Republic of Rwanda
Ministry of Education

RTB | RWANDA TVET BOARD

TVET LEVEL IV

SOFTWARE DEVELOPMENT

BASICS AND FUNDAMENTALS OF DATABASE

TRAINEE MANUAL

RTB | RWANDA TVET BOARD

giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

MODULE:

BASICS AND FUNDAMENTALS OF DATABASE

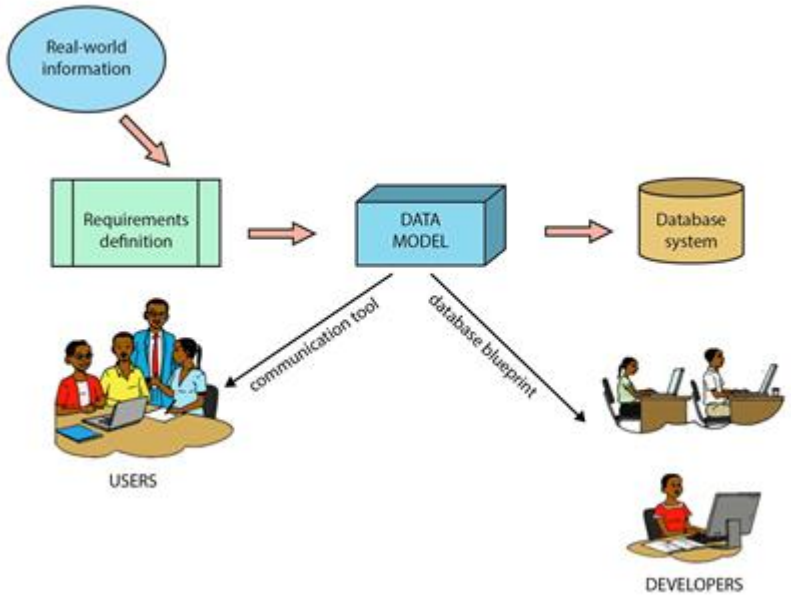
Unit 1: Analyze database requirements

Unit 2: Design database

Unit 3: Create database

UNIT 1: ANALYZE DATABASE REQUIREMENTS

Illustration of Learning Unit



Picture Brief

On the illustration above, from left to the right side , the first figure in circle design represents real-world information, The next figure in rectangle design, represents the requirements definition, the next figure in cube design, represents the data model, and the last figure in cylinder design represents the database system.

Then People bellow the figure of Requirements design represents, the team of users who are sharing ideas on how they can design data model, and the people under the figure of database, the first two people you are seeing, are people who use the database system, then under them, we have the developer who transform data model into database system.

Topics

1.1 Definition of database key terms, types of database models and relationship.

1.2Reviewing organizational and task requirements to identify user requirements.

1.3Determining the information that the database is required to hold.

Unit Summary:

This unit describes the skills, knowledge, and attitudes required to define database terms and types, to review organizational and task requirements to identify user requirements and to determine the information the database is required to hold.

Self-Assessment: Unit 1

1. Look at the illustration. What is happening? What do you think this unit will be about?

2. Fill in the self assessment below.

There are no right or wrong ways to answer this survey. It is for your own use during this course. Think about yourself: do you think you can do this? How well? Read the statements across the top. Put a check in column that best represents your situation. At the end of this unit, we'll take this survey again.

My experience Knowledge, skills and attitudes	I don't have any experience doing this.	I know a little about this.	I have some experience doing this.	I have a lot of experience with this.	I am confident in my ability to do this.
Define database terms					
Classify database models					
Identify entity relationships					
List data collection techniques and methods.					
Describe the types of requirements.					
To plan managerial levels of a company/organization.					
Describe database elements					
List database properties					
Identify fields data types					
Explain database applications					

Topic 1.1: Definition of database key terms, types of database models and relationship.

Key Competencies:

Knowledge	Skills	Attitudes
1. Describe database terms	**1.** Identify database terms	**1.** Be analyst
2. List database models	**2.** Differentiate database models	**2.** Be detail-oriented
3. Name entity relationships	**3.** Identity entity relationships	**3.** Be attentive

GettingStarted: What do we know and where are wegoing?

Task:1. Tell what you know about database.

2.What is the relationship between database and the topic?

Activity 1: ProblemSolving

A fashion shop manager wants to keep the information about shop items (products), customers, and suppliers. You are asked to do the following tasks:

1. Write possible entities.
2. Show how entities are related.
3. List attributes of products

Key Facts 1.1

1. INTRODUCTION TO DATABASE CONCEPTS

1.1 Definitions of terms

- ✓ Database: is an organized collection of data, generally stored and accessed electronically from a computer system.
- ✓ Data: mean known facts that can be recorded and that have implicit meaning. For example, consider the names, telephone numbers, and addresses of the people you know.
- ✓ Entity: is an object either concrete or abstract in the real world that we want to model and store the information about. For example employee, student, car, climate. Etc
- ✓ Attribute: also known as field or column is a property or a characteristic of an entity. For example, age, weight, height are your characteristics (Person).
- ✓ Tuple: is a single rowof an entity which contains a data set of attributes.
- ✓ Database model: refers to the logical structure, representation or layout of a database and how the data will e stored, managed and processed within it.

1.2. Types of database models

1.2.1.Relational model

The relational model represents the database as a collection of relations.
Informally, each relation resembles a table of values
When a relation is thought of as a table of values, each row in the table represents a collection of related data values.

1.2.2.Hierarchical model

Hierarchical model is a data model in which the data is organized into a tree-like structure. The data are stored as records which are connected to one another through links.

1.2.3.Network data model

A network model is a database model that is designed as a flexible approach to representing objects and their relationships. A unique feature of the network model is its schema, which is viewed as a graph where relationship types are arcs and object types are nodes.

1.2.4.Object-oriented model

Object oriented data model is based upon real world situations. These situations are represented as objects, with different attributes. All these object have multiple relationships between them.

1.3.Database relationships (Or entity relationships)

Database relationship relies on both concepts of cardinality and ordinarily.

Cardinality refers to the maximum number of times an instance in one entity can relate to instances of another entity. Ordinality, on the other hand, is the minimum number of times an instance in one entity can be associated with an instance in the related entity. The following are possible database relationships:

1.3.1.One -to-one relationship

Entities A and B are related such one element of A may only be linked to one element of B, and vice versa. Example: Husband (Entity A) and Wife(Entity B)

1.3.2.One-to-many relationship

Entities A and B are related such one element of A may be linked to many elements of B but element of B is linked to only one element of A. Example: Father (entity A) and Children (entity B).

1.3.3.Many-to-many relationship

Entities A and B are related such one element of A may be linked to many elements of B, and vice versa. Example: Students (A) and Courses(B).

Activity 2: Guided Practice

A fashion shop manager wants to keep the information about shop items (products), customers, and suppliers. Based ondatabase schema, together with the trainer, perform the following tasks:

a. Identify relationship cardinalities.
b. List each entity attributes
c. Demonstrate columns and rows in relational data model.

Activity 3: Application

A restaurant manager wants to keep into a database the information related to restaurant menu, employees, and clients. You are asked to do the following tasks.

1. Identify the database entities.
2. Examine the cardinality and ordinarily of related entities.
3. List five possible attributes for each entity.

Points to Remember

- Entity relationship Diagram (ERD)
- Parent and child entities
- Flat file database

Formative Assessment

1. Match database term with a right definition.

Term	Definition
Database	A property or a characteristic of an entity
Entity	organized collection of data, generally stored and accessed electronically from a computer system
Attribute	An object either concrete or abstract in the real world that we want to model and store the information about.

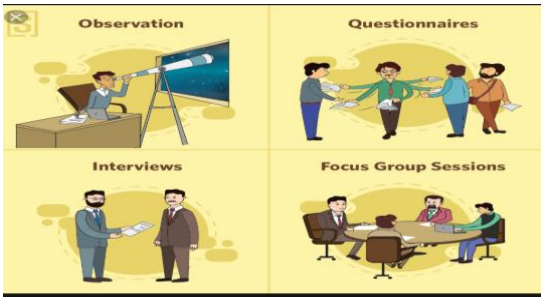
2. List database models that you know.
3. Write two examples of entities for each entity relationship cardinality.

TOPIC 1.2: REVIEWING ORGANIZATIONAL AND TASK REQUIREMENTS TO IDENTIFY USER REQUIREMENTS.

Key Competencies

Knowledge	Skills	Attitudes
1. Describe techniques and methods of data collection	**1.** Apply techniques and methods of data collection	**1.** Be polite
2. List the types of requirements	**2.** Differentiate types of requirements	**2.** Be motivated
3. Tell management levels within organization/company	**3.** Construct managerial levels within organization/company	**3.** Be analyst

GettingStarted: What do we know and where are wegoing?



Task: Tell what data collection techniques used in the picture above are.

Activity 1: Problem Solving

Task:A website developer wants to gather data from rice farmer cooperative, for initially developing cooperative website.

What are the techniques and methods of data collection do you think He will be using?

Key Facts 1.2

1. Data collection techniques and methods
   1.1 . Interview

An interview is a conversation where questions are asked and answers are given. It refers to one-on-one conversation between an interviewer and an interviewee. Interview questions include open-ended questions, closed-ended questions, and probing questions.

1.2. Documentation

Documentation is a set of documents provided on papers, or online, or by digital or analog media to give user guides or information about an organization or a subject.

1.3. Questionnaire

A questionnaire is a set of printed or written questions with a choice of answers, devised for the purpose of a survey or statistical study.

1.4. Observation

Observation is the action or process of carefully watching someone or something.

2. Types of requirements

2.1. Functional requirements

The functional requirements are specification documents of operations and activities that a system must be able to perform. Functional Requirements should include: Descriptions of data to be entered into the system. Descriptions of operations performed by each screen.

2.2. Non-functional requirements

Nonfunctional Requirements. Nonfunctional Requirements (NFRs)
define system attributes such as security, reliability, performance, maintainability, scalability, and usability... Also known as system qualities.

3.Management levels of an organization/company

3.1. Upper/top level management

Top management is made up of senior-level executives of an organization, or those positions that hold the most responsibility. It is focused on market positioning through long-range strategic planning and decision making.

3.2. Middle level management

Middle management is the intermediate leadership level of a hierarchicalorganization and deals with short range decision-making within area or branches of responsibility,to implement projects that will meet the strategic objectives of the organization.

3.3. Lower/operational management

Lower management in a business refers generally to daily operations or transactions of an organization.

Activity 2: Guided Practice

A website developer has gathered data from rice farmer cooperative, for initial development of cooperative website using interview. Specifications such as content publication, user guidance, links to social media, accessibility from different devices, hosting domain name are given by cooperative committee. Together with the trainer, you are asked to answer the questions below:

a. Write two examples of questions for each interview question type.
b. Select functional and non-functional requirements from given specification.

