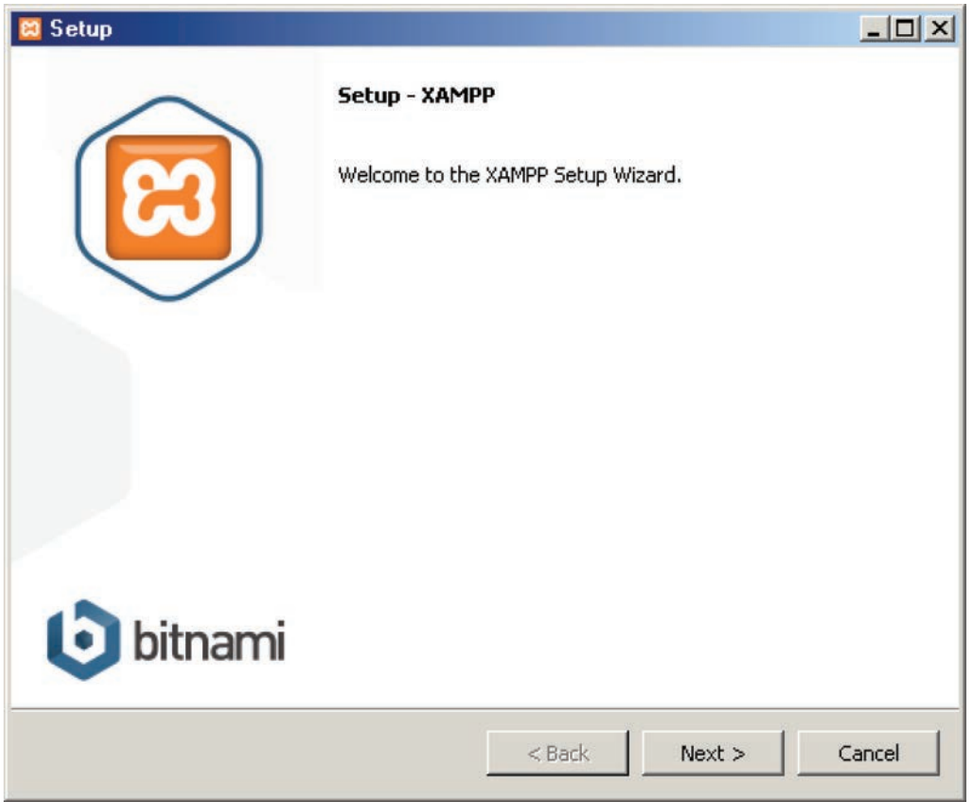


Figure: 3.15 XAMPP Server Welcome Wizard

Select the Required component along with MYSQL component and click next button

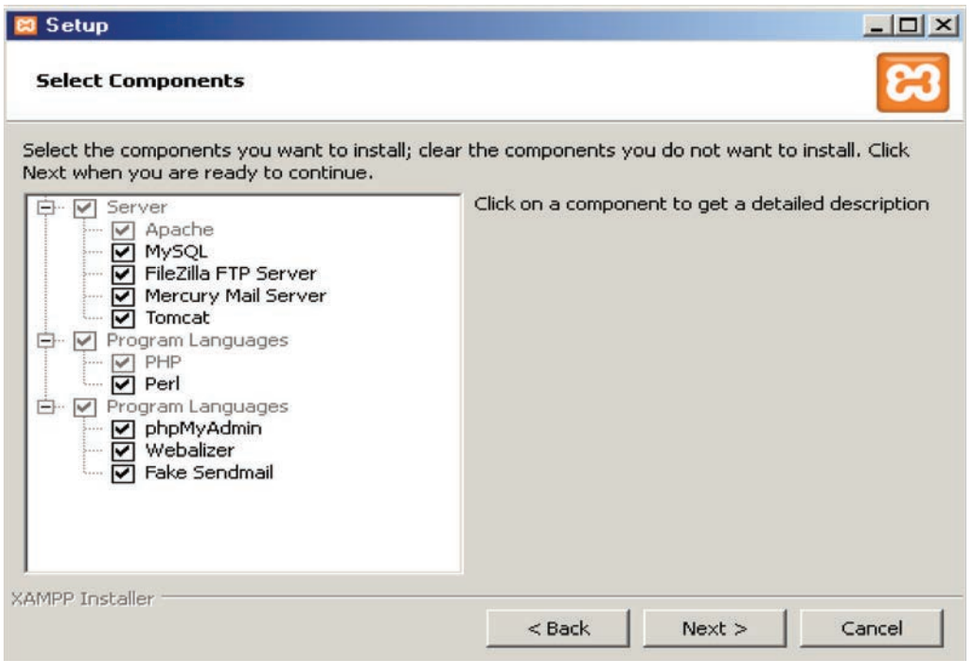


Figure: 3.16 XAMPP Server Component Wizard

Choose The Installation Folder and click Next

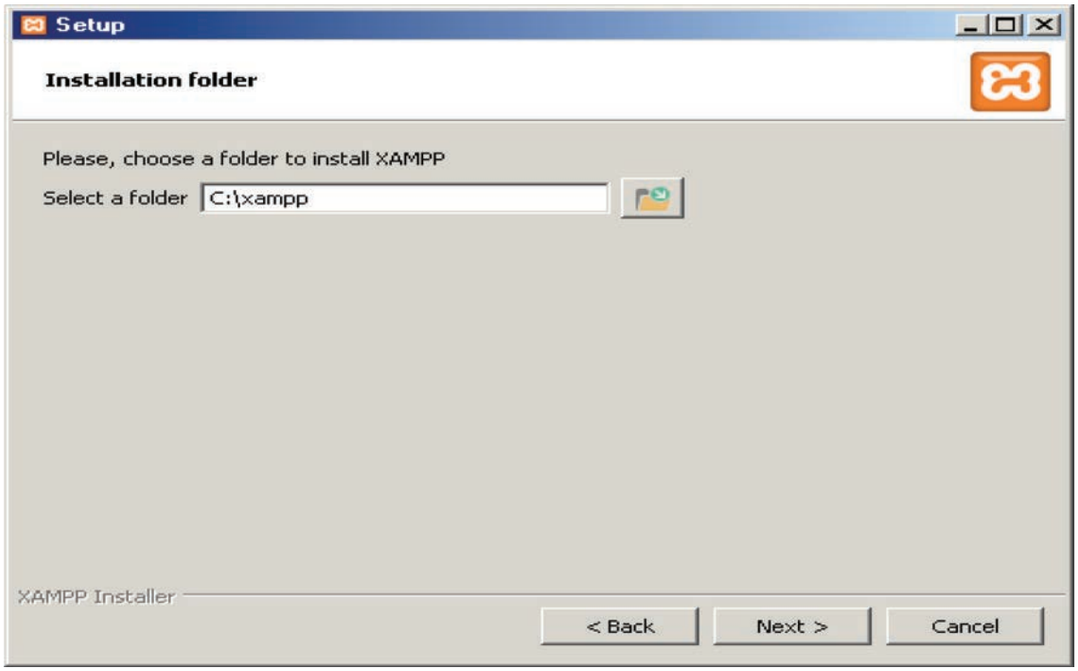


Figure: 3.17 XAMPP Server install path

Click Next Button in Setup ready page

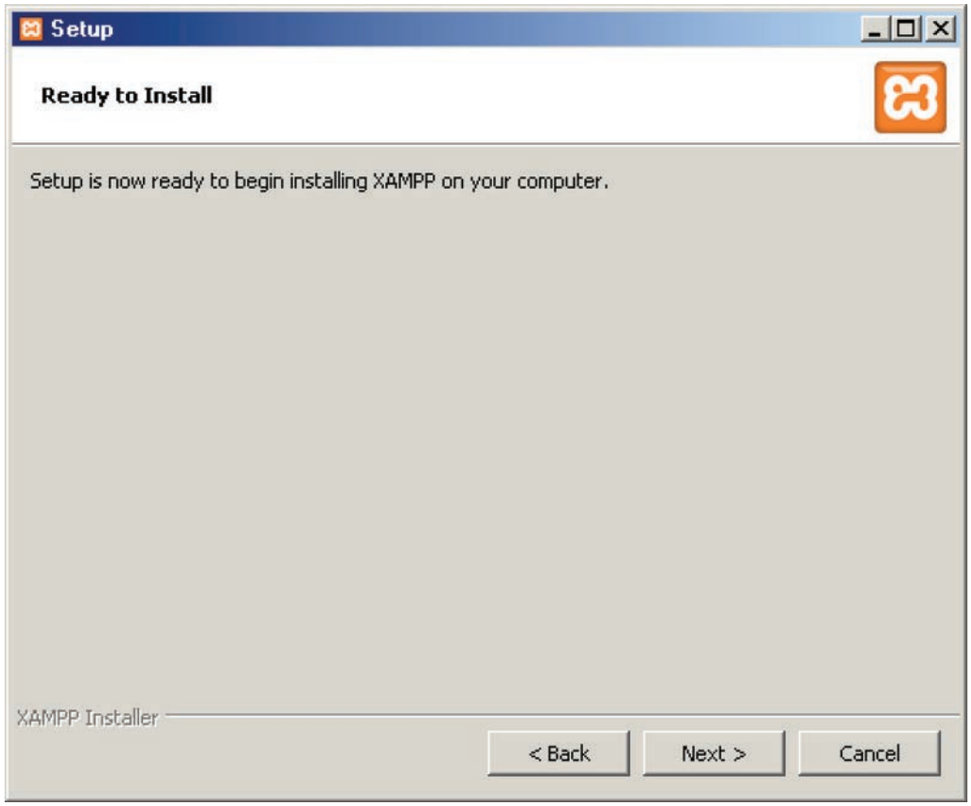


Figure: 3.18 XAMPP Server setup completion

Installation get started

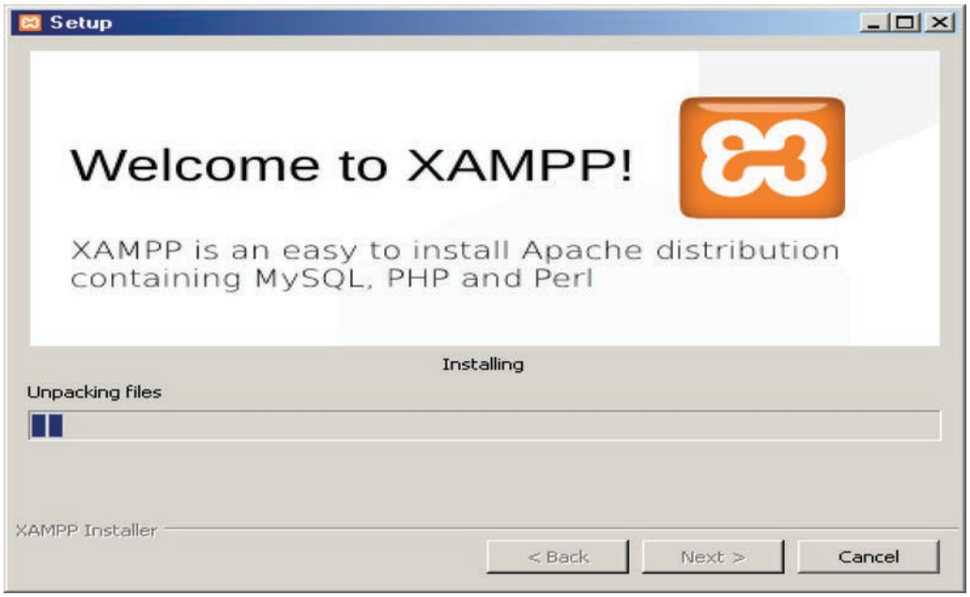


Figure: 3.19 XAMPP Server setup Progress window

