

# DATABASE MANAGEMENT SYSTEMS

## INTRODUCTION

Lecture One

**JIMMY M.R.**

# Lecture Objectives

- In this lecture, one will learn:
  - Introduction to DBMS
  - Advantages and Disadvantages of DBMS
  - Basic DBMS Terminologies
  - Data and Database
  - Evolution of Databases
  - Components of a Database
  - Application of Databases
  - File-Oriented and Database Oriented Approaches
  - Instance and Schemas
  - Data Models

© Muwawu JR 2024 - - - rickmuwa@gmail.com

# Introduction to DBMS

- **Database Management System** (**DBMS**) is a collection of interrelated data and a set of programs to access those data.
- The primary goal of a DBMS is to provide a way to store and retrieve database information that is both ***convenient*** and ***efficient***.

# Advantages of DBMS

- The data is stored in a neater way and hence, more data can be stored.
- A DBMS is a highly secure platform so confidential and high-risk data can also be stored and accessed, securely.
- DBMS makes handling of data very simple.
- Data inconsistency is greatly reduced by a well-designed DBMS.
- Data can be accessed quickly.

© Muwawu JR 2024 - - - rickmuwa@gmail.com

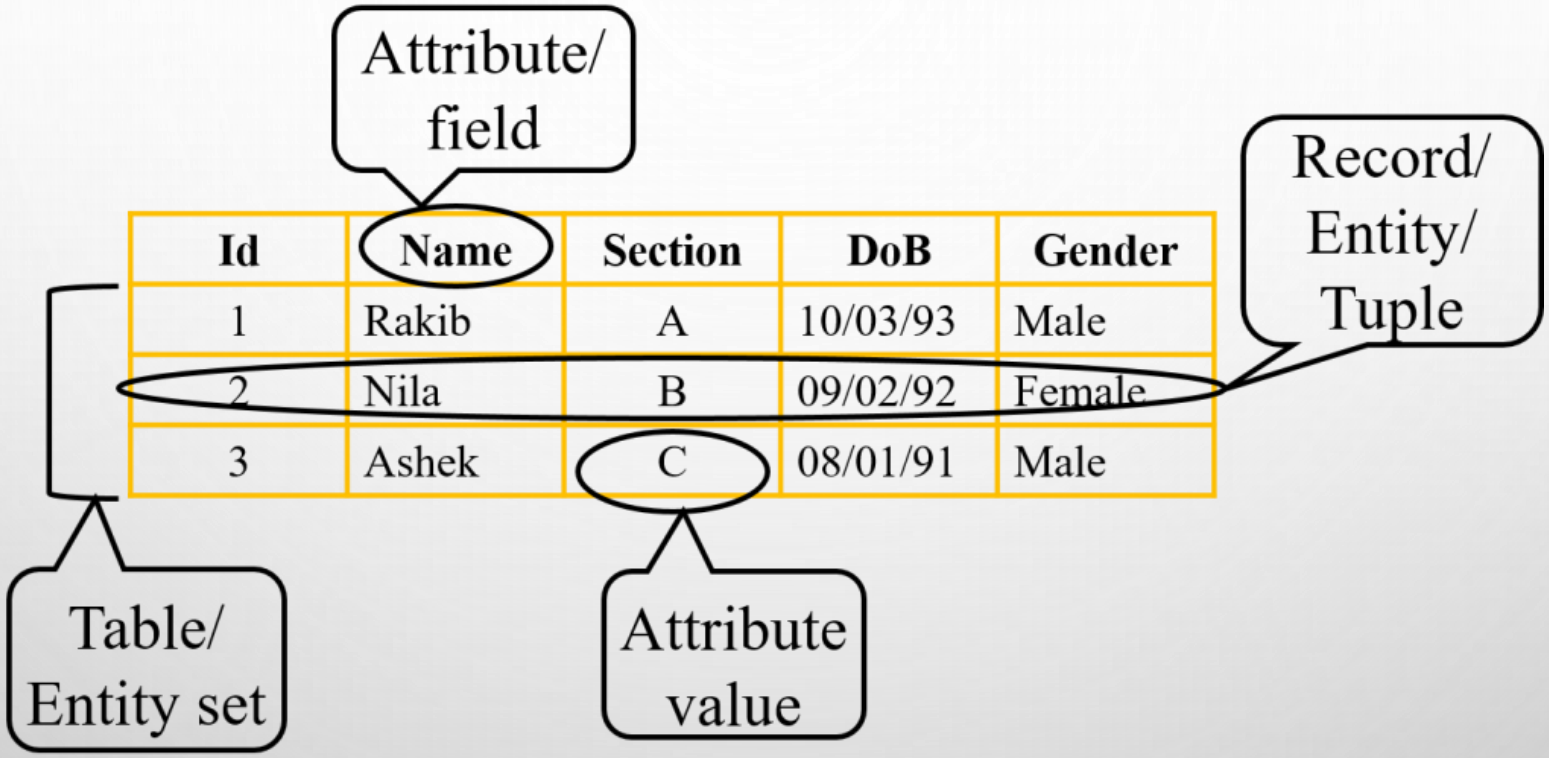
# Disadvantages of DBMS

- Maintaining the software and hardware required for a DBMS is usually expensive.
- The more data is fed into the DBMS, the more disk space it occupies.
- Using DBMS could seem very difficult for someone from a non-technical background.
- Since all the data is stored in one DBMS, if the software fails, all the data of the organization could be lost.

# Basic DBMS Terminologies

- **Table:** is a collection of data elements organised in terms of rows and columns.
- **Record**: is a single entry or set of related data in a table or **row**.
- **Tuple**: is a single entry in a table.
- **Field** is used to refer to the individual cells within a row or column.
- **File**: is a collection of records.
- **Attribute**: refers to the specific field or column in a table.
- **Entity**: is a piece of data that is stored in the database, e.g. person, place, thing, or even an event.

# Basic DBMS Terminologies




| Id | Name | Section | DoB | Gender |
|---|---|---|---|---|
| 1 | Rakib | A | 10/03/93 | Male |
| 2 | Nila | B | 09/02/92 | Female |
| 3 | Ashek | C | 08/01/91 | Male |

# Data

- **Data** is unprocessed information that is collected in various formats such as numbers, text, media, and others.
- Data is also be defined as raw facts that can be recorded.
- **Datum** – a unit of data.
- **Information** is interpreted data.
