# FIRST QUARTER MODULE

| MODULE 1: INTRODUCTION TO DATABASES | |
|---|---|
| Content Standard | The learners demonstrate an understanding of key concepts, underlying principles and core competencies in database. |
| Performance Standard | The learners shall be able to independently create/provide quality product database using Microsoft access. |
| 21st Century Learning Skills | Critical Thinking-and-Doing, Computing/ ICT Literacy, Career and Self-reliance. |
| Core Values | Diligence and accuracy |

**REFERENCES:** *(Please be guided with the given references to help you perform the given activities. Click the given links and hyperlinks to access the suggested learning resources.)*


A. *Printed:*

- Bombase, Lilibeth S., et. al. My Computer Microsoft Access XP, Quezon City, Philippines: ABIVA Publishing House Incorporated

B. *Online Source:*

- Uganda ICT Teachers Association (UITA). (n.d.). ICT Teachers Association of Uganda (ITAU). Retrieved August 9, 2021, from https://www.ictteachersug.net/manual-vs-electronic-databases-advantages-disadvantages-and-features/_
- Creating an Append Query in Microsoft Access | Database Solutions for Microsoft Access | databasedev.co.uk. (n.d.). Http://Www.Databasedev.Co.Uk/Append_query.Html. Retrieved August 9, 2021, from http://www.databasedev.co.uk/append_query.html/
- *Introduction to Databases*. (n.d.). Https://Www.Tutorialspoint.Com/Introduction-to-Databases. Retrieved August 9, 2021, from https://www.tutorialspoint.com/Introduction-to-Databases/
- What is DBMS? Advantages and Disadvantages of Database Management System(DBMS). (n.d.). Http://Www.Myreadingroom.Co.in/Notes-and-Studymaterial/65-Dbms/462-Advantages-and-Disadvantages-of-Dbms.Html. Retrieved August 9, 2021, from http://www.myreadingroom.co.in/notes-and-studymaterial/65-dbms/462-advantages-and-disadvantages-of-dbms.html/



## TOPIC: INTRODUCTION TO DATABASES

### Introduction

A collected information which is in an organized form for easier access, management, and various updating is known as a database.

Before going into a further discussion of databases, we must have a prior knowledge of exactly what is a DATA? Data can be defined as a collection of facts and records on which we can apply reasoning or can-do discussion or some calculation. The data is always easily available and is in plenty. It can be used for processing some useful information from it. Also, it can be in redundant, can be irrelevant. Data can exist in form of graphics, reports, tables, text, etc. that represents every kind of information, that allows easy retrieval, updating, analysis, and output of data by systematically organized or structured repository of indexed information.

The goal of this lesson is to provide this basic introduction. We're going to explain the basics of what a database actually is, look at the history, understand relational databases, get into some fundamental concepts from columns and rows on up, touch on other types of databases, get versed on some additional concepts to understand, and wrap it all up with a quick review of the major commercial systems on the market today.

For the most part there are no pre-requisites to this tutorial other than basic computing knowledge.

**Below are the Learning Targets/ Specific Objectives:**

- Define Database System.
- Distinguish and explain the types of databases
- Create an info graphics about the types of database management system.
- Identify and Explain the Database Management System
- Compare and contrast the Database Management System to the Database System.

**Exploration of Prior Knowledge**

**Directions**: Fill in the K-W-H-L Chart below to assess your prior knowledge and understanding of the topic, **Introduction to Databases**

| What I Know | What I Want to Find Out | How I Can Learn More | What I Have Learned |
|---|---|---|---|
| | | | |
| **Skills I expect to use:** | | | |

**PROCESS QUESTIONS/ FOCUS QUESTIONS:**

**Below are the key guide questions that you should remember as you perform all the activities in this lesson. You should be able to answer them at the end of the week.**

1. **What is a database?**
2. **How do we classify a database?**
3. **What are the advantages of a database?**

**SHORT EXERCISES/DRILLS**

**Activity 1: Database Classification**

A. Classify whether the given database is **Manual** or **Electronic.**

| | |
|---|---|
| _____ | 1. Teachers' Grading Sheet in MS Excel |
| _____ | 2. List of visitors in logbook |
| _____ | 3. List of books available in the library |
| _____ | 4. Simple notes on the refrigerator |
| _____ | 5. List of telephone numbers and addresses |
| _____ | 6. Automated Enrolment System |

_____ 7. Computerized Library System

_____ 8. List of recipes in a restaurant

## CONTENT DISCUSSION:

A database is usually large collection of data organized especially for rapid search and retrieval (as by a computer) (Merriam-Webster's Collegiate dictionary, 11th Edition 2003). In simple terms, we can say that a database stores a huge amount of important data that can be easily retrieved. The manner in chic the data is stored, that is, on a computer, makes a database different from a simple collection of data on sheets of paper or in hard copy.

**What Is a Database?**

A database, in the most general sense, is an organized collection of data. More specifically, a database is an electronic system that allows data to be easily accessed, manipulated and updated.

In other words, a database is used by an organization as an electronic way to store, manage and retrieve information. The database is one of the cornerstones of enterprise IT, and its ability to organize, process and manage information in a structured and controlled manner is *the* key to many aspects of modern business efficiency.

However, databases go way beyond simply storing data. As we'll see later, the inherent logic and efficiency in *how* the data is stored and retrieved can provide an incredibly powerful business tool to an organization. This is especially true when databases are properly exploited for their reporting and business intelligence capabilities.

**The Use and Importance of Databases in Today's World**

So, what kinds of organization requires a database and can benefit from its use? Well, the short answer is any business or organization that needs to keep track of large numbers of customers or products. By "large" we mean more than can be stored by a human brain - a lot more.

At this point, a skeptic might still argue that there are countless mom-and-pop stores whose owners keep track of inventory and profit/ loss using the trusty ledger and calculator, and are doing OK. That's true, but the use of an electronic database can still pay off, even for very small businesses. For instance, a ledger cannot run a simulation to extrapolate profits if say, the shop was to increase the price of ballpoint pens by 2 cents. A database can do that. The ledger cannot run a report tracking down re-order levels for all items to show the store owner which items should be restocked at what times during the year. A database can do that too. A database can even automatically alert the business owner via email or text message.

The most significant benefit of databases, however, is still limited to large organizations with customers and products numbering in the hundreds of thousands, millions, or tens of millions, and the need to store large numbers of individual data items for customers. For instance, a commercial bank needs the personal details of all of its millions of customers, such as name, date of birth, address, Social Security number, etc. Each customer in turn spawns another collection of data depending on the products he or she has signed up for, such as account type, account number, account balance, mortgage amount, credit card loan, repayment period and so on. A third collection of data relates to the customer's specific transactions, such as the time of transaction, amount, balance left, bank charges, loan amount left to repay, etc.

Clearly, a single customer can generate a huge amount of data in a very short time. Multiply this by millions of customers, and it's easy to see why having an efficient data storage and retrieval medium is not only a good idea, but is also absolutely essential in preventing the bank's operations from grinding to a halt.

Commercial banks are a prime example of the use of databases in today's organizations. Other industries whose operations are heavily reliant on databases are insurance companies, hospitals and health care, schools and colleges, manufacturing, telecommunications companies, and hotels and the hospitality industry.

**History of Databases**

It is important to first realize that the organized, systematic methodology of storing records we know and heavily depend on in databases is not a recent invention. What is recent is the computerization of this methodology beginning in the 1960s. Note that even paper-based records, including ledger-based bookkeeping, are (technically) all forms of a database. That is, a database does not necessarily have to be computerized. Computerization only produced a database management system (DBMS), which is obviously several orders of magnitude more powerful, accurate and capable than what a humble ledger or a puny human brain can achieve. And although we are mostly using the term "database" to refer to the DBMS, the two are not the same thing; all thumbs (DBMSs) are fingers (databases), but not all fingers are thumbs.

The ancient Egyptians used elaborate record-keeping systems to keep stock of grain harvests. The Library of Alexandria employed a sophisticated method to keep track of huge numbers of books and scrolls. These were all early examples of databases, although of course their capabilities would be laughable compared to the hugely capable computerized DBMSs of the 21st century.