Activity 3: Application

Task: From formed groups, each group selects an interviewer to gather information required to develop a website of group choice. The procedure is that the interviewer conducts the interview with teammates**.** You are asked to report the following information into a table:

a. Functional and non-functional requirements
b. Identify other data collection techniques and methods rather than interview.

Points to Remember

- Honest
- Confidentiality
- Self-confidence

Formative Assessment

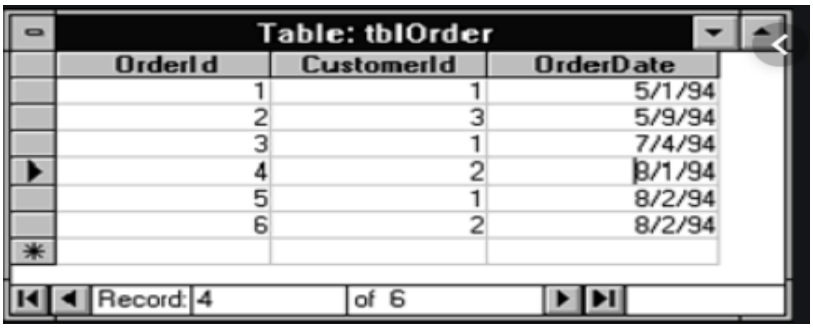
1. Identify different data collection techniques.
2. Identify various types of requirements
3. Explain management levels of an organization.

TOPIC 1.3: DETERMINATION OF THE INFORMATION THAT THE DATABASE IS REQUIRED TO HOLD

Key Competencies

Knowledge	Skills	Attitudes
1. Describe database elements	1. determine database elements	1.Be detail-oriented
2. List data types	2. determine data types	2.Be decisive
3. Application of database	3. Analyze Application of database	3.Be creative

GettingStarted: What do we know and where are wegoing?



Task: According to you what does the picture above means to you in database. Discuss about it in small groups

Activity 1: ProblemSolving

An enterprise wants to record the information about **Employees**(EmployeeID,Firstname, Lastname, gender, phone, email), **Payroll**(PayrollID, EmployeeID,Date, amount). You are asked to do the following tasks:

Task: 1. To draw both tables **Employees** and **Payroll** with their attributes inparentheses**.**

2. Fill in the tables five rows/tuples of your choice.

Key Facts 1.3

1. Elements of a database

1.1. Tables

A database table is a structure that organizes data into rows and columns.

Tables are similar to a worksheet in spreadsheet applications such as Microsoft Excel.

1.2. Properties

Database properties are also known as transaction properties, ACID for short. Here a transaction means a single logical unity executed to define a particular action on a database.

1.2.1. Atomicity: It means that either the entire transaction takes place at once or doesn't happen at all. There is no midway. It involves both operations:
—**Abort**: If a transaction aborts, changes made to database are not visible.
—**Commit**: If a transaction commits, changes made are visible.

1.2.2 **Consistency:** This means that integrity constraints must be maintained so that the database is consistent before and after the transaction. It refers to correctness of a database.

1.2.3. **Isolation:**This property ensures that multiple transactions can occur concurrently without leading to inconsistency of database state. Transactions occur independently without interference.

1.2.4. **Durability:**This property ensures that once the transaction has completed execution, the updates and modifications to the database are stored in and written to disk and they persist even if system failure occurs.

1.3. Records

A record is composed of fields and contains all the data about one particular table instance such as person, product. Etc

1.4.Fields

The database fields are the names of the columns where different pieces data of a record are stored.

2. Data types

Data types define the type of value that can be stored in a table column. The table below illustrate common SQL data types:

Data type	Description	Example (Value)
Character	Variablelength character (Varchar)	John
	Fixedlengthcharacter (char)	
Number	Integers	121
	Floating point numbers	625.102
Date	Short and long date formats	1/1/2019

3. Application of database

Database applications are software programs designed to collect, manage and disseminate information efficiently. Many enterprises and small business owners create databases to manage business information.

For examples:

- Accounting Applications

An accounting system is a custom database application used to manage financial data. Custom forms are used to record assets, liabilities, inventory and the transactions between customers and suppliers.

- Education applications

Most of education applications are used to deliver online trainings, and online scholarship applications. Etc.

Activity 2: Guided Practice

An enterprise wants to record the information about **Employees**(EmployeeID,Firstname, Lastname, gender, phone, email), **Payroll**(PayrollID, EmployeeID,Date, amount). Together with the trainer, you are asked to do the following tasks:

Task: 1. To draw both tables **Employees** and **Payroll** with their attributes inparentheses

2.Fill in the tables five rows/tuples of your choice.

3. Specify attributes data types

Activity 3: Application

A school database includes **students, courses,** and **marks** tables. You are invited to:

Task: 1. To draw the above tables and give them possible fields.

2.Specify fields data types for each table.

3. Write ten examples of database applications from business industry.

Points to Remember

- ACID
- A field only receives values of one data type
- Database never becomes full.

Formative Assessment

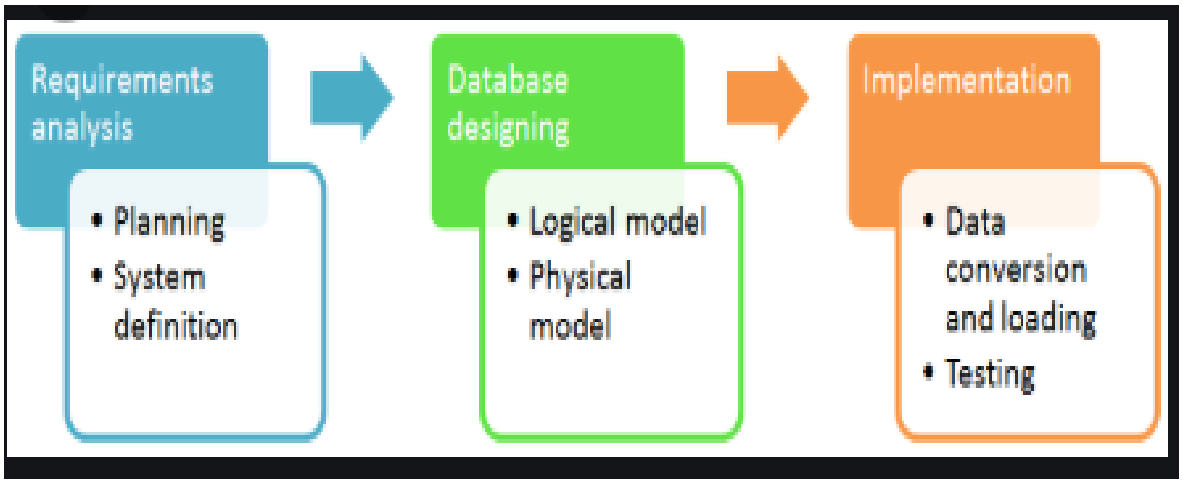
1. Name the data type of each of the following table field:
   A. Date_Of_Birth
   B. Age
   C. Middle_name
   D. Height(in meter)
   E. Country
2. List database elements that you know.

Self Reflection

Areas of strength	Areas for improvement	Actions to be taken to improve
1. **2.**	**1.** **2.**	**1.** **2.**

Unit 2: DESIGN A DATABASE

Illustration of Learning Unit 2



Picture Brief

The picture above is made by three couples of front and back rounded rectangles. The back rounded rectangles map database design phases which are requirement analysis (conception), database designing (Logical phase), and Implementation (physical phase), read left to right in arrow direction.

The front rounded rectangles indicate activities of each phase named by their titles: (Planning, system definition), (Logical model, physical model), (data conversion and loading, testing).

Topics

2.1. Designing an entity relationship diagram (ERD)

2.2. Identifying the constraints

2.3. Developing data dictionary

Unit Summary: This unit describes the skills, knowledge, and attitudes required to create an Entity Relationship Diagram of an organizational database, to identify constraints required, and to document data used into a data dictionary.

Self-Assessment: Unit 2

1. Look at the illustration. What is happening? What do you think this unit will be about? What topics might be covered?

2. Fill in the self assessment below.

There are no rights or wrong ways to answer this survey. It is for your own use during this course. Think about yourself: do you think you can do this? How well? Read the statements across the top. Put a check in column that best represents your situation. At the end of this unit, we'll take this survey again.

My experience Knowledge, skills and attitudes	I don't have any experience doing this.	I know a little about this.	I have some experience doing this.	I have a lot of experience with this.	I am confident in my ability to do this.
Construct Entity Relationship Diagram					
Differentiate database design abstraction levels					

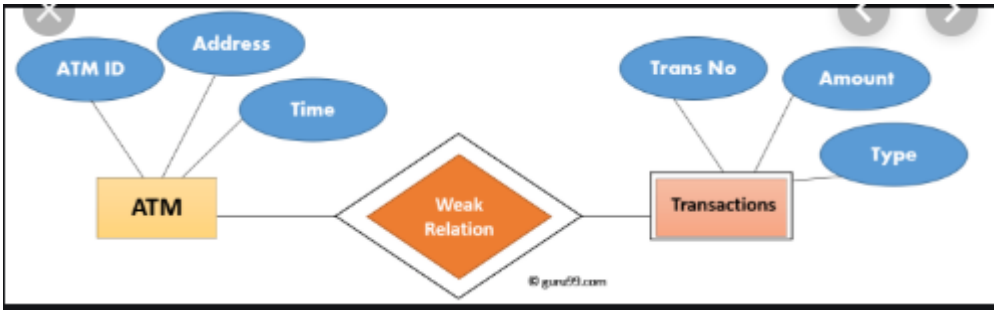
My experience Knowledge, skills and attitudes	I don't have any experience doing this.	I know a little about this.	I have some experience doing this.	I have a lot of experience with this.	I am confident in my ability to do this.
Evaluate system metadata					
Compare data constraints					
Evaluate database constraints rules					
Apply basic database constraints					
Plan a DBMS to use					
Create database organization					
Compare types of databases					

Topic 2.1: Designing an Entity Relationship Diagram

Key Competencies:

Knowledge	Skills	Attitudes
1. Describe Entity relationship diagram	1. Construct Entity Relationship Diagram	1.Be decisive
2. Describe database design abstraction levels	2. Differentiate database design abstraction levels	2. Be detail-oriented
3.Describe system metadata	3. Evaluate system metadata	3. Be attentive

Getting Started: What do we know and where are wegoing?



In pair, you are asked to do the following tasks:

Task: 1.what do you think the picture represent?

2. What is the relationship between the picture and the topic?

Activity 1: Problem Solving

A garage has a problem of developing an Entity Relationship Diagram and finalizing the design process of database. This garage does not have a database designer who is familiar with database design tools. In group of four trainees, you are asked to do the following tasks

Task:

1. List all Entity Relationship Diagram symbols and their meaning.

2. Describe database design levels or phases.

Key Facts 2.1

1. INTRODUCTION TO ENTITY RELATIONSHIP DIAGRAM

1.1. Description of Entity Relationship Diagram (ERD)

Entity Relationship Diagram or ERD for short is a type of structural diagram for use in database design.