After installing Click finish button and open the XAMMP Control panel

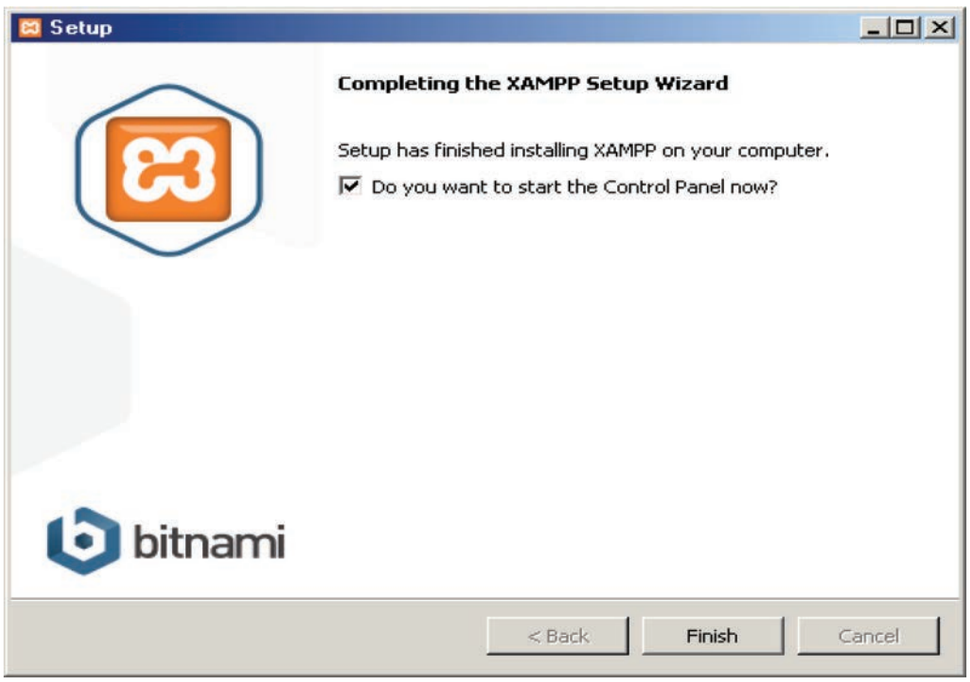


Figure: 3.20 XAMPP Server setup completion

In the Control Panel start the Apache and MySQL Services one by one

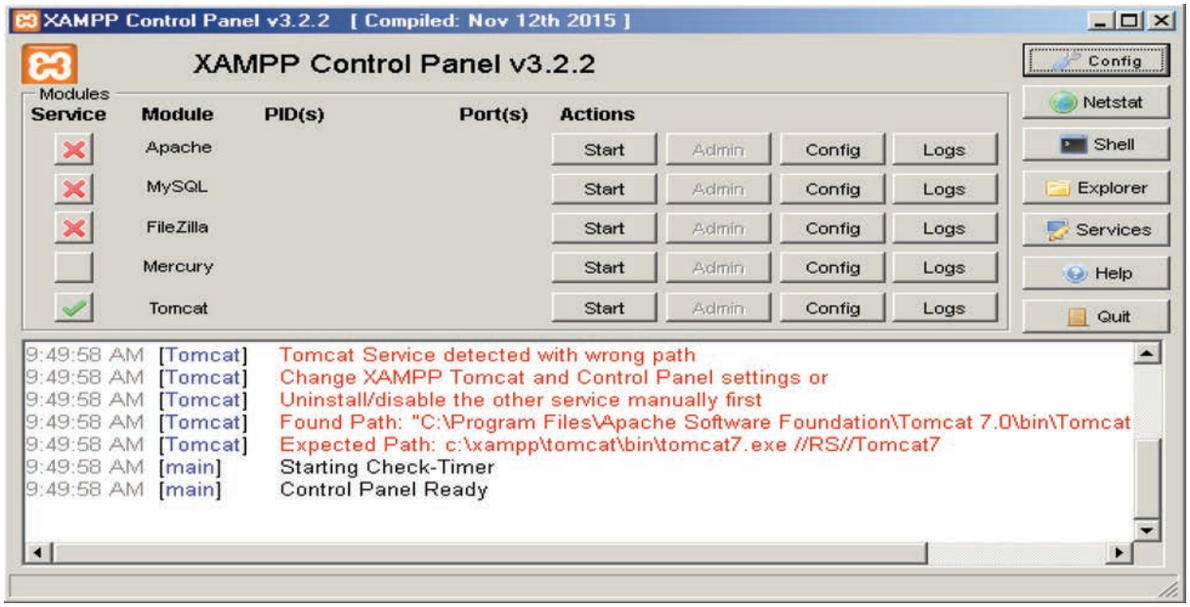


Figure: 3.21 XAMPP Server Control panel

The two services get started one by one

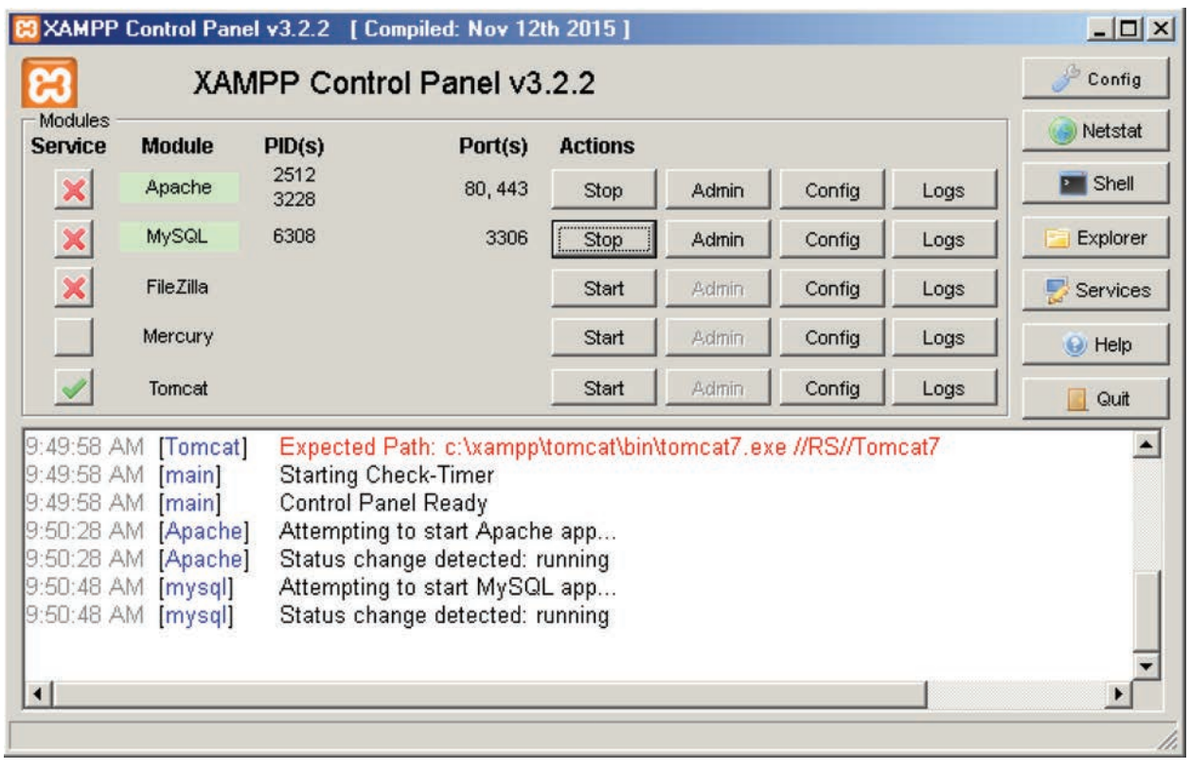


Figure: 3.22 XAMPP Server Services start option

Open the URL http://localhost/phpmyadmin URL in a browser to access MySQL database.

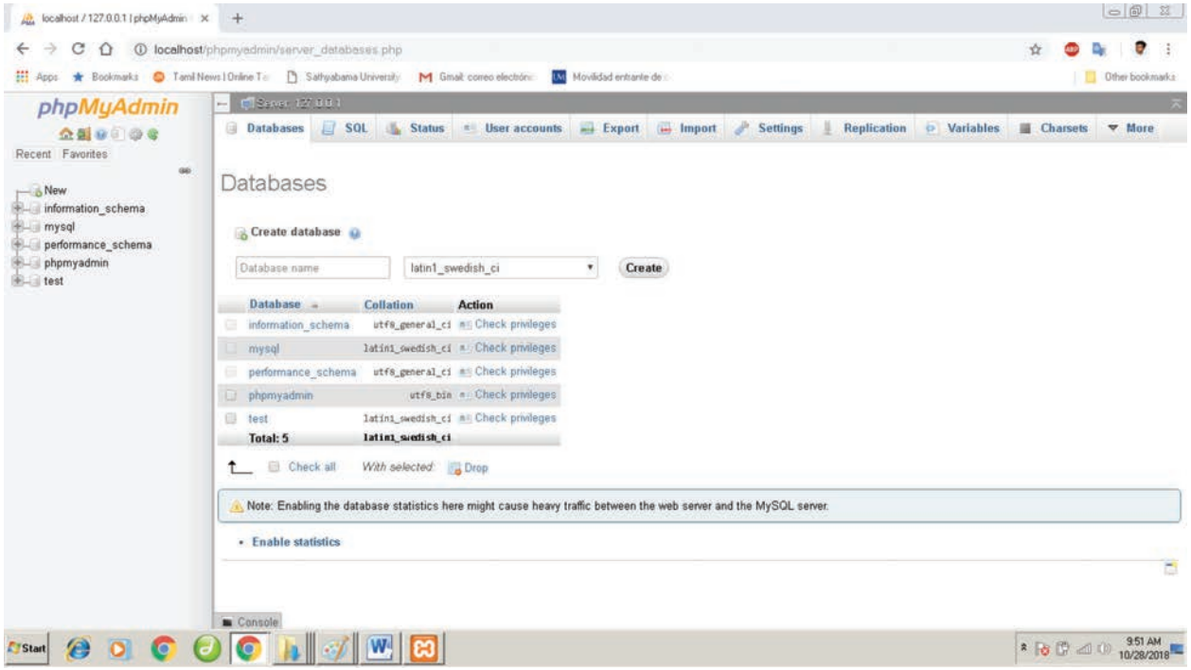