But even way back in time, back when the entire field of computing was still in its single-celled-organism stage (the 1960s), many people could already visualize that computer would be truly useful if they could provide a way of reliably storing and retrieving data. The development of databases therefore occurred almost in perfect step with the general development and growth of the computing capabilities of the day. As disk capacity and processor speed grew, so did the storage capacity and feature sets of the contemporary database offerings. One important leap that occurred in the mid-1960s was the switch from tape-based storage to direct access storage, or disks. This change allowed multitasking interactive data access, as opposed to the single-operator, batch-type processing necessitated by tapes.

The earliest database systems were navigational in nature. This means that applications processed and read data by using pointers embedded in the data itself. The pointer led to the next data item and could be doubly linked, allowing linkage to both the previous and next data items. This is similar to how hyperlinks work on a Web page by leading the reader to a related Web page from the current one. The two main data models at this time were the hierarchical model epitomized by IBM's IMS system, and the Codasyl, or network, model. But all these were bested and reduced to mere interesting footnotes in history by the emergence of the relational model by a brilliant computer scientist by the name of E.F. Codd.

E.F. Codd and the Relational Model

The relational model was a radical departure from the reigning hierarchical model in that it focused on the ability to search a database by content rather than by following a linked navigation system. This offered the significant advantage of allowing databases to grow and store more and more data, all without having to change or rewrite the applications that accessed that data. Essentially, Codd single-handedly designed a way to divorce the skeleton or structure of the database from the data records held in the database. So elegant was this model that it is the de facto standard for database design to this day, with such databases termed relational databases. There are a few very important non-relational databases (especially with the advent of big data and Web 2.0), but the relational model is still used for the overwhelming majority of commercial database offerings.

Today, E.F. Codd's name would mostly evoke a nonchalant "E.F. who?" among most people, even many in the IT industry. However, his work has directly led to the huge benefits and efficiency that relational databases provide. His contribution to the world of computing is comparable in scale to that of Sir Isaac Newton's to the world of physics.

Codd attended Oxford college, studying mathematics and chemistry, then worked as a pilot in the Royal Air Force during WWII before moving to the U.S. in 1948 to work as a mathematical

programmer for IBM. After spending a decade in Canada, he returned to the U.S. in 1963 and received his Ph.D. in 1965.

In 1970, Codd published a paper on data management titled "A Relational Model of Data for Large Shared Data Banks" for IBM. The giant company, however, was heavily invested in the hierarchical model via its Information Management System (IMS), and Big Blue executives were not interested in developing a competitor for one of their own lucrative product lines. Showing guile rarely seen in academic or scientific types, Codd slyly showed his model to select IBM customers, who upon viewing it needed little convincing of its superiority. The influential customers in turn put pressure on the very same IBM executives to develop the model and they reluctantly (and, one imagines, seething quietly with fury at Codd) placed the model under development in IBM's Future Systems project, with the system itself known as System R.

However, the head honchos were still unwilling to threaten IMS, and sabotaged Codd's work by placing the System R project in the hands of developers who were unfamiliar with it. The developers thus failed to use Codd's own Alpha language for development, instead electing to use a much simpler language known as SEQUEL. This turned out to be an accidental masterstroke, however, since SEQUEL is much easier to understand and use. For copyright reasons, the name was changed to SQL, and is very familiar to database developers and administrators today as the language of choice for writing database queries.

A shrewd young businessman who was developing his own database system read about SQL at a conference in 1979. He recognized its superiority and copied the language into a database product by his own small company. The businessman had also previously seen Codd's work on the relational model, and became convinced that it was the way to go for database systems. He based his own product on it, even though IBM refused to share System R's code with him. Remember, IBM was not interested in the relational model. That small company has grown quite a bit; today it's known as Oracle Corp. As for the businessman, his name is Larry Ellison, and his conviction helped him become one of the richest people in the world. It just goes to show how badly IBM miscalculated the potential of Codd's relational model. In fact, Oracle DB is the most widely used relational database for corporations today.

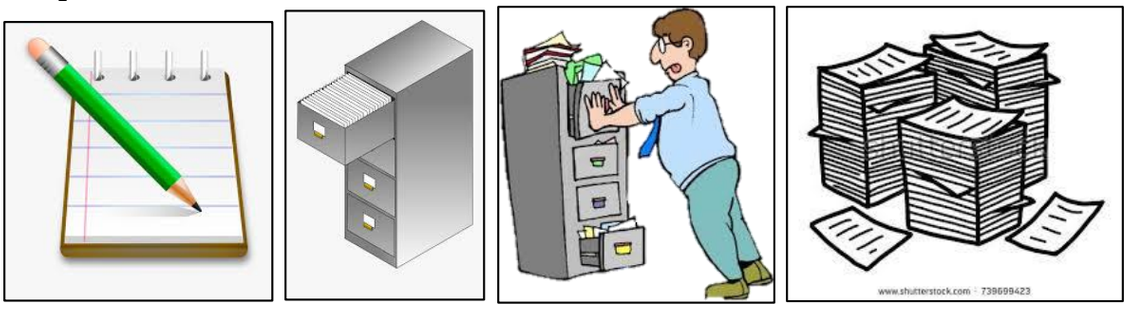
**Classification of Database**

There are two main types of databases-manual and electronic – defined in more detail below.

**Manual database**

- A manual database is simply a list of items on a sheet of paper. A notebook that contains information on a particular topic such as a list of students in a class or a list of items sold in a store, for instance, is considered a manual database in the sense that data is stored by means of writing by hand.
- A filing cabinet in an office can be considered a manual database because it contains hundreds or even thousands of records in hard copy stored in folders. These folders are then arranged alphabetically and stored in filing cabinets. The manner in which the records are kept makes the filing cabinet a manual database.
- In this kind of database, imagine how hard it is to find a record by sorting through hundreds or even thousands of folders.

**Sample Manual database**



**Electronic database**

- An electronic database, also known as a computerized database, on the other hand, is a database created and kept using a computer.
- This kind of database can store hundreds and even thousands or records in a single file, providing users with the following advantages compared with using a manual database:
  - Speed – An electronic database allows users to quickly find or browse through thousands of records in a second with just a few mouse clicks, you can search through all of the records stored in a particular database.
  - Compact – it is also compact, which means you can store thousands of records in a single disk. An electronic database can store all of the records in a filing cabinet in a single floppy disk.
  - Flexibility – an electronic database also provides users flexibility, allowing them to find data by just typing in a set of criteria. For instance, you can find a record of all the students living in Manila or students in Mr. Santos' class or even students in Mr. Santos' class that live within Metro Manila.

**Sample Electronic Database**

- To explain it further you may watch the simple video below:


Access: Introduction to Databases

**YouTube Link:** https://www.youtube.com/watch?v=eXiCza050ug/

**What Is A Database Management System (DBMS)**

*"A database management system (DBMS) is a collection of programs that manages the database structure and controls access to the data stored in the database"*.

The DBMS serves as the intermediary between the user and the database. The database structure itself is stored as a collection of files, So, we can access the data in those files through the DBMS.

The DBMS receives all application requests and translates them into the complex operations required to fulfill those requests. The DBMS hides much of the database's internal complexity from the application programs and users.

**Advantages of Database Management System (DBMS)**

***1. Improved data sharing*** - An advantage of the database management approach is, the DBMS helps to create an environment in which end users have better access to more and better-managed data. Such access makes it possible for end users to respond quickly to changes in their environment.

***2. Improved data security*** - The more users access the data, the greater the risks of data security breaches. Corporations invest considerable amounts of time, effort, and money to ensure that corporate data are used properly. A DBMS provides a framework for better enforcement of data privacy and security policies.

3. Better data integration - Wider access to well-managed data promotes an integrated view of the organization's operations and a clearer view of the big picture. It becomes much easier to see how actions in one segment of the company affect other segments.

***4. Minimized data inconsistency*** - Data inconsistency exists when different versions of the same data appear in different places. For example, data inconsistency exists when a company's sales department stores a sales representative's name as "Bill Brown" and the company's personnel department stores that same person's name as "William G. Brown," or when the company's regional sales office shows the price of a product as $45.95 and its national sales office shows the same product's price as $43.95. The probability of data inconsistency is greatly reduced in a properly designed database.

***5. Improved data access*** - The DBMS makes it possible to produce quick answers to ad hoc queries. From a database perspective, a query is a specific request issued to the DBMS for data manipulation—for example, to read or update the data. Simply put, a query is a question, and an ad hoc query is a spur-of-the-moment question. The DBMS sends back an answer (called the query result set) to the application. For example, end users, when dealing with large amounts of sales data, might want quick answers to questions (ad hoc queries) such as:

- What was the dollar volume of sales by product during the past six months?
- What is the sales bonus figure for each of our salespeople during the past three months?
- How many of our customers have credit balances of 3,000 or more?