- Components of an ERD

There are five main components of an entity relationship diagram

I. Entities, which are represented by rectangles.

II. Relationships (Actions), which are represented by diamond shapes, show how two entities share information in the database.

III. Attributes, which are represented by ovals.

IV. Connecting lines, solid lines that connect attributes to show the relationships of entities in the diagram.

V. Cardinality specifies how many instances of an entity relate to one instance of another entity. Ordinarily is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinarily describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinarily specifies the absolute minimum number of relationships.

2. DATABASE DESIGN ABSTRACTION LEVELS

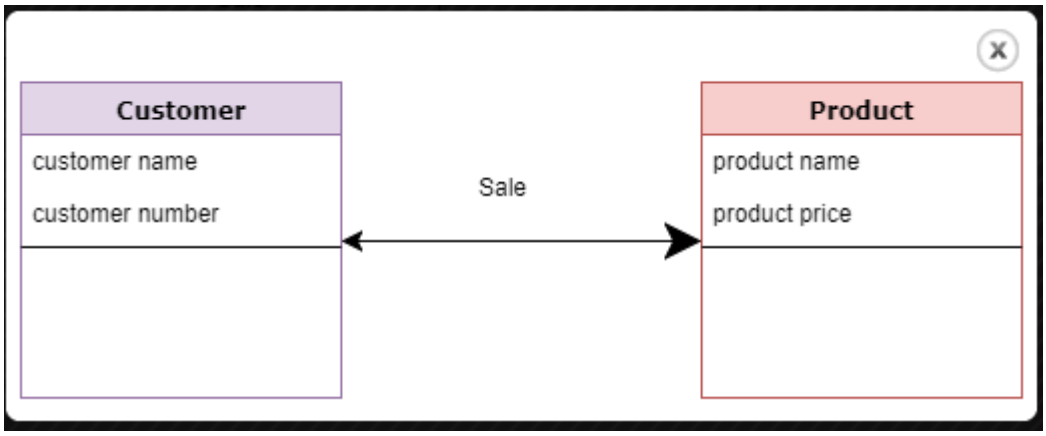
Database design is a process which applies three phases called abstraction levels.

2.1.Conceptual or view model of data

The conceptual model is also known as the data model that can be used to describe the conceptual schema when a database system is implemented. It hides the internal

details of physical storage and targets on describing entities, data type, relationships and constraints.

The main aim of this model is to establish the entities, their attributes, and their relationships. In this Data modeling level, there is hardly any detail available of the actual Database structure.



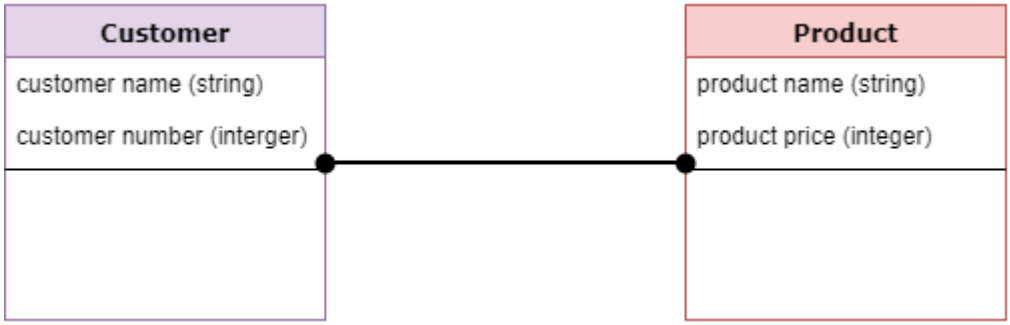
❖ Characteristics of a conceptual data model

- Offers Organization-wide coverage of the business concepts.
- This type of Data Models are designed and developed for a business audience.
- The conceptual model is developed independently of hardware specifications like data storage capacity, location or software specifications like DBMS vendor and technology. The focus is to represent data as a user will see it in the "real world.

  Conceptual data models known as Domain models create a common vocabulary for all stakeholders by establishing basic concepts and scope.

2.2.Logical model of data

Logical data models add further information to the conceptual model elements. It defines the structure of the data elements and set the relationships between them.



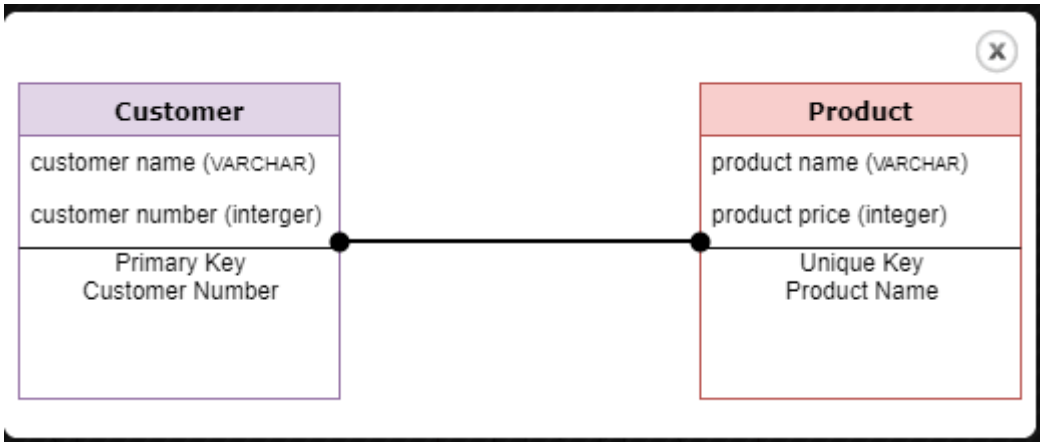
The advantage of the Logical data model is to provide a foundation to form the base for the Physical model.

❖ Characteristics of a Logical data model

- Describes data needs for a single project but could integrate with other logical data models based on the scope of the project.
- Designed and developed independently from the Database Management System
- Data attributes will have data types with exact precisions and length.
- Normalization processes to the model is applied typically till third normal form.

2.3.Physical Data Model

A Physical Data Model describes the database specific implementation of the data model. It offers an abstraction of the database and helps generate schema. This is because of the richness of meta-data offered by a Physical Data Model.



This type of Data model also helps to visualize database structure. It helps to model database columns keys, constraints, indexes, triggers, and other RDBMS features.

❖ **Characteristics of a physical data model:**

- The physical data model describes data need for a single project or application though it may be integrated with other physical data models based on project scope.
- Data Model contains relationships between tables that which addresses cardinality and null ability of the relationships.
- Developed for a specific version of a DBMS, location, data storage or technology to be used in the project.
- Columns should have exact data types, lengths assigned and default values.

- Primary and Foreign keys, views, indexes, access profiles, and authorizations, etc. are defined.

3.SYSTEM METADATA

Metadata is data that describes other data. Metadata summarizes basic information about data, which can make finding and working with particular instances of data easier. For example, author, date created and date modified and file size are examples of very basic document metadata.

There are three main types of metadata:

3.1. Descriptive metadata: describes a resource for purposes such as discovery and identification. It can include elements such as title, abstract, author, and keywords.

3.2. Structural metadata: indicates how compound objects are put together, for example, how pages are ordered to form chapters.

3.3. Administrative metadata: provides information to help manage a resource, such as when and how it was created, file type and other technical information, and who can access it. There are several subsets of administrative data; two that are sometimes listed as separate metadata types are: − Rights management metadata, which deals with intellectual property rights,
and
− Preservation metadata, which contains information needed to archive and preserve a resource.

4.COMPARISON OF DATABASE DESIGN ABSTRACTION LEVELS

The table below compares the different features and concerns of conceptual, logical, and physical abstraction levels:

Feature/Action	Conceptual	Logical	Physical
Entity Names	✓	✓	
Entity Relationships	✓	✓	
Attributes		✓	
Primary Keys		✓	✓
Foreign Keys		✓	✓
Table Names			✓
Column Names			✓
Column Data Types			✓

Activity 2: Guided Practice

Kigali City needs a database to manage special economic zone activities .You are hired as a database designer to build this database. The information you need to gather relates to: Plots, Owners, and Business.

In group of four trainees, you are asked to do the following tasks:

Task:

1. Create a conceptual data model and draw the ERD of the case study.
2. Perform logical data model of the case study.
3. Use physical data model of the case study.

Activity 3: Application

A school XX wants a database to manage their activities .You are hired as a database designer to build this database. The information you need to gather relates to: Students, Courses, and teachers.

every trainee is asked to do the following tasks:

Task:

A. Create a conceptual data model and draw the ERD of the school xx.
B. Perform logical data model of the school xx.
C. Use physical data model of the school xx.

Points to Remember

- Use up-to-date ERD Diagramming conventions.
- Remember to apply business privacy rules
- Database design process starts with conception phase.

Formative Assessment

1. Complete the following sentence
   a. …….. adds further information to the conceptual model elements.
   b. Within ERD, attributes are represented by using……
   c. …… describes a resource for purposes such as discovery and identification.
2. Describe database design abstraction levels
3. After listing types of metadata, write one example for each type.
4. Draw the ERD on Logical model of data for Student and Course

TOPIC 2.2: IDENTIFYING THE CONSTRAINTS

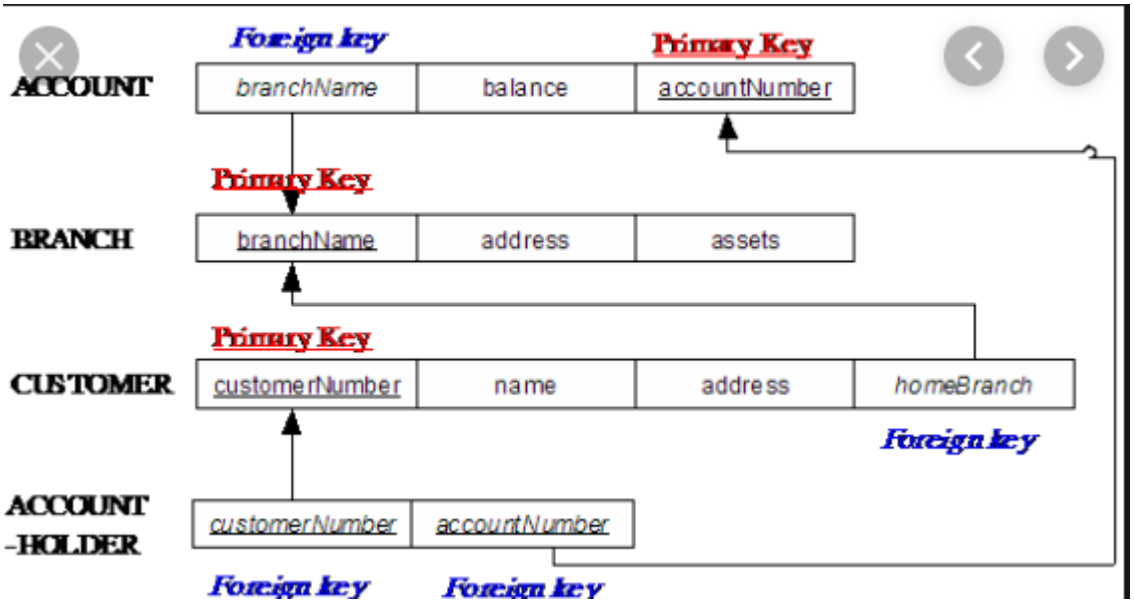
Key Competencies:

Knowledge	Skills	Attitudes
1. Describe data constraints	**1.** Compare data constraints	**1.** Be skillful
2. Describe database constraints rules	**2.** Evaluate database constraints rules	**2.** Be detail-oriented
3. Manage basic database constraints	**3.** Apply basic database constraints	**3.** Be creative

Getting Started: What do we know and where are we going?