- **Knowledge** refers to the practical use of information.
- Data can be name, age, and educational qualification of its students, details of the various courses.

© Muwawu JR 2024 - - - rickmuwa@gmail.com

# Database

- A **database** is a systematic or organized collection of related information that is stored in such a way that it can be easily accessed, retrieved, managed, and updated.
- Data in a database is organized into **rows** and **columns** forming a table. Indexing the data makes it easy to find and retrieve it again as and when required.
- Databases are managed by a **Database Management System** (DBMS).
- To interact with and manipulate the data stored in these databases, a widely used language called **Structured Query Language** (SQL) is utilized.

# Evolution of Databases

- The database started with a file-based system about 50 years ago. In due time, it has gone through generations of evolution.
- Databases were first introduced in 1968 as flat-file-based databases.
- Then the Hierarchical Database came into existence and lasted until 1980. IBM's first database, IMS (Information Management System) was based on this.
- Charles Bachman developed the first Network data model, called Integrated Data Store (IDS). It was introduced in the early 1960s and standardized in 1971.
- In 1970, the Relational Database was introduced.
- Today, it is the era of Relational Databases and Database Management.

© Muwawu JR 2024 - - - rickmuwa@gmail.com

# Components of a Database

- **Hardware:** Physical electronic devices such as storage devices, I/O devices, and many more. It can act as an interface between computers and real-world systems.

- **Software:** Programs for managing and controlling the overall database. DBMS itself is software. The OS, the database application programs that allow data access in DBMS, the network software that shares data, etc. are all examples.

- **Data:** It is the information that is gathered, stored, accessed, and processed by a DBMS, e.g., actual data, operational data, and metadata.

# Components of a Database

- **Procedure:** It is the specific set of instructions and rules to use a database for designing and running the DBMS, as well as to instruct users on how to operate and manage it.

- **Database access language:** This helps export data to and access the same from the database. To enter new data or update or retrieve the data from the database, you can write commands in the database access language. DBMS then displays the results in a user-readable form.