***6. Improved decision making*** - Better-managed data and improved data access make it possible to generate better-quality information, on which better decisions are based. The quality of the information generated depends on the quality of the underlying data. Data quality is a comprehensive approach to promoting the accuracy, validity, and timeliness of the data. While the DBMS does not guarantee data quality, it provides a framework to facilitate data quality initiatives.

***7. Increased end-user productivity*** - The availability of data, combined with the tools that transform data into usable information, empowers end users to make quick, informed decisions that can make the difference between success and failure in the global economy.

**Disadvantages of Database Management System (DBMS):**

Although the database system yields considerable advantages over previous data management approaches, database systems do carry significant disadvantages. For example:

***1. Increased costs*** - one of the disadvantages of DBMS is Database systems require sophisticated hardware and software and highly skilled personnel. The cost of maintaining the hardware, software, and personnel required to operate and manage a database system can be substantial. Training, licensing, and regulation compliance costs are often overlooked when database systems are implemented.

***2. Management complexity*** - Database systems interface with many different technologies and have a significant impact on a company's resources and culture. The changes introduced by the adoption of a database system must be properly managed to ensure that they help advance the company's objectives. Given the fact that database systems hold crucial company data that are accessed from multiple sources, security issues must be assessed constantly.

***3. Maintaining currency*** - To maximize the efficiency of the database system, you must keep your system current. Therefore, you must perform frequent updates and apply the latest patches and security measures to all components.

Because database technology advances rapidly, personnel training costs tend to be significant. Vendor dependence. Given the heavy investment in technology and personnel training, companies might be reluctant to change database vendors.

As a consequence, vendors are less likely to offer pricing point advantages to existing customers, and those customers might be limited in their choice of database system components.

4. Frequent upgrade/replacement cycles - DBMS vendors frequently upgrade their products by adding new functionality. Such new features often come bundled in new upgrade versions of the software. Some of these versions require hardware upgrades. Not only do the upgrades themselves cost money, but it also costs money to train database users and administrators to properly use and manage the new features.

**REVISED KNOWLEDGE: Actual answer to the process questions/ focus questions**

1. **What is a database?**
   A database, in the most general sense, is an organized collection of data. More specifically, a database is an electronic system that allows data to be easily accessed, manipulated and updated.

2. **How do we classify a database?**
   We can classify it as manual database and electronic database

3. **What are the advantages of a database?**

   1. Improved data sharing
   2. Improved data security
   3. Better data
   4. Minimized data inconsistency
   5. Improved data access
   6. Improved decision making
   7. Increased end-user

**FINAL KNOWLEDGE: Generalization/ Synthesis/ Summary**

Databases are indeed what the term says - the base or pool of (related) data upon which an organization's data systems should be built. Indeed, it is not a stretch to say that without databases, the computer in the workplace would be little more than a more efficient typewriter.

# DBES LEARNING ACTIVITY SHEET/GAWAING PAGKATUTO
# COMPUTER TECHNOLOGY 9
**First Quarter - Module 1**

_____
*Subject Teacher*

| | |
|---|---|
| **Name of Learner:** _____ _____ _____<br>***Lastname*** ***Firstname*** ***MI***<br>**Section :** _____ **Date Completed:** _____ | **SCORE:** |

**Activity 1:**

1. What is a Database System and why do we use them?
   _____
   _____
   _____
   _____
   _____
2. Why is there a need to create a computerized database?
   _____
   _____
   _____
   _____
   _____
3. Which is more effective and efficient to use, the manual database or the electronic database? Explain.
   _____
   _____
   _____
   _____
   _____

## ACTIVITY 2: Infographics for Database Classification (G.R.A.S.P.S Activity)

- The computer club president of a prestigious University aims to post an explanation about the two types of Database System in a graphical manner, to be posted on their own bulletin board located in front of their club office, so that the students and visitors can see it for their additional knowledge or awareness about the Database System by just looking on their bulletin board. As their layout artist you are tasked to create a simple Infographic showing an explanation about the types of Database System. The Infographic will be checked through the following standards: Content, Organization and efficiency.
- Guide: Draw your infographic in a short bond paper

**Rubric for Checking**

| | 5 | 4 | 3 | 2 | Score |
|---|---|---|---|---|---|
| Content | The content of the Info Graphics is complete | The content of the Info Graphics is somewhat complete | Some of the content of the Info Graphics are missing | Several content of the Info Graphics are missing. | |
| Organization | The Info Graphics is organized, clean and well presented. | The Info Graphics is somewhat organized, clean and well presented. | The organization, cleanliness of the Info Graphics are acceptable | The organization, cleanliness and presentation of the Info Graphics are not acceptable. | |
| Efficiency | The student finished the diagram in less than the given time limit. | The student is somewhat finished the Info Graphics within the given time limit. | The student is able to finished the Info Graphics on the last minute of the given time limit | The student did not finished the Info Graphics within the given time limit. | |
| | | | | | |

| MODULE 2: FAMILIARIZING THE DATABASE MANAGEMENT SYSTEM (DBMS) | |
|---|---|
| Content Standard | The learners demonstrate an understanding of key concepts, underlying principles and core competencies in database. |
| Performance Standard | The learners shall be able to independently create/provide quality product database using Microsoft access. |
| 21st Century Learning Skills | Critical Thinking-and-Doing, Computing/ ICT Literacy, Career and Self-reliance. |
| Core Values | Diligence and accuracy |

**REFERENCES:** *(Please be guided with the given references to help you perform the given activities. Click the given links and hyperlinks to access the suggested learning resources.)*


*A. Printed:*

- Bombase, Lilibeth S., et. al. My Computer Microsoft Access XP, Abiva Quezon City, Philippines: ABIVA Publishing House Incorporated

*B. Online:*

- Raza, M. (n.d.). DBMS: An Intro to Database Management Systems. BMC Blogs. Retrieved August 9, 2021, from https://www.bmc.com/blogs/dbms-database-management-systems/



## TOPIC: FAMILIARIZING THE DATABASE MANAGEMENT SYSTEM (DBMS)

### Introduction

Database Management Systems (DBMS) refer to the technology solution used to optimize and manage the storage and retrieval of data from databases. DBMS offers a systematic approach to manage databases via an interface for users as well as workloads accessing the databases via apps.

**Below are the Learning Targets/ Specific Objectives:**

- Analyze the table consisting records
- Identify and Explain the Database Management System
  Compare and contrast the Database Management System to the Database System.

**Exploration of Prior Knowledge**

**Directions**: Fill in the K-W-H-L Chart below to assess your prior knowledge and understanding of the topic, **Familiarizing the database management system.**

| What I Know | What I Want to Find Out | How I Can Learn More | What I Have Learned |
|---|---|---|---|
| | | | |
| **Skills I expect to use:** | | | |

**PROCESS QUESTIONS/ FOCUS QUESTIONS:**

**Below are the key guide questions that you should remember as you perform all the activities in this lesson. You should be able to answer them at the end of the week.**

1. What is the purpose of DBMS?
2. How does hierarchy of data in database work?

**SHORT EXERCISES/DRILLS**

**Identify the terms being described.**

_____ 1. This is like a filing cabinet drawer that contains several folders.
_____ 2. These are like folders in the drawer – they deal with different aspects of the
_____ 3. A collection of inter-related data and set of programs to store & access those data in an easy and effective manner.
_____ 4. This refers to the systematic organization of data, often in a hierarchical form.

## CONTENT DISCUSSION:

DBMS stands for **D**ata**b**ase **M**anagement **S**ystem. We can break it like this DBMS = Database + Management System. Database is a collection of data and Management System is a set of programs to store and retrieve those data. Based on this we can define DBMS like this: DBMS is a collection of inter-related data and set of programs to store & access those data in an easy and effective manner.