Picture brief

The picture below list table names left side in bold. After each table, its description is given and at the top of some fields, labels (Primary key, Foreign key) are used.



In pairs, you are asked to do the following tasks:

Task:

1. What do you think the picture represent?
2. What is the relationship between the picture and the topic?

Activity 1: Problem Solving

A transportation company has a problem for managing their company, by creating database schema, tables PASSENGER (PassengerID,FirstName,Departure,Destination,BusId), BUS(BusId,Number_Of_sits) are related in such way: only one instance of BUS table may relate to one or many instance(s) of PASSENGER table. First name field of PASSENGER and Number_Of_sits field of BUS cannot be empty. You are asked to perform bellow tasks in small group:

Task:

1. Describe what is the constraints
2. Identify common fields in both tables.

Key Facts 2.2

1. DATABASE CONSTRAINTS

Constraints are the rules enforced on the data columns of a table. These are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the database.

1.1. Primary key

PRIMARY Key – Uniquely identifies each row/record in a database table.

Example: StudentID of STUDENTS table. Each StudentID is unique and identifies a student record within the table. Note that primary key value cannot be null. Meanwhile you cannot have a student whose StudentID is empty.

1.2. Foreign key

This constraint helps to map two or more tables in the database. It enforces parent-child relationship in the database. Foreign key in the child table is the column which is a primary key in the parent table.

For example, each employee works for some department. Hence to map employee and department tables, we have to have DEPARMENT_ID of DEPARTMENT table in EMPLOYEE table too. DEPARTMENT_ID is the primary key in DEPARTMENT table (Parent table) and is foreign key in EMPLOYEE table (Child table).

1.3. Unique

This constraint ensures, the column will have unique value for each row. The column value will not repeat for any other rows in the table.

Passport number of individual person is unique. Hence passport column in the PERSON table is made UNIQUE. It avoids duplicate entry of passport number to other persons.

1.4. Check

This constraint is used to check for specific conditions on the column. For example, if age has to be entered between 25 and 32, we can use CHECK Constraint. This will not allow to enter the age<25 and age>32.

1.5. Not Null

This constraint forces the column to have non-null value. We cannot enter/update any NULL value into such columns. It must have valid value all the time. For example, each student in STUDENT table should have class specified. No student can exist without class. Hence class column in the STUDENT table can be made NOT NULL.

1.6.Default

This constraint specifies the default value to be entered when no value is entered to it. Suppose whenever we enter an entry in the STUDENT table, apart from Student details we also have to store the date when it is being entered. This entry would always be SYSDATE. Instead of entering it each time when we do an entry, if we set the default value of this column as SYSDATE, this column will be always inserted with SYSDATE. If we need to override this value with any other date, then we have to explicitly insert the new date.

2. DATABASE CONSTRAINTS RULES

2.1. Entity integrity rule

To ensure *entity integrity*, it is required that every table has a primary key. Neither the primary key nor any part of it can contain null values. This is because null values for the primary key mean we cannot identify some rows. For example, in the EMPLOYEE table, Phone cannot be a primary key since some people may not have a telephone.

2.2. Referential integrity

Referential integrity requires that a foreign key must have a matching primary key or it must be null. This constraint is specified between two tables (parent and child); it maintainsthe correspondence between rows in these tables. It means the reference from a row in one table to another table must be valid.

Examples of referential integrity constraint in the Customer/Order database of the Company:

Customer(<u>CustID</u>, CustName)
Order(<u>OrderID</u>, CustID, OrderDate)

2.3. Domain integrity

Domain restricts the values of attributes in the relation and is a constraint of the relational model. However, there are real-world semantics for data that cannot be specified if used only with domain constraints.

3. MANAGING CONSTRAINTS

a. Table level constraints

Only primary key and foreign key constraints can be either created at table level or column level with CREATE TABLE command as shown below:

CREATE TABLE EMPLOYEE (EMPLOYEE_ID VARCHAR2 (10) PRIMARY KEY,
EMP_NAME VARCHAR2 (50),
DOB DATE,
DEPT_ID NUMBER **REFERENCES DEPARTMENT(DEPARTMENT_ID),**

……);

Column level Fk

OR

CREATE TABLE EMPLOYEE (EMPLOYEE_ID VARCHAR2 (10) PRIMARY KEY,
EMP_NAME VARCHAR2 (50),
DOB DATE,

Table level Fk

DEPT_ID NUMBER
CONSTRAINT fk_DeptId FOREIGN KEY (DEPT_ID) REFERENCES DEPARTMENT (DEPARTMENT_ID));

CREATE TABLE STUDENT (STUDENT_ID NUMBER (10) NOT NULL,
STUDENT_NAME VARCHAR2 (50) NOT NULL,
AGE NUMBER
CONSTRAINT pk_StdID PRIMARY KEY (STUDENT_ID));

Table level Pk

OR

column level Pk

CREATE TABLE STUDENT (STUDENT_ID NUMBER (10) NOT NULL **PRIMARY KEY**,
STUDENT_NAME VARCHAR2 (50) NOT NULL,
AGE NUMBER);

a. Column level constraints

All domain integrity constraints (NOT NULL, UNIQUE, DEFAULT, and CHECK) are defined at column level. For example:

CREATE TABLE STUDENT (STUDENT_ID NUMBER (10) NOT NULL **UNIQUE**
STUDENT_NAME VARCHAR2 (50) NOT NULL,
AGE NUMBER);

b. Dropping constraints

Any constraint that you have defined can be dropped using the ALTER TABLE command with DROP CONSTRAINT option. For example, to drop the primary key constraint in the employees table, you can use the following command:

ALTER TABLE EMPLOYEES

DROP CONSTRAINT PRIMARY KEY;

Activity 2: Guided Practice

A transportation company has a database schema, PASSENGER (PassengerID,FirstName,Departure,Destination,BusId), BUS(BusId,Number_Of_sits) are not related and there is no limitations on fields data, but specifications given by the management team are: only one instance of BUS table may relate to one or many instance(s) of PASSENGER table. First name field of PASSENGER and Number_Of_sits field of BUS cannot be empty. Together with the trainer, you are asked to do the following tasks in pair:

Task:

1. Identify primary key(s) and foreign key(s) fields for the senario above.
2. List mandatory (NOT NULL) column(s) for each table.

Activity 3: Application

A transportation company database schema, tables PASSENGER (PassengerID,FirstName,Departure,Destination,BusId), BUS(BusId,Number_Of_sits) DRIVER(DriveID,FirstName,LastName, PhoneNumber, Licence) are not related and there is no limitations on fields data, but specifications given by the management team are: only one instance of BUS table may relate to one or many instance(s) of PASSENGER table, and one instance of DRIVER table may relate to only one instance of BUS table. First name field of PASSENGER and Number_Of_sits field of BUS cannot be empty,PhoneNumber field of DRIVER is unique. Individually, you are asked to do the following tasks in small group:

Task: 1. Fulfill the following table:

Table	Column/Fiel	Constraint set
PASSENGER	PassengerID	
	FirstName	
BUS	BusID	
	DriverID	
	Number_Of_Sits	
DRIVER	DriverID	
	PhoneNumber	

2. Identify primary key(s) and foreign key(s) fields for the scenario above.

Points to Remember

- Always consider users specifications and rules
- Relational database keys
- Modification of constraints based on business rules

Formative Assessment

1. Describe the following constraints:

i. Primary key
ii. Foreign key
iii. Unique
iv. Not null
v. Check

ANSWER i

2. Match a vocabulary with an appropriate definition of the table below:

Vocabulary Definition

Referential integrity	Specifies the default value to be entered when no value is entered to it.
Default	restricts the values of attributes in the relation and is a constraint of the relational model
Domain integrity	requires that a foreign key must have a matching primary key or it must be null

3. Contrast the following:

a. NOT NULL vs UNIQUE
b. Primary key vs Check

TOPIC 2.3: DEVELOPING A DATA DICTIONARY

Key Competencies:

Knowledge	Skills	Attitudes
1. Describe DBMS	1. Plan a DBMS to use	1. Be detail-oriented
2. Describe database organization	2. Create database organization	2. Be creative
3. Describe types of databases	3. Compare types of databases	3. Be skillful

Getting Started: What do we know and where are we going?

Field Name	Data type	Field Length	Constraint	Description
Client_id	Int	10	Primary key	Client id, Auto generated
Client_name	Varchar	20	Not null	Name of client
Password	Varchar2	30	Not null	Login Password for client
Contact_no	Int	15	Not null	Landline or mobile number
Email_id	Varchar2	30	Not null	Any email id
Max_Users	Int	10	Not null	Maximum number of users
Current_users	Int	10	Not null	Currently present user

In pair, you are asked to do the following tasks:

Task:

1. What the table above means to you?

2. What is the relationship between the table above and the topic?

Activity 1: Problem Solving

A manufacturing industry is facing a problem of their data control, they decided to use a DBMS, where they will start by creating a data dictionary.

In group of four trainees, you are asked to do the following tasks :

 **Task:**

1. What is DBMS and write it in full words.
2. Describe what the data dictionary is.

Key Facts 2.3

1. DATABASE MANAGEMENT SYSTEMS

1.1.Definition and goal of DBMS

- Database Management System (DBMS): is a software package designed to define, manipulate, retrieve and manage data in a database.
- Primary goal of database management system is to provide a way to store and retrieve database information that is convenient and efficient.