# Application of Databases

| Field | Application |
|---|---|
| • Education | • Student information, courses, grades |
| • Library | • Book information, issue date |
| • Accounts | • Account information, transactions |
| • Railways | • Reservation information, tickets, train schedules |
| • Banking | • Client information, account, activities, loans |
| • E-Commerce | • Inventory information, purchases, invoices |
| • Manufacturing | • Supplier information, bills, inventory |

© Muwawu JR 2024 - - - rickmuwa@gmail.com

# Types of Databases

- **Relational database**: is a collection of information that organizes data in predefined relationships where data is stored in one or more tables (or "relations") of columns and rows

- **Distributed database**: It has two or more files located in different places. The database may be in the same physical location on multiple computers or scattered over different networks.

- **Cloud database**: Cloud database runs on a Cloud Computing platform, and access is provided '**as a service**.'

# Types of Databases

- **Operational database**: also known as OLTP or online transactional processing database, it is designed to create or update big amounts of data and store transactions performed by multiple users in real-time.

- **Data warehouses**: it is a central repository for data. It holds current and historical data in a single location for analytical reporting throughout the enterprise.

- **Centralization database**: is located, stored, and maintained in a single centralized location, for example, a mainframe computer, desktop, or server CPU.

# Advantages of Databases

- Minimum data redundancy
- Improved data security
- Increased consistency
- Lower updating errors
- Reduced costs of data entry, data storage, and data retrieval
- Improved data access using host and query languages
- Higher data integrity from application programs

# File Oriented Approach

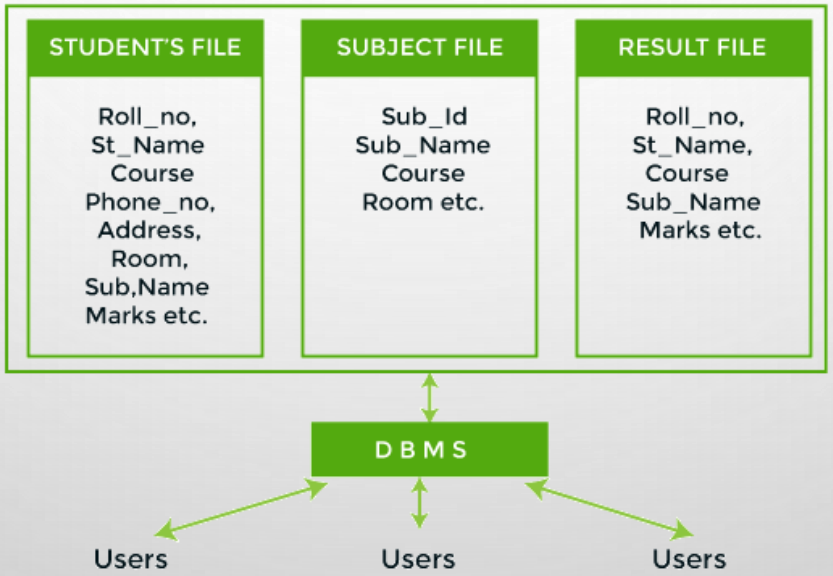
- In earlier days, data was stored manually, using pen and paper, but after computer was discovered, the same task could be done by using files.
- These systems stored groups of records in separate files, and so they were called **File Processing Systems**.
- **File Processing System** (FPS) is a way of storing, retrieving and manipulating data which is present in various files.
- A computer **File** is a resource which uniquely records data, in a storage device in a computer.
- There are various formats in which data can be stored. e.g. Text files can be stored in .txt format while pictures can be stored in .jpeg format etc.

# Disadvantages of File Oriented Approach

- Data redundancy and inconsistency.
- Difficulty in accessing data.
- Data isolation, data are scattered in various files.
- Integrity problems. data values must satisfy certain types of consistency constraints.
- Difficulty in recovery of corrupt data – Recovery or backup of lost and corrupt data is nearly impossible.
- Unauthorized Access – Anyone who gets access to the file can read or modify the data.
- Lack of Atomicity – operations either takes place as a whole or not

# Database Oriented Approach

- **Data Oriented Approach** is a well-organized collection of data that are related in a meaningful way which can be accessed by different users but stored only once in a system.