**Purpose of Database Systems**

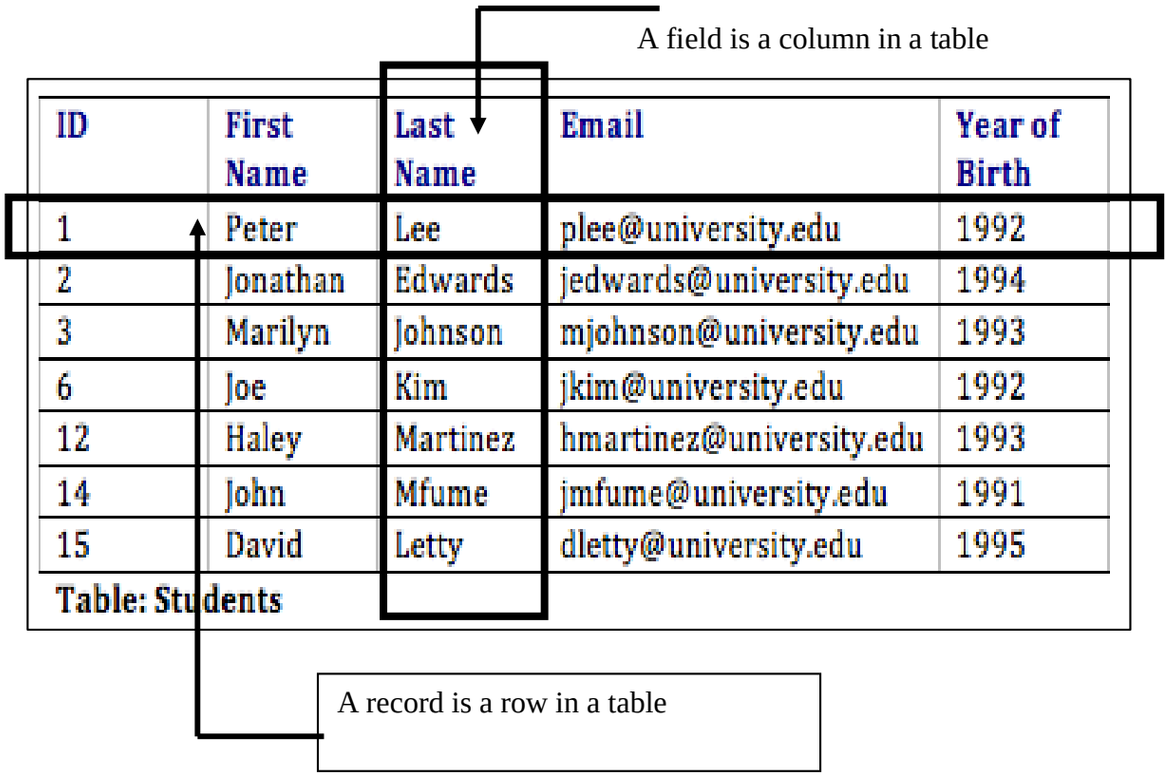
The main purpose of database systems is to manage the data. Consider a university that keeps the data of students, teachers, courses, books etc. To manage this data we need to store this data somewhere where we can add new data, delete unused data, update outdated data, retrieve data, to perform these operations on data we need a Database management system that allows us to store the data in such a way so that all these operations can be performed on the data efficiently. Database systems are much better than traditional file processing systems which we have discussed in the separate article: DBMS vs File System.

## HIERARCHY OF DATA IN A DATABASE

Data Hierarchy refers to the systematic organization of data, often in a hierarchical form. Data organization involves characters, fields, records and files.

A database is composed of **tables**. Each **table** is composed of **records**. Each **record** is composed of **fields**. This is the hierarchy of data in a database. It is the way the contents of the database are organized.

**Sample database table**




| ID | First Name | Last Name | Email | Year of Birth |
|---|---|---|---|---|
| 1 | Peter | Lee | plee@university.edu | 1992 |
| 2 | Jonathan | Edwards | jedwards@university.edu | 1994 |
| 3 | Marilyn | Johnson | mjohnson@university.edu | 1993 |
| 6 | Joe | Kim | jkim@university.edu | 1992 |
| 12 | Haley | Martinez | hmartinez@university.edu | 1993 |
| 14 | John | Mfume | jmfume@university.edu | 1991 |
| 15 | David | Letty | dletty@university.edu | 1995 |

**Table: Students**

## DATABASE MANAGEMENT SYSTEM (DBMS)

A database on its own can only store data and used as a source of information. This means that it is simply a collection of related data. How data is stored in a database requires knowledge in creating a DBMS. A DBMS is a collection of programs that enables you to store, modify, and extract information from a database (Webopedia, 2008). In other words, it is a system designed for the following purposes:

1. To add more data into an already existing database.
2. To delete unnecessary data in an existing database
3. To update data in an existing database.
4. To provide users various ways to view data stored in a database (example: in soft copy or on a computer monitor or in hard copy or printed form).

**Note:** Always remember that DBMS is an Application Software that allows you to organized and compile information like for example; the Microsoft Office Word, Excel and PowerPoint wen can consider those applications as DBMS because we use them to organized information. In computer programming we are going to use the DBMS such as MS Access, SQL and the like.

Please take a look and analyze the diagram below to better understand the DBMS.

Here are the data that is collected by the user

Now after collecting the data, the user now encode the data using the DBMS


And now the DBMS is going to store the data collected by the user to the Database System

This is now the Database System where the Information or data is store by the user through the different DBMS Software.

# DBES LEARNING ACTIVITY SHEET/GAWAING PAGKATUTO
# COMPUTER TECHNOLOGY 9
**First Quarter - Module 2**

_____
*Subject Teacher*

| | |
|---|---|
| **Name of Learner:** _____ _____ _____<br>***Lastname*** ***Firstname*** ***MI***<br>**Section :** _____ **Date Completed:** _____ | **SCORE:** |

**Written Works.**

**Activity 1.** Answer the questions briefly but substantially.

1. What is a Database Management System (DBMS)? And how does it work?

2. Why is the DBMS important?

3. If you were a computer programmer, can you create a system without the database management system at ease? Explain your answer.

**Activity 2: Table Analysis**

| ID | First Name | Last Name | Email | Year of Birth |
|---|---|---|---|---|
| 1 | Peter | Lee | plee@university.edu | 1992 |
| 2 | Jonathan | Edwards | jedwards@university.edu | 1994 |
| 3 | Marilyn | Johnson | mjohnson@university.edu | 1993 |
| 6 | Joe | Kim | jkim@university.edu | 1992 |
| 12 | Haley | Martinez | hmartinez@university.edu | 1993 |
| 14 | John | Mfume | jmfume@university.edu | 1991 |
| 15 | David | Letty | dletty@university.edu | 1995 |

**Table: Students**

**1.** How many records do we have in the students' table?

_____

2. Name the five (5) fields in the table.

_____
_____
_____
_____
_____

3.How does the records in the table being stored?

_____

_____

_____

_____

| MODULE 3: LEARNING THE BASIC STEPS IN DATABASE DESIGN | |
|---|---|
| Content Standard | The learners demonstrate an understanding of key concepts, underlying principles and core competencies in database. |
| Performance Standard | The learners shall be able to independently create/provide quality product database using Microsoft access. |
| 21st Century Learning Skills | Critical Thinking-and-Doing, Computing/ ICT Literacy, Career and Self-reliance. |
| Core Values | Diligence and accuracy |

**REFERENCES:** *(Please be guided with the given references to help you perform the given activities. Click the given links and hyperlinks to access the suggested learning resources.)*


*A. Printed:*

- Bombase, Lilibeth S., et. al. My Computer Microsoft Access XP, Quezon City, Philippines: ABIVA Publishing House Incorporated

*B. Online:*

- Naeem, T. (2021, July 12). Database Design - Overview, Importance, and Techniques. Astera. https://www.astera.com/type/blog/all-you-need-to-know-about-database-design/

- Microsoft. (n.d.). Database design basics. Retrieved August 9, 2021, from https://support.microsoft.com/en-us/office/database-design-basics-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5/



**TOPIC: Learning the Basic Steps in Database Design**

## Introduction

A database includes bulk information deposited in a framework, making it easier to locate and explore relevant information. A well-designed database contains accurate and up-to-date information, allowing data to be fetched easily whenever needed. It is easy to understand the importance of a database for a company dealing with heaps of data regularly. However, it is significant to note that it requires a data design that can make analysis faster and reliable.

**Below are the Learning Targets/ Specific Objectives:**

- Distinguish and discuss the kinds of electronic database
- Create a simple flat file database in a table
- Define the data needed in the database
- Identify and discuss the three phases of database design.
- Analyze and apply the breaking down of fields into smaller parts.