Figure: 3.23 PHP Myadmin Mysql Database server User interface

3.9 MYSQL Administration open source software tools

Types of software tools

Many open source tools are available in the market to design the database in a better and efficient manner. PhpMyAdmin is most popular for Web Administration. The popular Desktop Application tools are MySQL Workbench and HeidiSQL.

PHPMYADMIN (Web Admin)

This administrative tool of MySQL is a web application written in PHP. They are used predominantly in web hosting. The main feature is providing web interface, importing data from CSV and exporting data to various formats. It generates live charts for monitoring MySQL server activities like connections, processes and memory usage. It also helps in making the complex queries easier.

MySQL Workbench (Desktop Application)

It is a database tool used by developers and DBA's mainly for visualization. This tool helps in data modeling, development of SQL, server configuration and backup for MySQL in a better way. Its basic release version is 5.0 and is now in 8.0 supporting all Operating Systems. The SQL editor of this tool is very flexible and comfortable in dealing multiple results set.

HeidiSQL (Desktop Application)

This open source tool helps in the administration of better database systems. It supports GUI (Graphical User Interface) features for monitoring server host, server connection, Databases, Tables, Views, Triggers and Events.

3.10 Designing Databases

The process of creating, implementing and maintaining the enterprise data in a system is known as Designing of databases. Better understanding of the application is necessary before designing the database. The performance and success of an application depends on good database design. MySQL provides performance dashboard, reports and statistics regarding the design of database.