1.2.Objective of DBMSs

Various objectives of database management system include:
- Data availability: in terms of databases, availability means that if a database is available, the users get its data.
- Data integrity: is the maintenance of, and the assurance of the accuracy and consistency of data.
- Data security: refers to protective digital privacy measures that are applied to prevent unauthorized access to computers, databases and websites. Data security also protects data from corruption.
- Data independence: it refers to the immunity of user applications to changes made in the definition and organization of data.
- Data backup and recovery: **Sometimes database failure occurs so there is no option like one can say that all the data has been lost. There should be a backup of database so that on database failure it can be recovered. DBMS has the ability to backup and recover all the data in database**.
- Multiple users' access: **no one handles the whole database alone. There are lots of users who are able to access database. So this situation may happen that two or more users are accessing database. They can change whatever they want, at that time DBMS makes it sure that they can work concurrently.**
- Removes duplicity: **if you have lots of data then data duplicity will occur for sure at any instance. DBMS guarantee it that there will be no data duplicity among all the records.**
- Mass storage: **DBMS can store a lot of data in it. So for all the big firms, DBMS is really ideal technology to use. It can store thousands of records in it and one can fetch all that data whenever it is needed.**
- Platform Independent: **One can run DBMS at any platform. No particular platform is required to work on database management system.**

3.4. Disadvantages of DBMS

- DBMS offers a variety of techniques to store & retrieve data
- DBMS serves as an efficient handler to balance the needs of multiple applications using the same data.
- Uniform administration procedures for data
- Application programmers never exposed to details of data representation and storage.
- A DBMS uses various powerful functions to store and retrieve data efficiently.
- Offers Data Integrity and Security
- The DBMS implies integrity constraints to get a high level of protection against prohibited access to data.
- A DBMS schedules concurrent access to the data in such a manner that only one user can access the same data at a time
- Reduced Application Development Time.

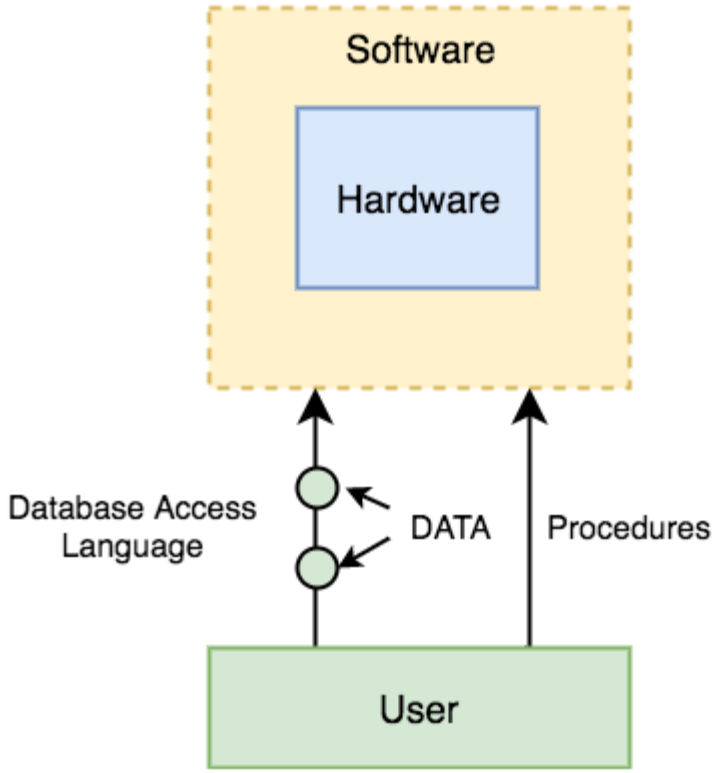
3.5. Disadvantages of DBMSs

DBMS may offer plenty of advantages but, it has certain issues:

- Cost of Hardware and Software of a DBMS is quite high which increases the budget of your organization.
- Most database management systems are often complex systems, so the training for users to use the DBMS is required.
- In some organizations, all data is integrated into a single database which can be damaged because of electric failure or database is corrupted on the storage media
- Use of the same program at a time by many users sometimes leads to the loss of some data.
- DBMS can't perform sophisticated calculations.

3.6. Components of DBMSs

The database management system can be divided into five major components, which are: Hardware, Software, Data, Procedures, Database Access Language as illustrated the figure below:



3.7. Applications of DBMSs

3.7.1. Railway Reservation System: Database is required to keep record of ticket booking, train's departure and arrival status. Also if trains get late then people get to know it through database update.

3.7.2. Library Management System: **There are thousands of books in the library so it is very difficult to keep record of all the books in a copy or register. So DBMS used to maintain all the information relate to book issue dates, name of the book, author and availability of the book.**

3.7.3. Banking: We make thousands of transactions through banks daily and we can do this without going to the bank. So how banking has become so easy that by sitting at home we can send or get money through banks. That is all possible just because of DBMS that manages all the bank transactions.

3.7.4. **Universities and colleges:** Examinations are done online today and universities and colleges maintain all these records through DBMS. Student's registrations details, results, courses and grades all the information are stored in database.

3.7.5. Credit card transactions: For purchase of credit cards and all the other transactions are made possible only by DBMS. A credit card holder knows the importance of their information that all are secured through DBMS.

3.8. DBMS software

Depending on features supported by DBMS software, you may face different variations of DBMS such as RDBMS (Relational Database Management System), DRDBMS (Distributed Relational Database Management System) and so on and so on. There are a number of DBMS software on the market, but here is a list of popular ones: MySQL, Microsoft SQL Server, MariaDB, Oracle, SQLite, PostgreSQL, etc.

4. Database organization

Database organization refers to the layout of the database itself and data it contains.

4.1. Relational database

A relational database is one which employs the relational model, in which the raw data is organized into sets of tuples, and the tuples organized into relations.

4.2. Structured Query Language (SQL)

SQL is a relational database query and manipulation language. Its power and flexibility allows for the creation of databases and tables, and the manipulation and query of data.

4.3. NoSQL

NoSQL is used to mean "not only SQL," meaning that these solutions are more flexible and less rigid in nature.

5. DESCRIPTION OF DATA DICTIONARY

5.1. Definition

A data dictionary is a collection of descriptions of the data objects or items in a data model for the benefit of programmers and others who need to refer to them. A first step in analyzing a system of objects with which users interact is to identify each object and its relationship to other objects.

5.2. Data dictionary template

Field Name	Data Type	Field Size for display	Description	Example
Employee Number	Integer	10	Unique ID of each employee	1645000001
Name	Text	20	Name of the employee	David Heston
Date of Birth	Date/Time	10	DOB of Employee	08/03/1995
Phone Number	Integer	10	Phone number of employee	6583648648

This is a data dictionary describing a table that contains employee details.

6. Types of databases

The classification of databases obviously depends on some parameters like size and business usage. By size, here we have the following databases:

- Flat-file database or text database. It stores data in a plain text file, with each line of text typically holding one record. A flat file database uses a single structure and unlike a relational database cannot contain multiple tables and relations.

- Relational database. It contains multiple tables of data with rows and columns that relate to each other through special key field.

Activity 2: Guided Practice

Akagera wants to implement a database to keep the information of car stored at Muhanga branch. this is how the table has to be filled :

- CARS(carID, Mark, Mfg_Date, Origin)
- EMPLOYEES(EmpID,FName,LName, Gender, Phone,Email).

Conditions to be followed when implementing this database:

- The origin of a car can be known or not, and is 40 character length, other car data must be present.
- Email of employee is optional and can be 15 character length.
- The employee's phone, EmpID is unique.
- First name, Last name of employee are mandatory and 30 character length.

Guided by the trainer, you are asked to do the following tasks:

Task:

1. Develop a data dictionary for Akagera at Muhanga branch
2. Describe the possible software which can be used for implementing akagera store at Muhanga branch.

Activity 3: Application

Akagera wants to implement a database to keep the information of car stored at Muhanga branch. this is how the table has to be filled :

- CARS(carID, Mark, Mfg_Date, Origin)
- EMPLOYEES(EmpID,FName,LName, Gender, Phone,Email).

Conditions to be followed when implementing this database:

- The origin of a car can be known or not, and is 40 character length, other car data must be present.
- Email of employee is optional and can be 15 character length.
- The employee's phone, EmpID is unique.
- First name, Last name of employee are mandatory and 30 character length.

Task:

1. Develop a data dictionary for Akagera at Muhanga branch
2. With your analysis, what attributes can be added to the table given in the scenario

Points to Remember

- Before buying a DBMS, check its cost
- Put only necessary object data into a data dictionary to help developers
- Always use up-to-date DBMS version

Formative Assessment

1. Write DBMS in full words and list its components.
2. Explain what meant by relational database.
3. Describe the template of a data dictionary.

Unit 3: CREATE DATABASE

Illustration of Learning Unit

Picture Brief

The illustration above is made by a blue cylinder in middle which represents the database, surrounded by DBMS (Database Management Systems), logos and their names (Cassandra, TITAN, couch DB, SQLite, mongoDB, ORACLE, HBASE, MySQL, SQL Server) as tools to create database.

Topics

1.1. Create tables and attributes

1.2. Create views

1.3. Manage access control of database

Unit Summary:This unit describes knowledge, skills, and attitudes required to create database objects such as tables and views, and to control database access.

Self-Assessment: Unit 3

3. Look at the illustration. What is happening? What do you think this unit will be about? What topics might be covered?

4. Fill in the self assessment below.

There are no rights or wrong ways to answer this survey. It is for your own use during this course. Think about yourself: do you think you can do this? How well? Read the statements across the top. Put a check in column that best represents your situation. At the end of this unit, we'll take this survey again.

My experience Knowledge, skills and attitudes	I don't have any experience doing this.	I know a little about this.	I have some experience doing this.	I have a lot of experience with this.	I am confident in my ability to do this.
Select DBMS tools					
CREATE TABLE command					
Apply data type on table field					
Modify existing table					
Use views					
List types of views					
Modify views					
Identify types of database users					
Apply user authentication and authorization					
Use various access methods					

Topic 3.1: Create tables and attributes

Key Competencies:

Knowledge	Skills	Attitudes
4. List DBMS	**4.** Select DBMS to use	**4.** Be decisive
5. Describe database tables	**5.** Create database tables	**5.** Be creative
6. Name tables' fields data types	**6.** Apply tables' fields data types	**6.** Be skillful
7. Indicate database tables modification	**7.** Modify database tables	**7.** Be skillful

GettingStarted: What do we know and where are wegoing?

```
-----------Persons (
PersonID int,
LastName varchar(255),
FirstName varchar(255),
Address varchar(255),
City varchar(255)
);
```

In pair, you are asked to do the following tasks:

Task:1. Describe what does the codes above means and what do you think is missing?

2. What is the relationship between the codes above and the topic?

Activity 1: ProblemSolving

A pharmacy hasa problem of its management then decide to use DBMS and database language to implement a data dictionary of its medicine store. In group of four trainees, you are asked to do the following tasks:

Task: 1. Describe a DBMS to use for managing the medicine store.