**Exploration of Prior Knowledge**

**Directions**: Fill in the K-W-H-L Chart below to assess your prior knowledge and understanding of the topic, **Learning the Basic Steps in Database Design**

| What I Know | What I Want to Find Out | How I Can Learn More | What I Have Learned |
|---|---|---|---|
| | | | |
| **Skills I expect to use:** | | | |

**PROCESS QUESTIONS/ FOCUS QUESTIONS:**

**Below are the key guide questions that you should remember as you perform all the activities in this lesson. You should be able to answer them at the end of the week.**

1. **What is a database design?**
2. **What is a good database?**
3. **What are the phases of database design?**

**SHORT EXERCISES/DRILLS**

**TRUE OR FALSE. Tell whether the statement is correct or not. Write TRUE if the statement tells the truth and FALSE if otherwise.**

_____ 1. The main purpose of designing a database is to produce physical and logical models of designs for the proposed database system.

_____ 2. A good database design process is governed by specific rules.

_____ 3. Flat-file database is designed to put up all the selected information in one table.

_____ 4. Relational database is a type where one table is linked or related to other tables in a database.

_____ 5. In data definition, all the data and information needed for the database are collected and determined.

## CONTENT DISCUSSION:

This lesson will teach the basis of relational database design and explains how to make a good database design. It is a rather long text, but we advise to read all of it. Designing a database is in fact fairly easy, but there are a few rules to stick to. It is important to know what these rules are, but more importantly is to know why these rules exist, otherwise you will tend to make mistakes. Standardization makes your data model flexible and that makes working with your data much easier.

A good database design starts with a list of the data that you want to include in your database and what you want to be able to do with the database later on. This can all be written in your own language, without any SQL. In this stage you must try not to think in tables or columns, but just think: "What do I need to know?" Don't take this too lightly, because if you find out later that you forgot something, usually you need to start all over. Adding things to your database is mostly a lot of work.

**What is Database Design?**

Database design is defined as a collection of steps that help with designing, creating, implementing, and maintaining a business's data management systems. The main purpose of designing a database is to produce physical and logical models of designs for the proposed database system.

**What is a Good Database Design?**

A good database design process is governed by specific rules. The first rule dictates that redundant data must be avoided; as it wastes space and increases the probability of faults and discrepancies within the database. The next rule is that the accuracy and comprehensiveness of information is extremely imperative. If the database contains erroneous information, any documents that fetch data from such a database will also include inaccurate information. Consequently, any decisions based on those documents will be misleading, thus, increasing the importance of a database design that caters to all of the above rules.

So, how can you ensure that your database design is good? A well-designed database is the one that:

- Distributes your data into tables based on specific subject areas to decrease data redundancy
- Delivers the database the information needed to link the data in the tables
- Provides support, and guarantees the precision and reliability of data
- Caters to your information processing and reporting requirements
- Functions interactively with the database operators as much as possible

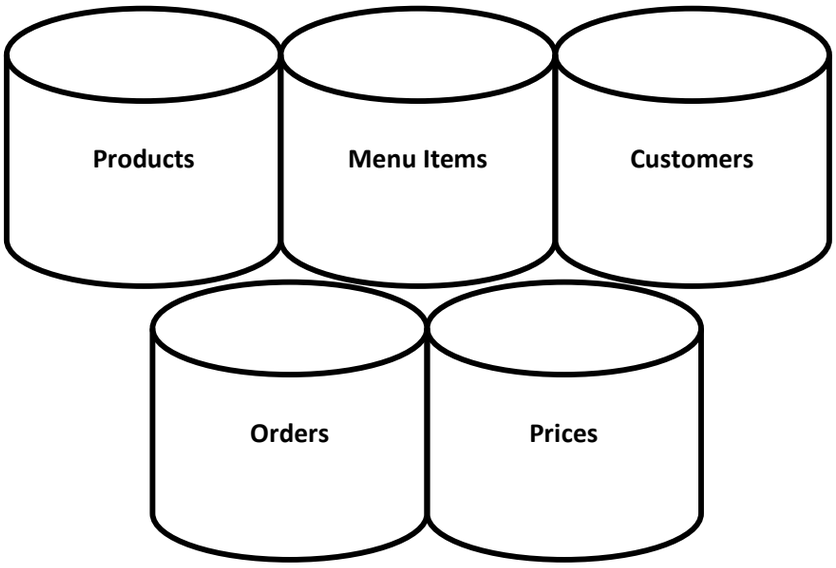
## KINDS OF ELECTRONIC DATABASE

There are two kinds of electronic databases, namely:

1. **Flat-file database –**is designed to put up all the information in one table. An example of this is a simple list of students enrolled in different departments using Microsoft Excel. This kind of database only uses one table to store all kinds of data. It is often referred to as a simple database. For instance, you only need to keep a list of all of the items inside your cabinet; you can then use a flat-file database to store the names of the items in it. A flat-file database is very similar to a spreadsheet in MS Excel wherein data is arranged in rows and columns. The figure below shows a flat-file database created in MS Excel.

| Student ID | Student Name | Contact Number | College | Course | SubjectCode | SubjectDescription | Units |
|---|---|---|---|---|---|---|---|
| 1425-56 | Leo Bernas | 0917-7782334<br>0918-9660002 | Science | Botany | B-058 | Ecology 2 | 3 |
| 1425-56 | Leo Bernas | 0917-7782334<br>0918-9660002 | Science | Botany | B – 091 | Chemistry 1 | 3 |
| 1426-57 | Adrianne Bajar | 0917-6558290<br>0918-7000232 | Science | Physics | L-023 | Physics 2Lab | 2 |
| 1426-57 | Adrianne Bajar | 0917-6558290<br>0918-7000232 | Science | Physics | B-024 | Physics 2Lec | 3 |
| 1427-58 | Manilyn Co | 0917-4332209 | Arts & Letters | Economics | E-014 | English 1 | 3 |

2. **Relational database –** is a type where one or more tables are linked or related to other tables in a database. An example of this is the Products database wherein it comprises of 5 different Tables: Products table, Menu Items table, Customers table, Orders table and Prices table.

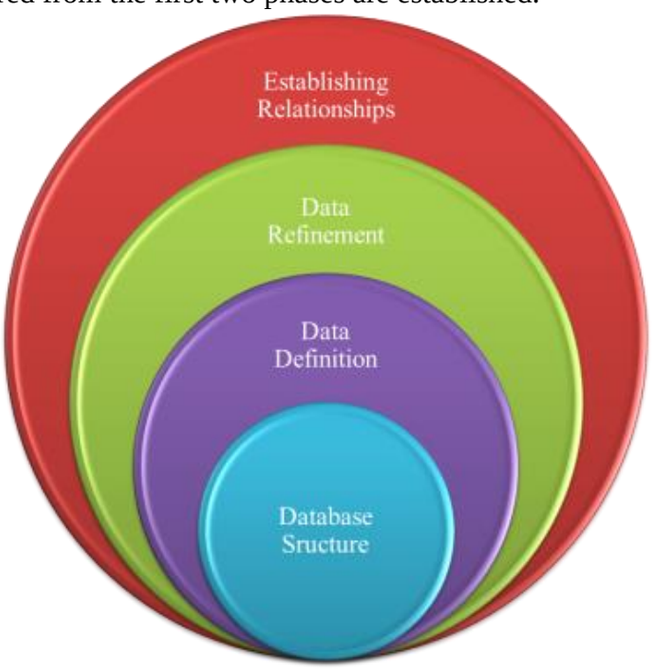



## DATABASE PLANNING AND DESIGN

Before you start creating an actual database application you first need to plan and design your database. The planning stage plays a very important role in creating a database because it allows you to determine what you need/ do not need to do. It also allows you to avoid problems during the development stage. During the design stage, you can define and refine data then just make further adjustments to your database as needed by eliminating potential errors.

**There are three phases of database design, namely:**

1. Data definition – in this phase, all the data and information needed for the database are collected and determined.
2. Data refinement – in this phase, the data and information you gathered are then refined by adding necessary data or eliminating unnecessary data
3. Establishing relationships – in this phase, relationships or links between the information you gathered from the first two phases are established.




In simpler terms, these three phases can be translated into the following:

1. Identify fields for the database and determining in which tables they belong.
2. Removing unnecessary fields or replacing and adding other fields
3. Linking tables together by identifying common data fields among them

However, simply knowing these three phases does not guarantee that your database will work perfectly. You need to know what things should be done under each phase. The following sections discuss the basic steps involved in the three phases of planning and designing a database.