# Differences between DOA and FOA

| File Oriented Approach | Database Oriented Approach |
| --- | --- |
| • Data is distributed in many files, and it may be of different formats, so it isn't easy to share data. | • Due to the centralized approach, data sharing is easy. |
| • Data is distributed in many files, and it may be of different formats, so it isn't easy to share data. | • Due to the centralized approach, data sharing is easy. |
| • It isn't easy to protect a file under the file system. | • DOA provides a good protection mechanism. |
| • There exists a lot of duplication of data which may lead to inconsistency. | • Data redundancy and inconsistency are controlled. |

# Instance in DBMS

- **Instances** are the collection of information stored at a particular moment in a database or the current state of a database

- **Example**

  Let's say a table named students in the database called university, has 50 records, so the instance of the database has 50 records for now and if tomorrow 30 records are added, so tomorrow the instance will have 80 records. Therefore, the data stored in the database at a particular moment is know as the **Instance**.

© Muwawu JR 2024 - - - rickmuwa@gmail.com

# Schema in DBMS

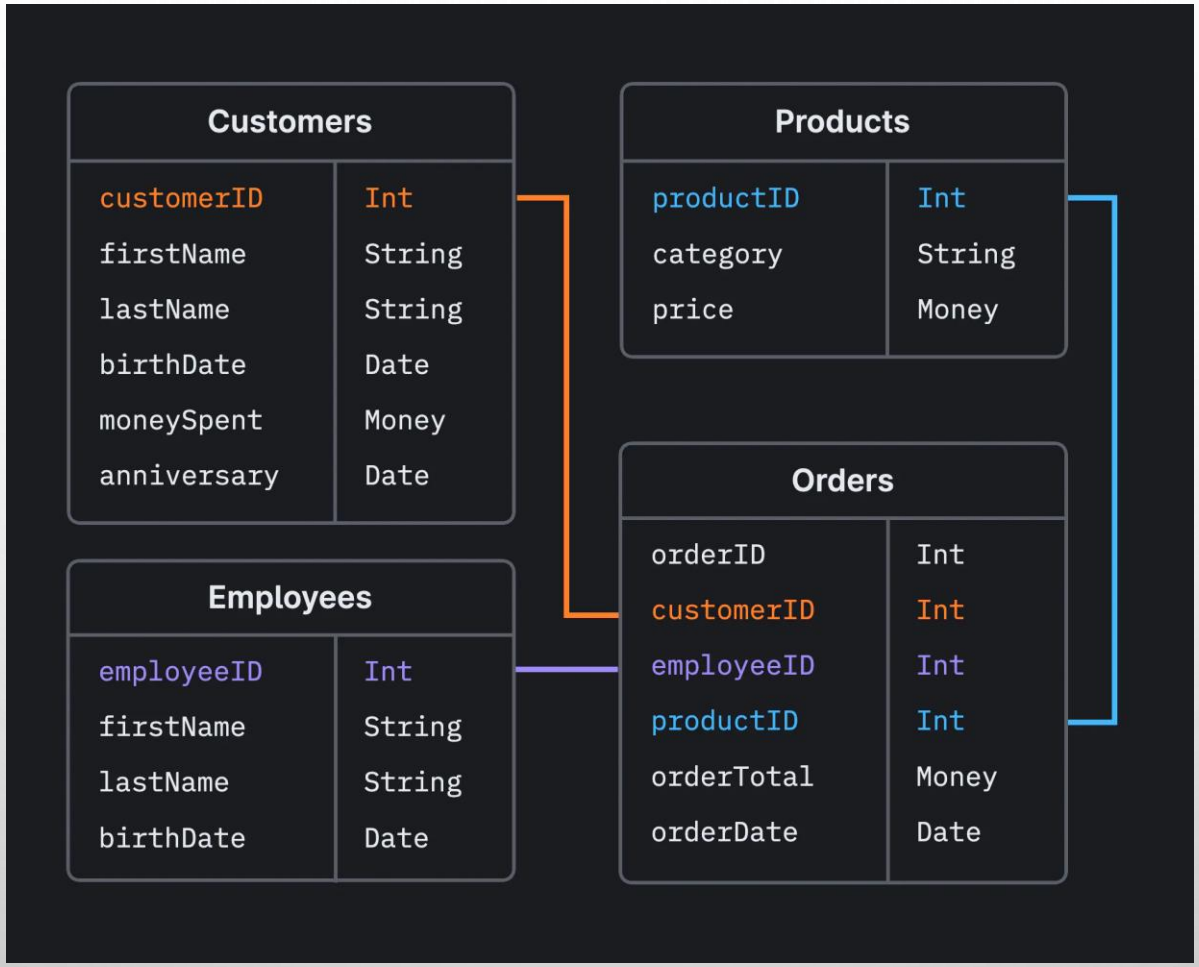
- **Schema**: is the overall design of a database showing table names, fields, data types, and the relationships between these entities.