2.Describe SQL languages which can be used to resolve the issue of medicine store

Key Facts 3.1

1. DBMS TOOLS

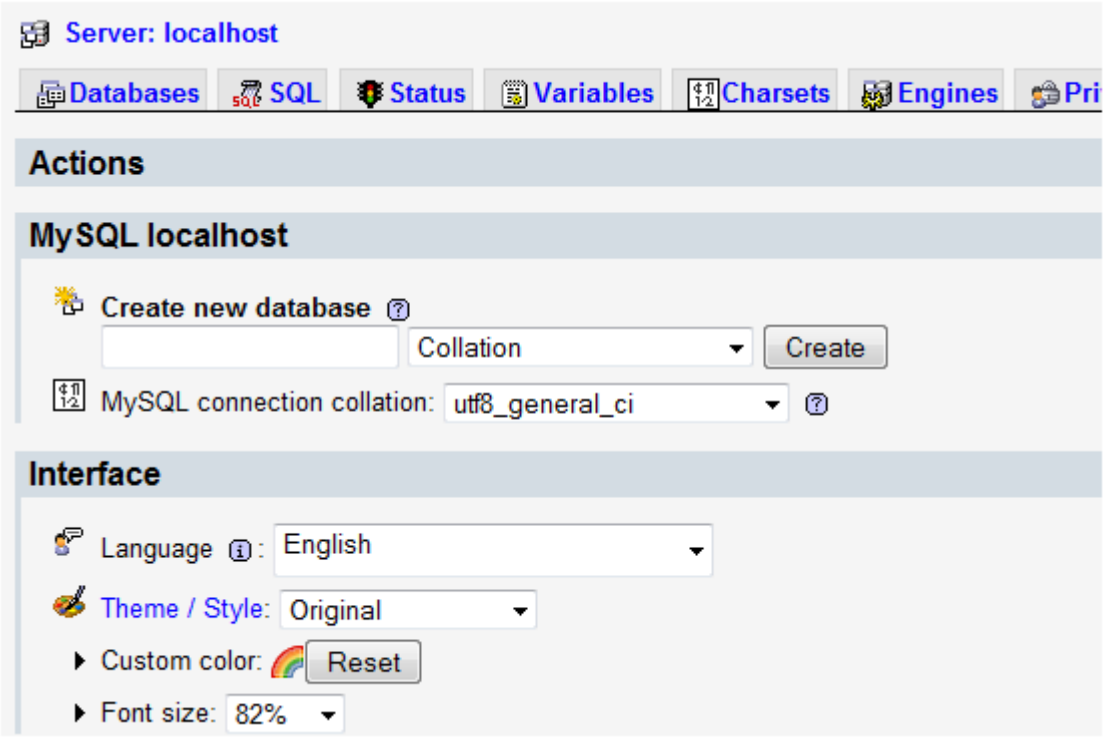
A DBMS tools are frequently used interchangeably with “DBMS APIs”, they are application software that provide a convenient, user-friendly GUI for working with a DBMS. they differ from their purposes, some are web-applications, others are desktop applications, etc. or they differ from their features. Here we talk a little bit on MySQL popular tools.

1.1. Types of DBMS TOOLS

1.1.1. MySQL tools

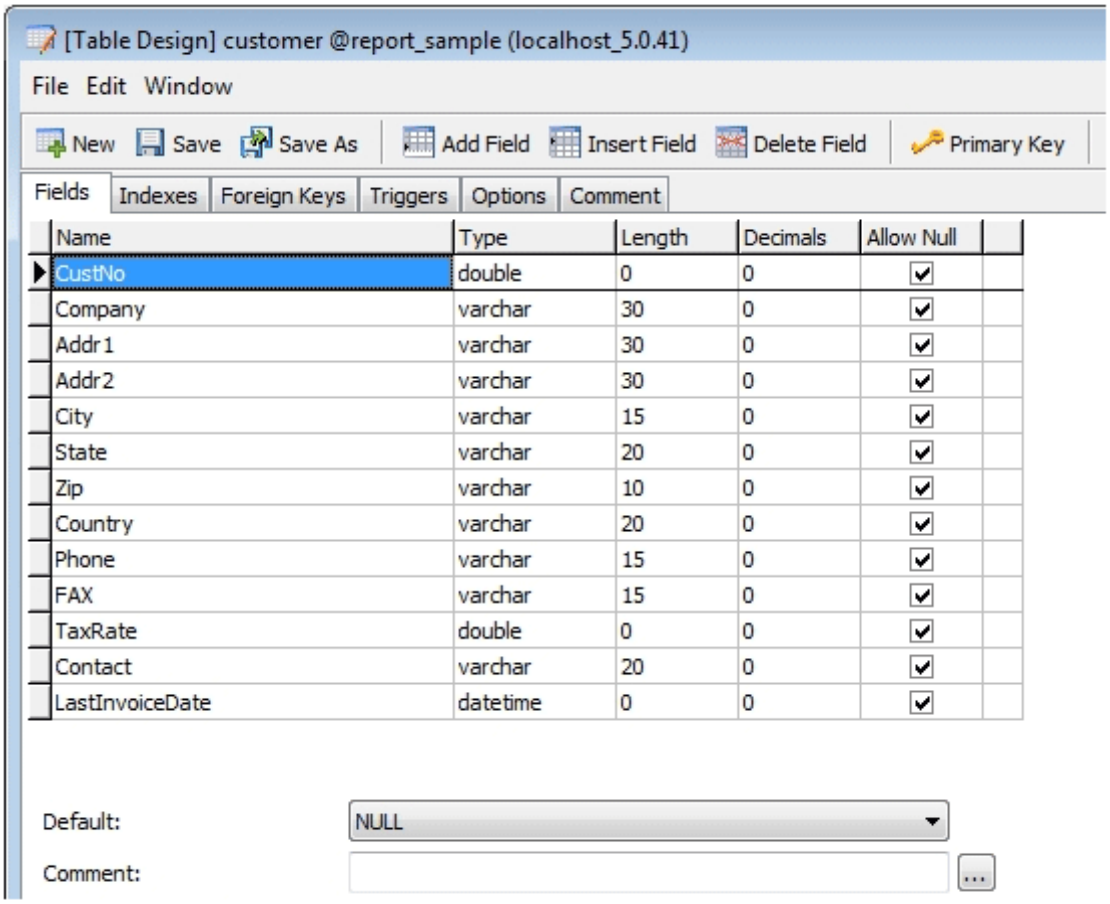
1.1.1.1. phpMyAdmin

phpMyAdmin is the most popular open source database management tool. It is a web-based MySQL manager. you can create/manage databases, tables, database users, optimize and check for errors in your database tables, run queries, export/import data in plenty of file formats, and much more.



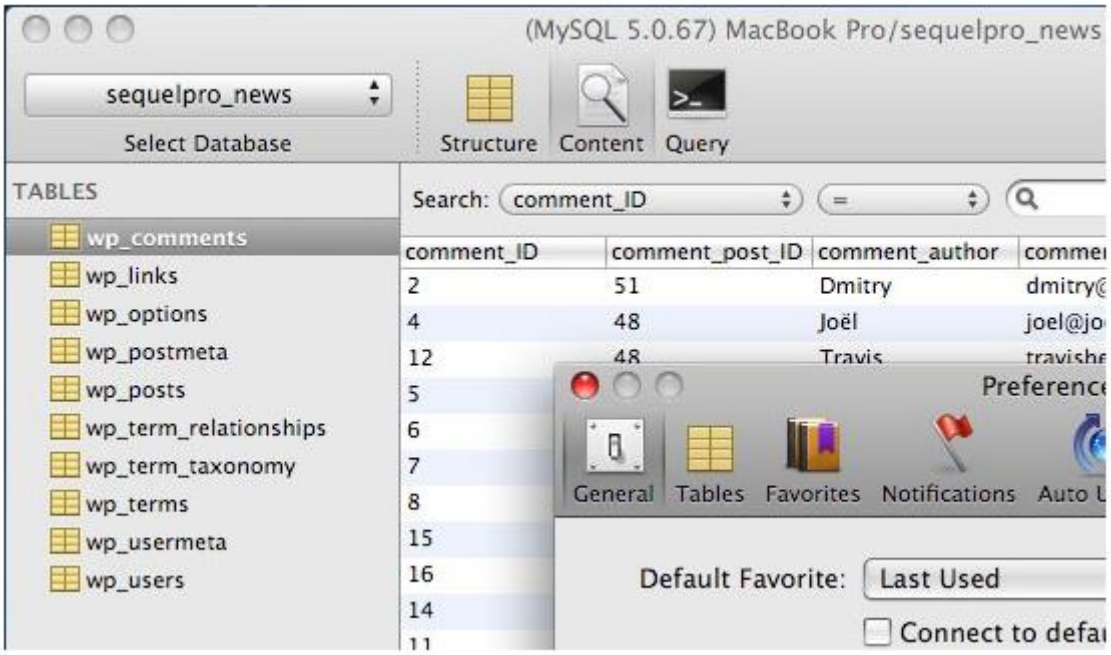
1.1.1. 2. Navicat

Navicat for MySQL is part of the family of cross-platform (Windows/Mac/Linux) relational database management tools created by PremiumSoft. It has a visual query builder for creating complex database queries in a snap, import/export/backup tools for your databases, a report builder for printing out db queries that the boss can understand, a task scheduling assistant for scheduled execution of jobs, and much more. Navicat also produces front-ends for Oracle and PostgreSQL. Navicat products have a **Lite** version (non-commercial) and a **Premium** version. The following picture illustrates Navicat interface:



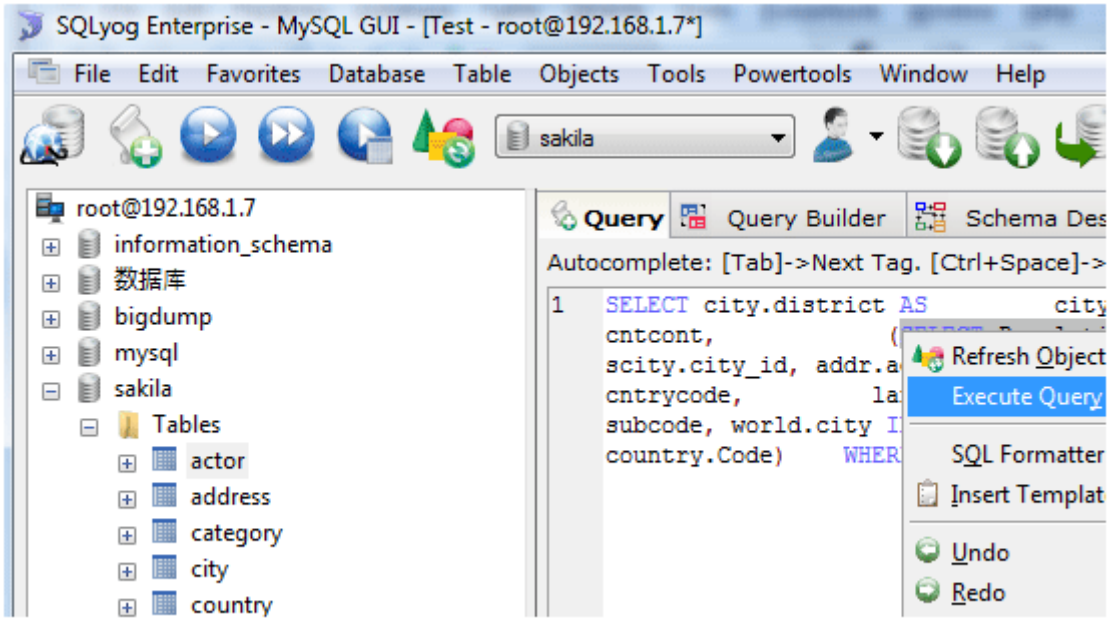
1.1.1.3. SequelPro

For Mac OS users, *Sequel Pro*isan open source MySQL database management front-end tool. It has a slick and intuitive GUI that keeps it barebones so that you can easily navigate the application.