Create Conceptual Design

It is the primary phase in database design, where detailed discussion about the creation of databases, tables, columns and data types is discussed based on the requirement of the application. As an end result the model is framed to attain the expectation of the application's end user.

- **Create Database**

The three major parts that forms a database are Tables, Queries and Views.

Tables - similar to an excel sheet, containing multiple rows and columns. Where each row is a record and each column is an attribute.

Queries – It is a question with multiple conditions posted to the database. The records in the database that satisfies the passed conditions are retrieved.

Views – A set of stored queries.

Example: create a database to store the personaldetails.

```
mysql> create database personaldetails;
Query Ok, 1 row affected
mysql> USE personaldetails;
Database changed.
```

The created database is listed using SHOW command. See Table 3.12

```
mysql> show databases;
```

Table 3.12 Database List

Database
employee
personaldetails
sports

- **Create Table**

In an application, each page reveals some functionality. Each such functions are designed to a table. For example, in an online shopping site like Amazon, multiple pages are maintained like customer profile, products, orders in cart and payment page. All these can be created as tables like Customer, Products, Order and Payment respectively.

- **Create Columns**

Each table will have many columns related to the functionality of the table. This column determines what values are stored in the table. For example, the Customer table contains the columns like firstname, lastname, phone, email, age, address and pincode. These columns hold the Customer information in the table. Each column is assigned with appropriate value type. The efficiency and performance of the table purely depends on the data types assigned to the columns.

- **Insert Rows**

Once the database is created, tables and the columns with the appropriate value type are defined. Then records are inserted to the table.

INSERT INTO TABLE_NAME (column1, column2, column3, ... columnN)

VALUES (value1, value2, value3, ...valueN);

3.11 SQL

SQL- Structured Query Language is a standard language used for accessing and manipulating databases. It is declared as standard by American National Standards Institute (ANSI) and International Organization for Standardization (ISO) in 1986 and 1987 respectively.

About SQL

- Though SQL is standard language, different versions are maintained to meet out the requirements. Few major functions performed using SQL are listed below:
- Executes queries against a database.
- Retrieves data from database.
- Inserts and updates records in a database
- Delete records from database.
- Creates new databases and new tables in a database.

Types of SQL Commands

Different SQL commands are available to perform various functions. SQL commands are classified into five major categories depending on their functionality. See Table 3.13

Data Definition Language (DDL)

The DDL commands are used to define database schema (Structure). Also to create and modify the structure of the database object in the database. CREATE, ALTER, DROP,RENAME and TRUNCATE commands belongs to this category.

Table: 3.13 SQL DDL COMMANDS List

Commands	Description
CREATE	Used to create database or tables
ALTER	Modifies the existing structure of database or table
DROP	Deletes a database or table.
RENAME	used to rename an existing object in the database
TRUNCATE	Used to delete all table records

Data Manipulation Language (DML)

These SQL commands deals with the manipulation of data present in the database. Most of SQL commands come under DML. INSERT, UPDATE, and DELETE commands belong to this category. See Table 3.14

Table: 3.14 SQL DML COMMANDS List

Commands	Description
INSERT	Adds new rows into database table.
UPDATE	modifies existing data with new data in a table.
DELETE	Deletes the records from the table.

Data Query Language (DQL)

SELECT is the only SQL command used to fetch or retrieve the data from database tables that come under DQL. See Table 3.15

Table: 3.15 SQL DQL COMMANDS List

Commands	Description
SELECT	Retrieve data from the table.

Transaction Control Language (TCL)

These SQL commands manage the transactions in SQL databases. It also helps to save the change into database permanently. COMMIT, ROLLBACK, SET TRANSACTION and SAVEPOINT commands belongs to this category. See Table 3.16

Table: 3.16 SQL TCL COMMANDS List

Commands	Description
COMMIT	Permanently saves into database.
ROLLBACK	Restore database to original form since the last COMMIT.
SET TRANSACTION	Sets the transaction properties such as read-write or read only access.
SAVE POINT	Used to temporarily save a transaction so that we can rollback to that point whenever required.

Data Control Language (DCL)

The SQL commands that implement security on database objects like table, view, stored procedure etc. GRANT and REVOKE commands belongs to this category. See Table 3.17

Table: 3.17 SQL DCL COMMANDS List

Commands	Description
Grant	used to give permission to specific users on specific database objects like table, view etc.
Revoke	used to take out permission from specific users on specific database objects like table, view etc.

3.12 Basic SQL Commands

Create/Drop /Selecting Database

Create Database – used to create new SQL Database. The Syntax and example to create studentDB is given below.

Syntax: CREATE database databasename;

Example: mysql>create database studentDB;

The database created now can be viewed using the following Syntax:

mysql>Show databases;

As a result, the newly created studentDB will be listed.