- Schemas commonly use visual representations to communicate the architecture of the database

- **Example**

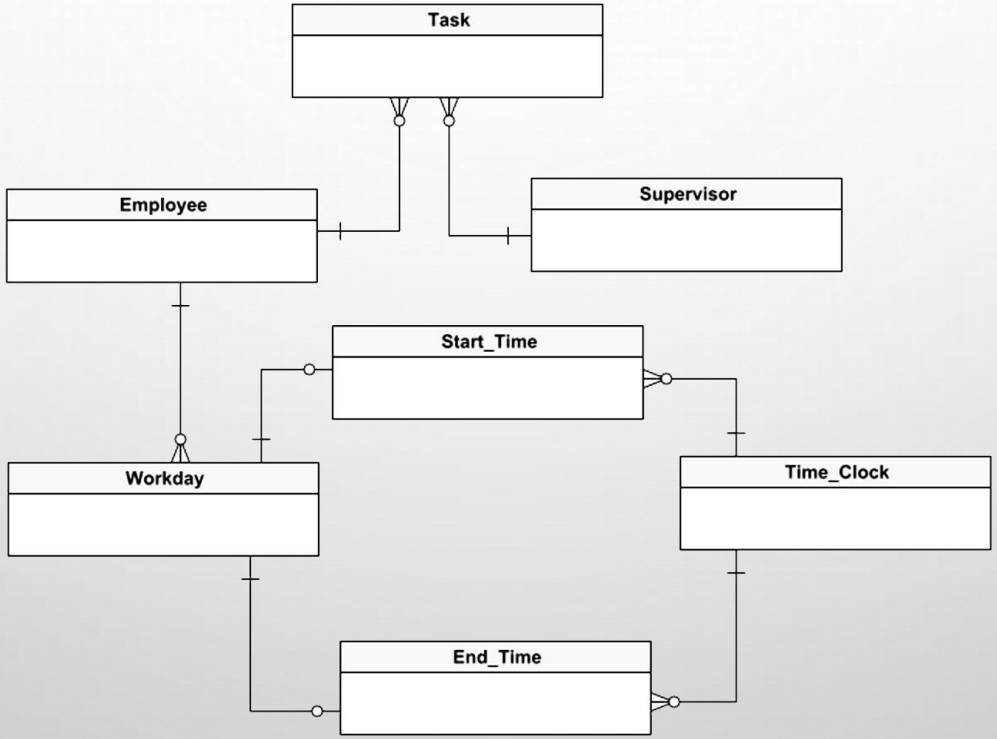
  Let's say a table named students in the database called university, has 50 records, so the instance of the database has 50 records for now and if tomorrow 30 records are added, so tomorrow the instance will have 80 records. Therefore, the data stored in the database at a particular moment is know as the **Instance**.