## BASIC STEPS IN DATABASE DESIGN

**Planning the database** comes before everything else when designing a database application. In this step, you should determine the purpose, goals, or objectives of the database. You need to address the following issues to determine how the database should work:

a. What data or information is needed for the database?
b. Who will use the database?
c. What data will users enter into the database?
d. What kind of outputs should the database produce?

The issues mentioned above are just examples of what you should know before you start designing a database's structure. Depending on the database application you will develop, more issues or questions regarding its operation may arise. As you go through this lesson, you will create your own simple database design, following each step in the process. We will call this project of yours the "Student Database."

**Example: The Student Database**

Now, determine the purpose of your database and what it should be able to do:

- Stores and display records of students' personal information
- Store and display records of students' subject information
- Store and display each of the students' teachers' names per subject
- Track and monitor each student's grade per subject
- Find out which students need to improve their grades

Now that you know what your database should do, you can proceed to the next step that is, determining what fields you need for your database.

### Determine the fields needed in your database

In this step, think of all the information you need to gather to build your database. As previous discussed, fields are small pieces of data that make up a record in a table. When defining a field, think of it as a piece of data that pertains to a certain person or subject. For instance, in your project, you know that your database should be able to store information about students. So, make a list of the necessary data for your database.

The list on the right may be referred to as raw data or fields. Carefully observe it and try to find potential problems that you may encounter once you start developing your database. You have yet to define the proper fields based on the list that you have. However, to do that, you need to be aware of the field definition principles that will help you avoid problems in the development stage.

Example:

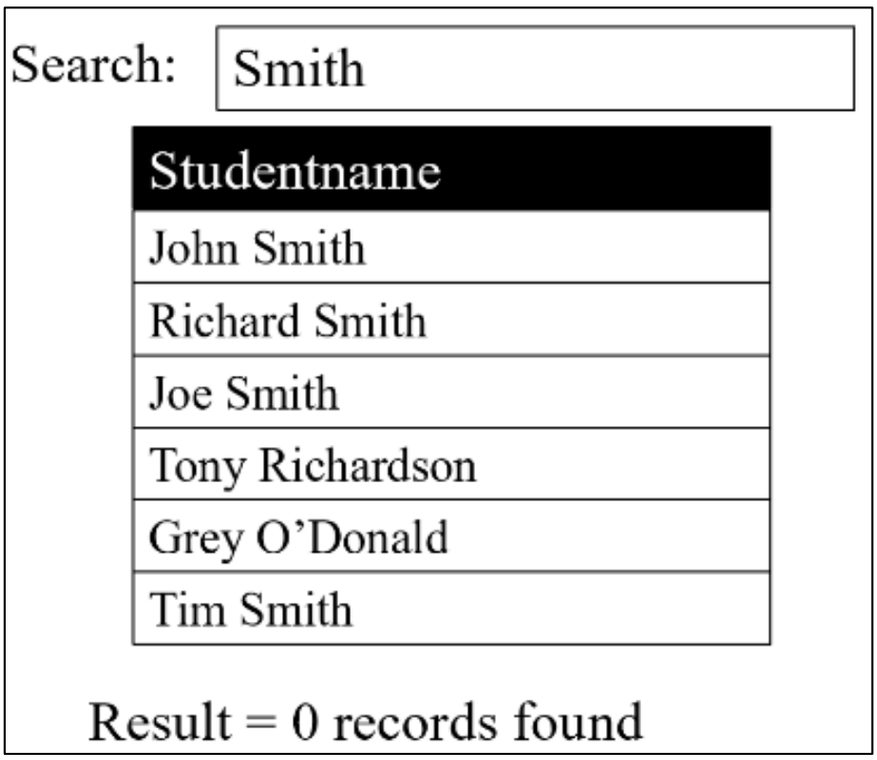
**Students Info – Student name**
**Address**
**Phone number**

## Field definition principles

- Break down into smaller parts.

This means you need to break down your original list into smaller pieces whenever necessary. You can determine if this is needed by analyzing each piece of data in your list. So, look at the example above the first data “Student name”

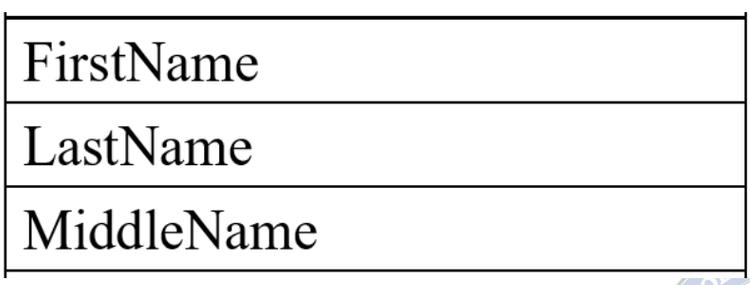
Looking at Student name, there does not seem to be a problem because all it should do is store the students’ names, right? However, suppose you need to find the record of a student whose last name is Smith? Remember that a field is just a small piece of data in a record and MS Access look through the records in a database using fields. So, if you try searching for just a student’s las or first name, you will not be able to find a matching field because the fields contain the full names of students.




As such, the result would be 0 because none of the data records in the field matches Smith even through at least four students have Smith as last name.

To eliminate this problem, you need to break down the field into smaller field as in the example in the figure on the right. Instead of just the field, Student name, you can replace this one field with the First Name, Last Name, and Middle Name fields.

With these three fields, you made your database more flexible by enabling users to search records by First Name, Last Name, or Middle Name. The figure below shows an example of how this will work.



Now, do the same to the other data. Carefully examine each field and think of ways by which you may use fields to search for a particular record or group of records then determine if a field needs to be broken down into smaller fields. With this in mind, proceed with the other data in your list.

**Find students with the last name "Smith"**

**Lastname = "Smith"**

| Firstname | Lastname | Middlename |
|---|---|---|
| John | Smith | A |
| Richard | Smith | B |
| Joe | Smith | C |
| Tony | Richardson | D |
| Grey | O'Donald | E |
| Tim | Smith | F |

**Result = 4 records found**

Notice that you left some of the fields unchanged. You also added and removed some fields then replaced the field names with more descriptive ones. For instance, you narrowed down the Teachers field into Teach name and Subject taught and the Subjects field into Subject name and Subject grade. Notice, too, that you took out the field, Grades, and combined it with Subjects to avoid confusion when determining which grade is for what subject.

| ORIGINAL FIELDS | NEW FIELDS | REASONS FOR BREAK DOWN |
|---|---|---|
| ~~Student Name~~ | Firstname<br>Lastname<br>Middlename | Will provide more flexible search options using either firstname, lastname, or middlename |
| ~~Address~~ | StAddress<br>City | Will allow users to search for students living in the same city. |
| Phone Number | Phonenumber | No change |
| ~~Teachers~~ | TeacherName<br>SubjectTaught | Will allow the user to determine the teacher for a particular subject. |
| Adviser | Advisername | No change |
| Subjects | SubjectName<br>SubjectGrade | To determine the student's grade for each subject. |
| Average | Average | No change. |
| Level | Level | No change |
| Section | Section | No change |
| Birthdate | Birthdate | No change |
| Age | Age | No change |

- Do not create a field that contains a list of values

Another principle when defining fields is avoid creating a field that contains a list of values. For instance, the Subject name field in your list was originally called subjects because a student takes more than just one subject.

The same thing goes for the Grades and Teachers fields. In the original list, your data entry for a record is shown in the following figure.

| Field | Data |
|---|---|
| Subject name | Math, Science, Filipino, English |

This kind of design will cause problems when a user searches for a particular subject. If you are only looking for a student's math grade, you may get the wrong result because the fields contain the names of all subjects – Math, Science, Filipino, and English.