1.1.1.4. SQLyog

SQLyog is a GUI administration tool for MySQL developed by Indian-based software company, Webyog. It comes in two flavors: **Enterprise** (commercial) and **Free Community** version.



1.2.CREATING TABLES

1.2.1. SQL Data Definition Language

Data Definition Language (DDL). It is a language used for defining and modifying the data and its structure. It is used to build and modify the structure of your tables and other objects in the database. To do it, DDL involves a number of commands to write SQL structures.

i. CREATE TABLE command

CREATE TABLE command is used to create a database table. The following is the SQL syntax of creating table:

```
CREATE TABLE table_name (
   column1 datatype,
   column2 datatype,
   column3 datatype,
  ....
);
```

The column parameters specify the names of the columns of the table.

The datatype parameter specifies the type of data the column can hold (e.g. varchar, integer, date, etc.).

Remember that at column level, constraints can also be defined as seen from previous topics.

Example of table creation:

```
CREATE TABLE Persons (
   PersonIDint PRIMARY KEY,
   LastNamevarchar(50),
   FirstNamevarchar(50),
   Address varchar(30),
   City varchar(25)
);
```

1.2.2. MODIFYING EXISTING TABLES

Sometimes, you may need to add a new column to a table, delete one or more table columns, change the data type of a column, rename your table, or remove the table from the database. This is also achieved by applying DDL commands as listed below:

1.2.3. ALTER TABLE command

The SQL ALTER TABLE command is used to add, delete or modify columns in an existing table. You should also use the ALTER TABLE command to add and drop various constraints on an existing table.

a. Adding a new column to an existing table

The syntax to add a new column is the following:

```
ALTER TABLE table_name ADD column_namedatatype;
```

b. Deleting a column from existing table

The basic syntax of an ALTER TABLE command to DROP COLUMN in an existing table is as follows:

```
ALTER TABLE table_name DROP COLUMN column_name;
```

c. Changing column data type

The basic syntax of an ALTER TABLE command to change the DATA TYPE of a column in a table is as follows:

```
ALTER TABLE table_name MODIFY COLUMN column_namedatatype;
```

d. Adding constraints on existing column(s)

These are different syntax used to add various constraints on columns:

- ALTER TABLE table_name MODIFY column_namedatatype NOT NULL;
- ALTER TABLE table_name
ADD CONSTRAINT MyUniqueConstraintUNIQUE(column1, column2...);
- ALTER TABLE table_name
ADD CONSTRAINT MyUniqueConstraint CHECK (CONDITION);

- ALTER TABLE table_name

ADD CONSTRAINT MyPrimaryKey PRIMARY KEY (column1, column2...);

You can also remove existing constraint such as:

```
ALTER TABLE table_name
DROP PRIMARY KEY;
```

1.2.4. Removing/Dropping a table

The drop table command is used to delete a table and all rows in the table. Its syntax is the following:

```
DROP TABLE table_name;
```

Example: DROP TABLE Persons;

1.2.5. Renaming a database table

RENAME TABLE command is used to rename a database table or any other object. Its syntax is the following:

RENAME TABLE <old_name>TO <new_name>

RENAME TABLE Clients TO Customers

1.2.6. Truncating database table

TRUNCATE TABLE command is used to delete all the rows from the table permanently.

It removes all the records from a table, including all spaces allocated for the records.

Its syntax is the following:

TRUNCATE TABLE <table_name>.

For example , TRUNCATE TABLE Employees;

Activity 2: Guided Practice

A pharmacy want to implement the following **data dictionaryshowingmedicine table details.**

In pair, guided by the trainer, you are asked to help the pharmacy by doingrelatedtasks.

Field Name	Data type	Field size	constraint	Description
MedID	character	15	Primary key	Medicine identification
MedName	Character	25	NOT NULL	Name of medicine
MfgDate	Date		NOT NULL	Manufacturer date
ExpDate	Date		NOT NULL	Expiration date
UnitPrice	number	10		Unit price

 **Task:**

1. Select a DBMS tool to use.
2. Write SQL code to create medicine table.
3. Use the ALTER TABLE command to add Quantity column of numbers, of 10 numbers size and always required.

Activity 3: Application

A pharmacy want to implement the following **data dictionaryshowingmedicine table details.individualy, you are asked to help the pharmacy by doingrelatedtasks.**

Field Name	Data type	Field size	constraint	Description
MedID	character	15	Primary key	Medicine identification
MedName	Character	25	NOT NULL	Name of medicine
MfgDate	Date		NOT NULL	Manufacturer date
ExpDate	Date		NOT NULL	Expiration date
UnitPrice	number	10		Unit price

 **Task:**

1. Write SQL code to create medicine table.

2. Rename the table to **Drugs.**

3. Use the ALTER TABLE command to add Quantity column of numbers, of 10 numbers size and always required.

Points to Remember

- Always use SQL standard version
- Read DBMS tool installation requirements
- Differentiation of truncate and delete commands

Formative Assessment

4. The table below shows tourists' details. You are asked to do related tasks.

Field name	Data type	size	constraint	Description
TouristID	Number/integer		Primary key	Tourist identification
FName	Variable character	30	Not Null	Tourist's first name
LName	Variable character	30		Tourist's last name
Nationality	Variable character	30	Not Null	Tourist's nationality
PhoneNumber	Number/integer			Tourist's contact number
Email	Variable character	25		Any tourist's email

A. Write SQL structure to create Tourists table.
B. Write SQL structure to modify the data type of PhoneNumber column into variable character of 12 character length.
C. Write SQL structure to remove the table from the database.

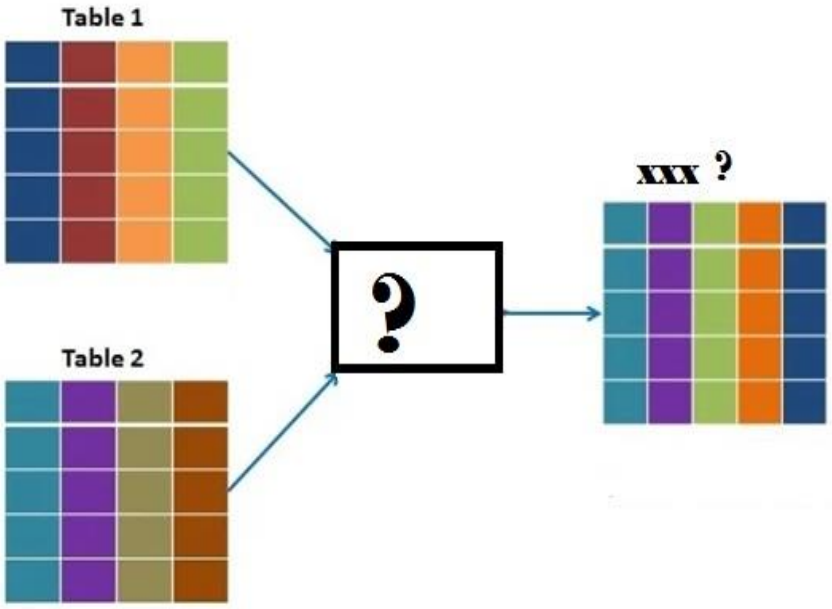
5. Describe what means Data Definition Language (DDL)?

Topic 3.2: Create views

Key Competencies:

Knowledge	Skills	Attitudes
1.Describe views	**4.** Create views	1.Be creative
2. List types of views	2. compare types of views	2. Detail-oriented
3. Describe how to modify	3.Modify views	3. Be skilful

GettingStarted: What do we know and where are wegoing?



Task:

1. Looking to the picture above, what represents to you the square which is inside with a question mark, and what will be the results on last square with xxx ?

2. What is the relationship between the picture above and the topic?

Activity 1: ProblemSolving

A company XYZ, which has employees record presentation, employees(EmployeeID, Ename, DeptID, Salary). It has an issue of retrieving data of employee who use DeptID No. 2, below is the list of their employee records.

In group of four people you are asked to respond to the question asked below.

EmployeeID	Ename	DeptID	Salary
1001	John	2	4000
1002	Anna	1	3500
1003	James	1	2500
1004	David	2	5000
1005	Mark	2	3000
1006	Steve	3	4500
1007	Alice	3	3500

Task:

1. Tell us how you can help the company XYZ, getting the solution they want?

2. Write a query to create a view which will give us EmployeeID, Ename,DeptID, Salary, of DeptIDof Mr. paul?

Key Facts 3.2

1. Creation of Views

1.1. Database View: refers to the result set of a query on the data stored in database table(s).
1.2. Queries: refers to the request for data or information from a database table or combination of tables.
1.3. Website form: is a window or screen that contains numerous fields, or spaces to enter data.
1.4. Syntax to create a view:

```
create or replace
view view_name
```

```
    as
select column_name1, column_name2,...
    from table_name
    where condition;
```

Example1:

Suppose, we have to create a student view table of view10.

create

view view10

select marks from student

whererollno = 10;

Example2: The following SQL creates a view that shows all customers from Brazil:

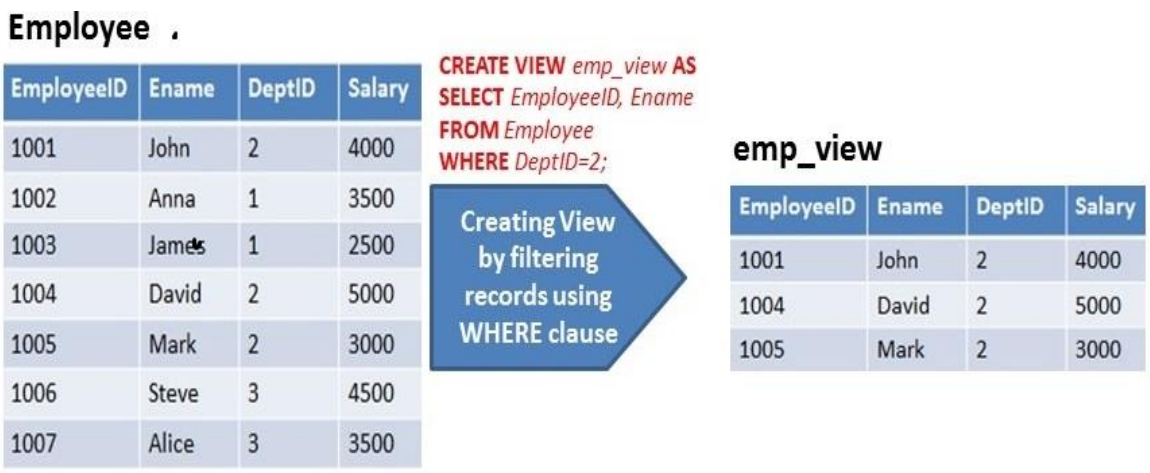
```
CREATEVIEW [Brazil Customers] AS
SELECTCustomerName, ContactName
FROM Customers
WHERE Country = "Brazil";
```

2. Types of Views

- Simple View: A view based on only a single table, which doesn't contain GROUP BY clause and any functions.
- Complex View: A view based on multiple tables, which contain GROUP BY clause and functions.
- Inline View: A view based on a subquery in FROM Clause, that subquery creates a temporary table and simplifies the complex query.
- Materialized View: A view that stores the definition as well as data. It creates replicas of data by storing it physically.