Drop Database – used to remove any of the existing SQL Database. The Syntax and example to delete student DB is given below.

Syntax: DROP database databasename;

Example: mysql>DROP database studentDB;

The deleted database will not be viewed,when we list all databases using the Syntax, Show databases;

Select Database – Many databases are available in the repository, from which the suitable database is selected using the below command.

Syntax: USE databasename;

Example: mysql>USE studentDB;

Once the database is selected, multiple operations are performed as per the needs of the application.

- Insert Record

  Database will have multiple tables. Tables are created using the create command with various fields added to it as per their need. Any table is complete only with the record available in it. So new row are added to the table using the Insert command. The Syntax and example for inserting new record into the table is given below:

Syntax 1:
INSERT INTO tablename (column1, column2, column3)

VALUES (value1, value2, value3);

Syntax 2:
INSERT INTO tablename VALUES (value1, value2, value3);

Consider we have a table named Biodata, which has three columns namely firstname, lastname and age. Now the new record is added to the table using either Syntax1 or Syntax2 as shown below.

mysql>INSERT INTO Biodata (firstname, lastname, age)

VALUES (Krishna, Sam, 10);

(or)

mysql>INSERT INTO Biodata VALUES (Krishna, Sam, 10);

- Select Record

From the multiple records available in the table, the enquired data are retrieved from the table-using the SELECT command with some conditions specified in it. We can retrieve all the fields of a record or specify the necessary fields in a table. The records of any table are retrieved using the SELECT Syntax given below in Table 3.18 and 3.19

Syntax1: SELECT * from tablename;

Example: **mysql**>SELECT * from Biodata;

Table:3.18 SQL Select Record List

Firstname	Lastname	Age
Krishna	S	10
Sugal	S	14
Arun	J	15
Mani	K	18

Syntax2: SELECT column1, column2 from tablename;

Example: **mysql**>SELECT firstname, age from Biodata;

Table:3.19 SQL Select Record List

Firstname	Age
Krishna	10
Sugal	14
Arun	15
Mani	18

Deleting Record

The existing record in a table is removed from the table using DELETE command. Entire record or specified columns in the table can be deleted. If we want to perform delete operation on specific columns, then the condition is given using WHERE clause. If the condition is not specified, then the entire data will be deleted. See Table 3.20 and 3.21

Syntax1: DELETE from tablename WHERE columnname=”value”;

Example: mysql>DELETE from Biodata WHERE firstname=”Mani”;

Table:3.20 SQL Delete Record List		
firstname	**lastname**	**age**
Krishna	S	10
Sugal	S	14
Arun	J	15

Syntax2: DELETE from tablename;

Example: mysql>DELETE from Biodata ;

Table:3.21 SQL Delete Record List		
firstname	**lastname**	**age**

Modifying Record

SQL provides us with modifying and updating the existing records in a table using UPDATE command. The age of Krishna in Biodata table is changed using the below Syntax.

Syntax1: UPDATE tablename

SET column1="new value"

Where column2="value2";

Example: mysql>UPDATE Biodata SET age=13 WHERE firstname="Krishna";

WHERE Clause

In SQL command WHERE clause is used to specify the selection criteria Based on that data's are retrieved or modified as per the query. In the WHERE conditions, operations like =, !=, >, >=, < , <= are used to frame the query statement. WHERE clause is used in SELECT and UPDATE query statement for the condition. The number of records updated in a table depends on the WHERE condition.

Using Operators

While forming the SQL query we use major operators like Arithmetic, Comparison and Logical in the WHERE clause. The purpose of each operator is listed below in Table 3.22.

Table:3.22 MySQL Operators	
Operator Type	**Operator**
Arithmetic Operator	+, -, *, /, %
Comparison Operator	=, !=, <, >, <>, >=, <=,
Logical Operator	AND, ANY, BETWEEN, EXISTS, IN, LIKE, NOT, OR, UNIQUE

Sorting Records

The Query results are listed in Ascending or Descending order using the command ORDER BY clause. In some databases the results are sorted by default in Ascending order and is given in Syntax1. The results are displayed in descending order as per Syntax2. See Table 3.23 and 3.24

Syntax1: select * from tablename ORDER BY columnname;

Example: select * from Biodata ORDER BY firstname;

Table:3.23 Select Record List		
firstname	**lastname**	**age**
Arun	J	15
Krishna	S	13
Sugal	S	14

Syntax2: select * from tablename ORDER BY columnname DESC;

Example: select * from Biodata ORDER BY firstname DESC;

Table:3.24 Select Record List

firstname	lastname	age
Sugal	S	14
Krishna	S	13
Arun	J	15

Grouping Records, Having Clause

In SQL we can have identical data in a group. Consider a table named Exams with fields Rollno, Subject and Marks. In the below table multiple rows of records are available for different subjects. Using the GROUP BY command, the rollno's are grouped and marks are added up against the Rollno with SUM(marks). See Table 3.25 and 3.26.