The same will be true for the Teacher name field since the students have different teachers per subject, your field will look like the figure below.

| Field | Data |
|---|---|
| Teacher name | Mr. Guevarra, Mrs. Guevarra, Ms. Santos |

Imagine how hard it will be to determine which teacher teaches what subject if you have a table that looks like the figure below.

| Field | Data |
|---|---|
| Subject name | Math, Science, Filipino, English |
| Teacher name | Mr. Guevarra, Mrs. Guiguinto, Ms. Santos |
| Subject grade | 85%, 80%, 87%, 88% |

- Do not create similar fields in one table

Some may think that this problem can be resolved by creating additional fields to hold other data. For instance, if a student has four subjects then you may try to create fields like Subject1, Subject2, Subject3, and Subject4. Then create another set of fields like Teacher1, Teacher2, Teahcer3, and teacher4. Look at how this will turns out in the following table.

| Fields | Data | Fields | Data | Fields | Data |
|---|---|---|---|---|---|
| Subject1 | Math | Teacher1 | Mr. Guevarra | Grade1 | 85% |
| Subject2 | Science | Teacher2 | Ms. Guiguinto | Grade2 | 80% |
| Subject3 | English | Teacher3 | Ms. Santos | Grade3 | 87% |
| Subject4 | Filipino | Teacher4 | Ms. Santos | Grade4 | 88% |
| | | | | | |

Looks okay, does it not? However, what if you need to add a fifth subject? Then you will need to reprogram and redefine your database and that just will not do! Once you start using your database and created all the forms, reports, and queries for your program, redefining the data structure will cause you a lot of work and you may end up reprogramming everything.

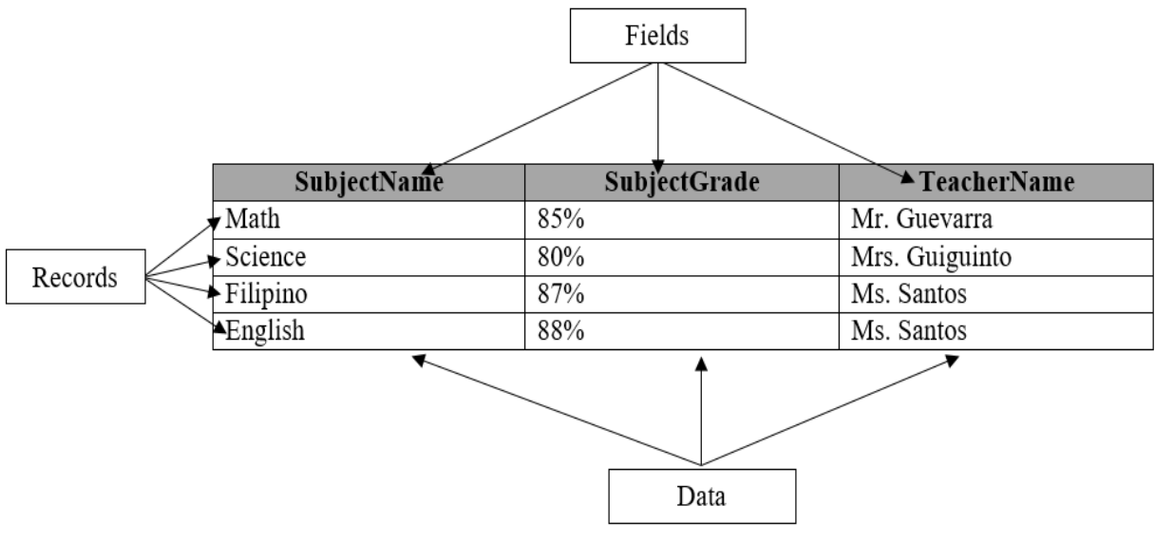
So, instead of creating a field that contains a list of data values or creating similar fields to hold other data, separate these fields from the first and make another list for subject names, their respective teachers, and the grades the students got for each subject as shown in the table below.

| First List | Second List |
|---|---|
| FirstName | SubjectName |
| LastName | SubjectGrade |
| MiddleName | TeacherName |
| StAddress | |
| City | |
| PhoneNumber | |
| ~~TeacherName~~ | |
| ~~SubjectTaught~~ | |
| ~~SubjectName~~ | |
| ~~SubjectGrade~~ | |
| AdviserName | |
| Average | |
| Level | |
| Section | |
| Birthdate | |
| Age | |

Observe the fields that were crossed out. These fields were transferred under a new list of fields

In this setup, your data entry for the second list should look like on the next figure below.

See how much easier it is if you define your fields properly? Now, you can find out what grade a student got in each subject and who his/her teacher was, not to mention how much more easily you can add more subjects as you wish, along with their corresponding teachers and student grade!




**REVISED KNOWLEDGE: Actual answer to the process questions/ focus questions**

1. **What is a database design?**
   Database design is defined as a collection of steps that help with designing, creating, implementing, and maintaining a business's data management systems. The main purpose of designing a database is to produce physical and logical models of designs for the proposed database system.

2. **What is a good database?**
   A good database design process is governed by specific rules. The first rule dictates that redundant data must be avoided; as it wastes space and increases the probability of faults and discrepancies within the database. The next rule is that the accuracy and comprehensiveness of information is extremely imperative. If the database contains erroneous information, any documents that fetch data from such a database will also include inaccurate information. Consequently, any decisions based on those documents will be misleading, thus, increasing the importance of a database design that caters to all of the above rules.

3. **What are the phases of database design?**
   Data definition – in this phase, all the data and information needed for the database are collected and determined.
   Data refinement – in this phase, the data and information you gathered are then refined by adding necessary data or eliminating unnecessary data
   Establishing relationships – in this phase, relationships or links between the information you gathered from the first two phases are established.

**FINAL KNOWLEDGE: Generalization/ Synthesis/ Summary**

A properly designed database provides you with access to up-to-date, accurate information. Because a correct design is essential to achieving your goals in working with a database, investing the time required to learn the principles of good design makes sense. In the end, you are much more likely to end up with a database that meets your needs and can easily accommodate change.

# DBES LEARNING ACTIVITY SHEET/GAWAING PAGKATUTO
## COMPUTER TECHNOLOGY 9
**First Quarter - Module 3**

_____
*Subject Teacher*

**Name of Learner:** _____ _____ _____ **SCORE:**
***Lastname*** ***Firstname*** ***MI***
**Section :** _____ **Date Completed:** _____

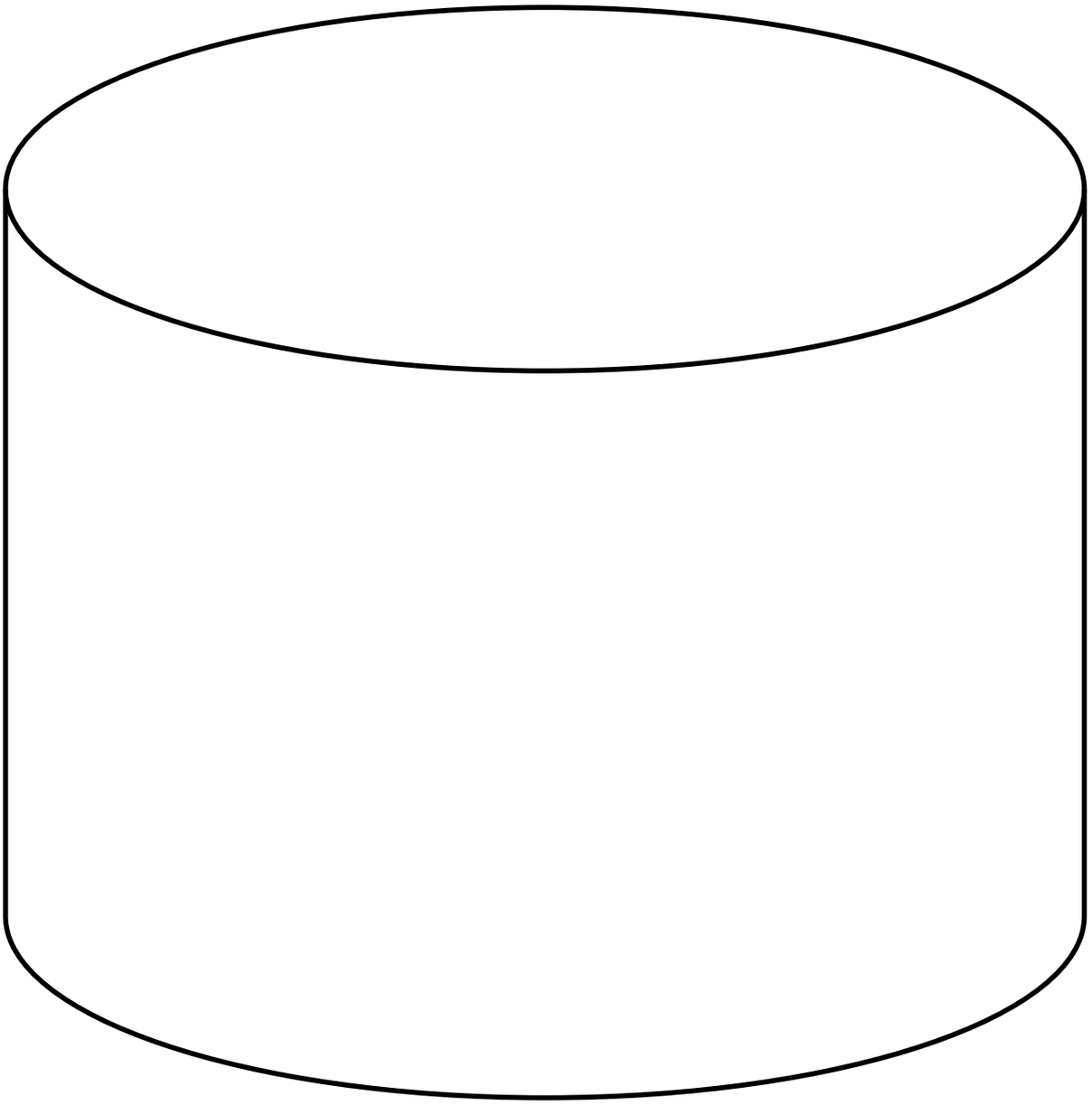
**A. Mini Task. Create a flat file database. Follow the instructions below.**