# Schema in DBMS



# Conceptual Database Schema

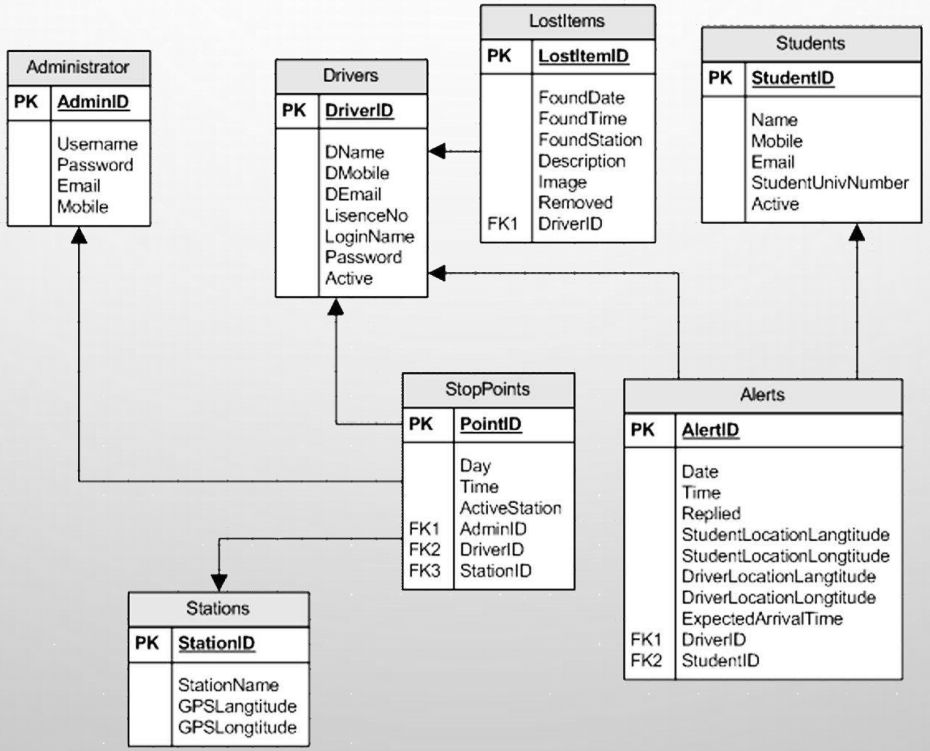
- **Conceptual schema** offers a big-picture view of what the system will contain, how it will be organized, and which business rules are involved.




© Muwawu JR 2024 - - - rickmuwa@gmail.com

# Logical Database Schema

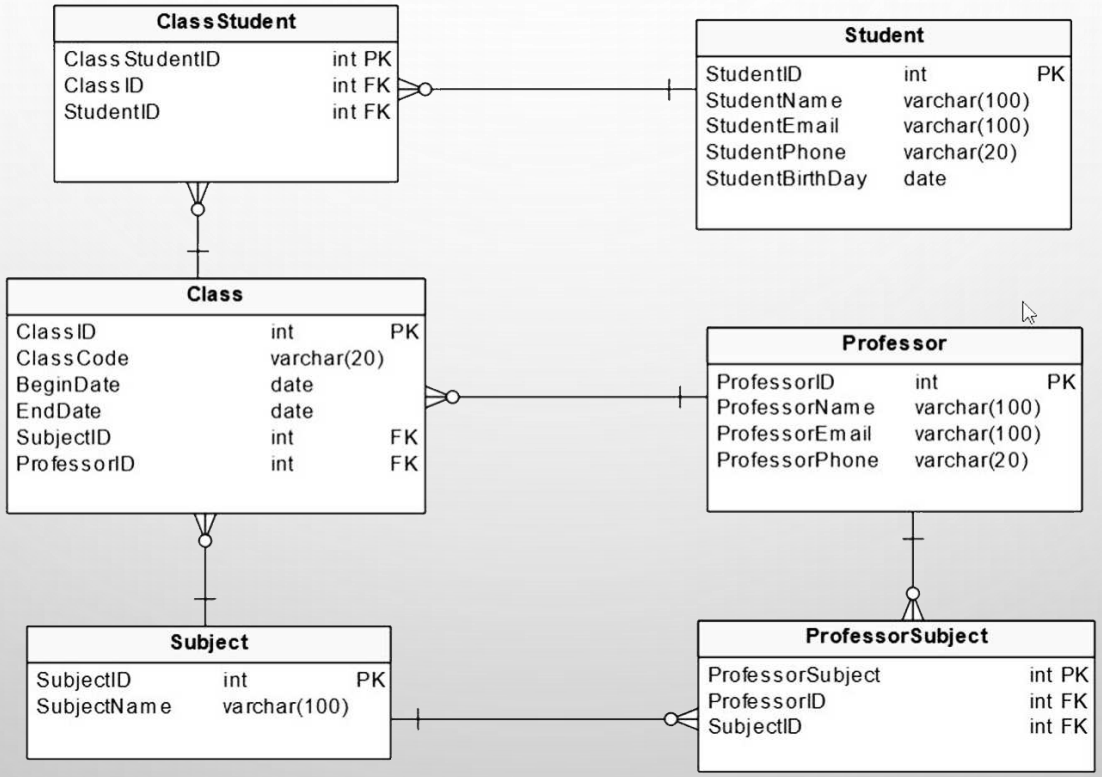
- **Logical database schema** is less abstract, compared to conceptual schemas. They clearly define schema objects with information, such as table names, field names, entity relationships, and integrity constraints