View	Description
Simple View	A view based on the only a single table, which doesn't contain GROUP BY clause and any functions.
Complex View	A view based on multiple tables, which contain GROUP BY clause and functions
Inline view	A view based on a subquery in FROM Clause, that subquery creates a temporary table and simplifies the complex query.
Materialized view	A view that stores the definition as well as data. It creates replicas of data by storing it physically.

2.1. Sample of Simple view



2.2. Sample of Complex View

Features	Simple Views	Complex Views
No. of tables	One	One or More
Containing Functions	No	Yes
Contain Group of data	No	Yes
DML through view	Yes	Not Allowed
Features	Simple Views	Complex Views

2.3. Sample of Inline View

```
SELECT column_names, ...
FROM (subquery)
WHERE ROWNUM<= N;
```

2.4. Sample of MaterializedView

ComparisonBetweenView and MaterializedView

View	Materialized View
View is a logical structure of the table which will be used to retrieve data from one or more table.	Materialized views are also logical structure but data is physically stored in database.
Data access is slower compared to materialized views	Data access is faster compared to simpler view because data is directly accessed from physical location
Views are generally used to restrict data from database	Materialized Views are used in Data Warehousing.

3. Modify views

3.1. Creating and replacing view

View can be created and replaced using CREATE VIEW and REPLACE VIEW.

```
CREATE OR REPLACE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

3.2. View can be deleted using DROP VIEW command.

```
DROP VIEW view_name;
```

Activity 2: Guided Practice

EmployeeID	Ename	DeptID	Salary
1001	John	2	4000
1002	Anna	1	3500
1003	James	1	2500
1004	David	2	5000
1005	Mark	2	3000
1006	Steve	3	4500
1007	Alice	3	3500

The figure above describes the information of employees' company; in group of four people guided by the trainer perform bellow tasks.

 **Task:**

1. **Write a query to create a view which will give us EmployeeID, Ename,DeptID, Salary, of DeptID which is equal to 1?**
2. **Write a query to create a view which will display the information of Employee Mark?**
3. **Write a query to create a view which will give us employees whose salary is less than 4,000?**

Activity 3: Application

salesman_id	name	District	commission
5001	James Hoog	Ruhango	20,000
5002	Nail Knite	Rulindo	13,000
5005	Pit Alex	Nyanza	25,000
5006	Mc Lyon	Huye	40,000
5007	Paul Adam	Musanze	11,000
5003	Lauson Hen	Gasabo	13,000

The figure above, shows the information of a sales man, everyone is asked to perform bellow tasks.

Task:

1. Write a query to create a view for those salesmen belongs to the District of Huye?
2. Write a query to create a view for all salesmen with columns salesman_id, name, and District?
3. Write a query to create a view for salesmen whose commission is above 13,000?

Points to Remember

- One syntax of views can help to obtain a complex report needed
- Views can be changed
- After each view, remember to put semicolon

Formative Assessment

4. Demonstrate the syntax of creating a view?

5.

EmployeeID	Ename	DeptID	Salary
1001	John	2	4000
1002	Anna	1	3500
1003	James	1	2500
1004	David	2	5000
1005	Mark	2	3000
1006	Steve	3	4500
1007	Alice	3	3500

Using the figure above, perform bellow tasks:

I. Write a query to create a view which will display the columns (EmployeeID, Ename and Salary)?

II. Write a query to create a view which will display all information of our employees?

Topic 3.3: Proper management of access control of database

Key Competencies:

Knowledge	Skills	Attitudes
1. List types of database users	1. Identify types of database users	1. Detail-oriented
2. Describe user authentication and authorization	2. Apply user authentication and authorization	2. Be creative
3. List Various access methods	3. Use various access methods	3. Be skillful

GettingStarted: What do we know and where are wegoing?

A bank xxx has employees where everyone uses the system of the bank, which means that each one has a username and password for login to the system, for the security purpose.

In pair, you are asked to do the following tasks:

Task:

1. Tell what the scenario means to you, and professionally what is the term to be used here?

2. What is the relationship between the scenario and the topic?

Activity 1: ProblemSolving

XYZ Company has a database used in their organization, and they are facing a problem restructuring database rights.

In group of four trainees, you are asked todo the following tasks:

Task:

1. List database users who can access the database?
2. What security mesures will you take, when users try to access the database's organization?
3. According to you who do you think will be managing the database of the company?

Key Facts 3.3

1. Types of database users

   This differentiation is made according to the interaction of users to the database. Database system is made to store information and provide an environment for retrieving information. There are four types of database users in DBMS we are going to discuss in this article:

   1.1. Application Programmers

As its name shows, application programmers are the one who writes application programs that uses the database. These application programs are written in programming languages like COBOL or PL (Programming Language 1), Java and fourth generation language. These programs meet the user requirement and made according to user requirements. Retrieving information, creating new information and changing existing information is done by these application programs.

   1.2. End Users

End users are those who access the database from the terminal end. They use the developed applications and they don't have any knowledge about the design and working of database.

There are basically two types of end users that are discussed below:

- **Casual User:** These users have great knowledge of query language. Casual users access data by entering different queries from the terminal end. They do not write programs but they can interact with the system by writing queries.

- **Naive:** Any user who does not have any knowledge about database can be in this category. There task is to just use the developed application and get the desired

results. For example: Clerical staff in any bank is a naïve user. They don't have any dbms knowledge but they still use the database and perform their given task.

1.3. BDA(Database Administrator)

DBA can be a single person or it can be a group of person. Database Administrator is responsible for everything that is related to database. He makes the policies, strategies and provides technical supports.

1.4. System Analyst

System analyst is responsible for the design, structure and properties of database. All the requirements of the end users are handled by system analyst. Feasibility, economic and technical aspects of DBMS is the main concern of system analyst.

2. User authentication and authorization

- Authentication:Authentication is the act of validating that users are who they claim to be. Passwords are the most common authentication factor—if a user enters the correct password, the system assumes the identity is valid and grants access.
- Authorization: Authorization in system security is the process of giving the user permission to access a specific resource or function.

2.1. Security issues

Understanding Security Authentication. To put it simply, authentication is the process that confirms a user's identity. ... The user enters their username, which allows the system to confirm their identity; this system relies on the fact that (hopefully) only the user and the site's server know the password.

2.2. Preventions from unauthorized user's access

Recently, systems are rapidly becoming more advanced and complex. For this reason, security risks to database systemscontinue to grow, necessitatingthe implementation of preventive controls. In this Section, we shall describe security controls that prevent unauthorized access efficiently, and at the same time minimizing the impact to authorized access

2.2.1. Initial Configuration

DBMS products that have not been updated, or running with unnecessary functionalities may possess multiple vulnerabilities. Furthermore, use of default ports or disabling passwords for network access increases the possibility of unauthorized access.

Installation precautions:

- Using the Latest Version
- Installing only the Minimum Functionality
- Changing Port Settings
- Restrict Network Access

2.2.2. Authentication Current

database authentication systems use password authentication during logins. To prevent unauthorized users from using authorized accounts to leak or modify information, account information must be tightly managed. Furthermore, as DBMS administrator accounts enable users to do everything, their account information must be managed more tightly than ordinary user accounts.

❖ Account management:

- Creating Necessary Accounts
- Deleting Unnecessary Accounts
- Locking Unused User Accounts
- Locking Account due to Failed Login

❖ Password management:

- Password Complexity
- Periodically Changing Password
- Setting Password Expiration Date

3. Various access methods

3.1. Permanent objects and access methods

There are three basic types of access methods used to manipulate the permanent and temporary database objects -- Create, Scan, and Probe.

3.2. Temporary objects and access methods

Temporary objects are created by the optimizer in order to process a query. In general, these temporary objects are internal objects and cannot be accessed by a user.

3.3. Objects processed in parallel

The DB2 Symmetric multiprocessing feature provides the optimizer with additional methods for retrieving data that include parallel processing. Symmetrical multiprocessing is a form of parallelism achieved on a single system where multiple CPU and I/O processors sharing memory and disk work simultaneously toward a single result.

3.4. Spreading data automatically

Db2® for i automatically spreads the data across the disk devices available in the auxiliary storage pool (ASP) where the data is allocated. This process ensures that the data is spread without user intervention.

Activity 2: Guided Practice

A bank XYZ has a software which manages all their clients, wherethey have a databasewhichcontains informations, of the clients.

Guided by trainer, you are asked to do the followingtasks in small group :

Task:

1. What mesures will you take to manage clients accounts?
2. Identify types of database user having by the bank xyz?
3. Use access method to create bank xyz users?

Activity 3: Application

A bank XYZ has a software which manages all their clients, wherethey have a database which contains informations of the clients.

Ask trainees to do the following tasks :

Task:

1. Having managers in the bank, as database administrator show how you will manage their passwords?
2. What precautions will you take when installing the software on user's Pcs?

Points to Remember

1. Always remember to log out after using the system
2. Use password which contains characters, numbers and signs.
3. Your Username and password are secret.

Formative Assessment

6. What is the difference between authentication and authorization?
7. List the types of database users?
8. What are the rules of database administrator on the system?

Self Reflection

Areas of strength	Areas for improvement	Actions to be taken to improve
1. **2.**	**1.** **2.**	**1.** **2.**