Table:3.25 Select Record List

Rollno	Subject	Marks
201901	Tamil	81
201904	English	75
201901	English	96
201903	Tamil	92
201902	Tamil	78
201904	Tamil	83
201903	English	81
201905	Tamil	89
201902	English	80
201905	English	90

Example: Select Rollno, SUM(Marks) from Exams GROUP BY Rollno;

Table:3.26 Select Record List

Rollno	Marks
201901	177
201902	158
201903	173
201904	158
201905	179

Joining Tables

When we have to select data from more than 2 tables SQL JOIN clause is used. Consider two tables Exams and Profile. The Exams table has the fields Rollno, Subject and Marks. The data of **Profile** table is shown below in Table 3.27 & 3.28.

Table: 3.27 Select JOIN Record List

Rollno	Name	Hobby
201901	Krishna	Gardening
201902	Sugal	Photography
201903	Charles	Reading
201904	Venilla	Singing
201905	Pragathi	Painting

Example: SELECT Profile.Name, Profile. Hobby, SUM(Exams.Marks)As Total

FROM Profile, Exams

WHERE Profile.Rollno = Exams.Rollno

GROUP BY Profile.Name, Profile.Hobby;

Table: 3.28 Select Join Record List		
Name	**Hobby**	**Total**
Krishna	Gardening	177
Sugal	Photography	158
Charles	Reading	173
Venilla	Singing	158
Pragathi	Painting	179

In the Query Statement, the marks are added and listed under the column name Total for each of the Rollno from both the tables.

GLOSSARY

Backup	It is a program or process of copying table contents into a file for future reference. It's a challenging task for DBA's
Primary Key	This key of relational table identifies each record in the table in a unique way.
Relationship	There exists a relationship between two tables when the foreign key of one table references primary key of other table.
Cardinality	It is defined as the number of different values in any given table column
Open Source	Open source – refers to the design that is publicly accessible by people for modification and sharing
SQL	It (Structured query Language) is a programming language designed mainly for managing data in RDBMS
Record	Record is referred in a table, which are composed of fields.
Query	In SQL, all commands are named as query. The query statement is executed against the databases.
Join	Retrieves data from two or more tables, by referencing columns in the tables that hold identical values

EVALUATION

SHCRGH

Part - I

Choose the correct answer

1. Which language is used to request information from a Database?
   a) Relational
   b) Structural
   c) Query
   d) Compiler
2. The ---------- diagram gives a logical structure of the database graphically?
   a) Entity-Relationship
   b) Entity
   c) Architectural Representation
   d) Database
3. An entity set that does not have enough attributes to form primary key is known as
   a) Strong entity set
   b) Weak entity set
   c) Identity set
   d) Owner set
4. ---------- Command is used to delete a database.
   a) Delete database database_name
   b) Delete database_name
   c) drop database database_name
   d) drop database_name
5. MySQL belongs to which category of DBMS?
   a) Object Oriented
   b) Hierarchical
   c) Relational
   d) Network
6. MySQL is freely available and is open source.
   a) True
   b) False
7. ---------- represents a "tuple" in a relational database?
   a) Table
   b) Row
   c) Column
   d) Object
8. Communication is established with MySQL using
   a) SQL
   b) Network calls
   c) Java
   d) API's
9. Which is the MySQL instance responsible for data processing?
   a) MySQL Client
   b) MySQL Server
   c) SQL
   d) Server Daemon Program
10. The structure representing the organizational view of entire database is known as -------- in MySQL database.
    a) Schema b) View
    c) Instance d) Table

Part - II

Short Answers

1. Define Data Model and list the types of data model used.
2. List few disadvantages of file processing system.
3. Define Single and multi valued attributes.
4. List any two DDL and DML commands with its Syntax.
5. What are the ACID properties?
6. Which command is used to make permanent changes done by a transaction?
7. What is view in SQL?
8. Write the difference between SQL and MySQL.
9. What is Relationship and List its types.
10. State few advantages of Relational databases.

Part - III

Explain in Brief Answer

1. Explain on Evolution of DBMS.
2. What is relationship in databases? List its types.
3. Discuss on Cardinality in DBMS.
4. List any 5 privileges available in MySQL for the User.
5. Write few commands used by DBA to control the entire database.

Part - IV

Explain in detail

1. Discuss on various database models available in DBMS.
2. List the basic concepts of ER Model with suitable example.
3. Discuss in detail on various types of attributes in DBMS.
4. Write a note on open source software tools available in MySQL Administration.
5. Explain the DDL command of their functions in SQL.