© Muwawu JR 2024 - - - rickmuwa@gmail.com

# Physical Database Schema

- **Physical database schema** provides technical information in its schema such as table names, field names, data types, entity relationships.

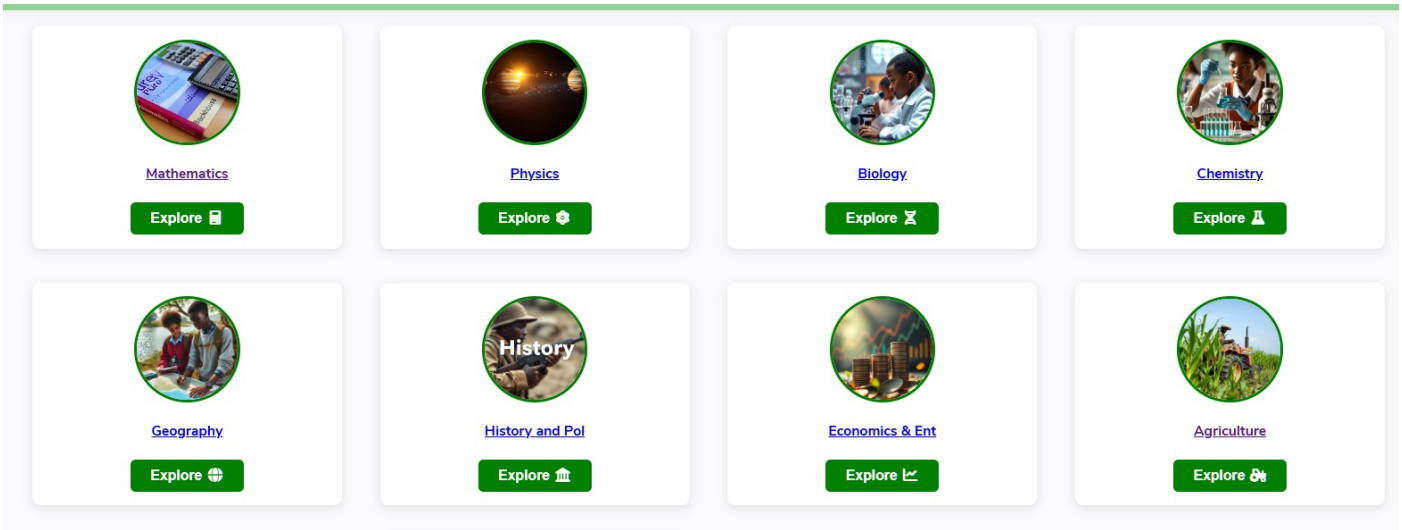



© Muwawu JR 2024 - - - rickmuwa@gmail.com




© Muwawu JR 2024 - - - rickmuwa@gmail.com

This document was downloaded from https://exoticnotes.com




**NOTES for Secondary (S1 – S4)**
https://exoticnotes.com/notes/secondary/o-level/

**NOTES for Secondary (S5 – S6)**
https://exoticnotes.com/notes/secondary/a-level/

**PAPERS and Assessment items for Secondary (S1 – S6)**
https://exoticnotes.com/papers/secondary/

**Scenario Questions (with some Answers) (S1 – S6)**
https://exoticnotes.com/scenario-questions

**For New A level Curriculum (Notes, Assessment and Constructs, etc)**
https://exoticnotes.com/new-a-level-curriculum

**For New O level Curriculum (Notes, Scenarios, Papers, AOI, Syllabi, etc)**
https://exoticnotes.com/new-o-level-curriculum