1. Buy 20 pieces of candies with different color, taste, size and brand.
2. List down the specification of each candy in the table below.
3. Candies can be repeated as long as they differ in color or taste.

| Brand/Name | Color | Taste | Weight | Manufacturer | |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

**B. Reflection**

**1. If database is a kind of storage, how will you explain to a nine (9) year-old kid playing with his toys the meaning of a database? Write you answer inside the container.**



| MODULE 4 – 5: UNDERSTANDING THE STAGES OF DATABASE DESIGN | |
|---|---|
| Content Standard | The learners demonstrate an understanding of key concepts, underlying principles and core competencies in database. |
| Performance Standard | The learners shall be able to independently create/provide quality product database using Microsoft access. |
| 21st Century Learning Skills | Critical Thinking-and-Doing, Computing/ ICT Literacy, Career and Self-reliance. |
| Core Values | Diligence and accuracy |

**REFERENCES:** *(Please be guided with the given references to help you perform the given activities. Click the given links and hyperlinks to access the suggested learning resources.)*

*A. Printed:*

- Bombase, Lilibeth S., et. al. My Computer Microsoft Access XP, Quezon City, Philippines: ABIVA Publishing House Incorporated

*B.* Online Source:

- Rungta, K. (2021, August 8). Database Design in DBMS Tutorial: Learn Data Modeling. Https://Www.Guru99.Com/Database-Design.Html. https://www.guru99.com/database-design.html


## TOPIC: UNDERSTANDING THE PHASES OF DATABASE DESIGN

### Introduction

**Database Design** is a collection of processes that facilitate the designing, development, implementation and maintenance of enterprise data management systems. Properly designed database is easy to maintain, improves data consistency and are cost effective in terms of disk storage space. The database designer decides how the data elements correlate and what data must be stored.

The main objectives of database design in DBMS are to produce logical and physical designs models of the proposed database system.

The logical model concentrates on the data requirements and the data to be stored independent of physical considerations. It does not concern itself with how the data will be stored or where it will be stored physically.

**Below are the Learning Targets/ Specific Objectives:**

- Define the data needed in the database
- Establish relationship in between tables
- Identify the purpose of their desired database system.
- Determine the field name of their database.
- Perform the field definition principle

**Exploration of Prior Knowledge**

**Directions**: Fill in the K-W-H-L Chart below to assess your prior knowledge and understanding of the topic, **Understanding the Phases of Database Design**

| What I Know | What I Want to Find Out | How I Can Learn More | What I Have Learned |
|---|---|---|---|
| | | | |
| **Skills I expect to use:** | | | |

## PROCESS QUESTIONS/ FOCUS QUESTIONS:

**Below are the key guide questions that you should remember as you perform all the activities in this lesson. You should be able to answer them at the end of the week.**

1. **What is being done in the first phase?**
2. **How is the second phase created?**
3. **What are the stages in database design?**

## SHORT EXERCISES/DRILLS

1. **What are the entities of a car dealer?**
   - _____
   - _____
   - _____
2. **If you are looking for a bookstore what are the entities that you need to consider?**
   - _____
   - _____
   - _____

## CONTENT DISCUSSION:

Your existing system can no longer cope. It's time to move on. Perhaps the existing paper system is generating too many errors, or the old Perl script based on flat files can no longer handle the load. Or perhaps an existing news database is struggling under its own popularity and needs an upgrade. This is the stage where the existing system is reviewed.

Depending on the size of the project, the designer may be an individual, responsible for the database implementation and coding, or may be a whole team of analysts. For now, the term *designer* will represent all these possibilities. Careful planning is very important in effective database management. This stage involves identifying the data, accessing and processing the data to produce the needed information, data printing to have a clear picture of the information.

## DATA DEFINITION

Data Definition – stage where all the necessary fields for the database are gathered and listed. All the data that need to be stored in the database must be considered. These fields are useful in manipulating the data.

Entity is the term given to the category of information in database.

- A **car dealer** would be interested in entities such as clients, cars and salesmen.
- A **school** would be interested in students, faculty and classes.
- A **bookstore** would be interested in books, authors, and publishers.
- A **mall** would be interested in customers, stores, branches and contractors.
- A **library** would be interested in students, transactions and books.

Each category in every entity will be used as table so as to assign fields.

Here is an example, the library system; the most likely fields needed for the tables are shown below:

**STUDENTS TABLE**

Student ID
Full Name
Home Phone
City
Year Level
Section

**TRANSACTIONS TABLE**

Borrower ID
Full Name
Home Phone
City
Yr Level & Sec
Book Number
Title
Category
Date Borrowed
Due date
Date returned

**BOOKS TABLE**

Book ID
Title
Subject

*List of necessary fields for the database*

## DATA REFINEMENT

Data Refinement – in this stage some fields need to be broken down to specific fields. For example, the student's name or borrower's name is often split into three fields: FIRST NAME, MIDDLE INITIAL, and LAST NAME. Data are much easier to sort according to the LAST NAME. If there is a need to search for students who have "Santos" as their last name, you can simply specify the last name for the search criteria and a list of all students with "Santos" as their last name will be displayed. In this stage we need to remove the unnecessary data and add the necessary data.

Student ID
Full Name
~~Home Phone~~
City
Year Level
Section

First name
Middle initial
Last name

Broken down the field full name into First Name, Middle Initial and Last name

- Break down the fields to the smallest logical parts to facilitate the search for information in the database.


- Remove the home phone, city, year level and section, title and category in the table because they are not needed in the table.

## DETERMINE WHAT TABLES YOU WILL NEED FOR YOUR DATABASE

Now that you have already determined what fields you need, it is time to determine what tables you will need for your database. You already split your list into two. This means that you should have two tables in your database – one table for the student information and another for the students' subjects. Going back to your lists simply assingn a descriptive name for your tables as shown in the example below.

| Student Info | Student Subjects |
|---|---|
| First Name | Subject Name |
| Last Name | Subject Grade |
| Middle Name | Teacher Name |
| St Address | |
| City | |
| Phone Number | |
| Adviser Name | |
| Level | |
| Section | |
| Birthdate | |
| | |

Assign table names that describe the data or information to be stored.

When determining what tables, you need for your database, always carefully analyze the fields you created, keeping in mind its purpose and the outputs you want it to produce so you can determine how many tables you will need. For instance, if you need to add information on teachers, you may need to add another table to contain such information.

## DETERMINE WHICH TABLE EACH FIELD BELONGS TO

Once you have named the tables for your database, review the fields that you have for each and see if you need to add other fields or move some fields between tables. In the example, everything seems to be in order so you may proceed to the next step.

## IDENTIFY FIELDS THAT HAVE UNIQUE VALUES IN EACH RECORD

For a database to be efficient, it must not contain useless and redundant records that only use up database space. This is where relational databases like MS Access come in handy in terms of avoiding erroneous data.

To avoid redundant records, you must identify unique fields that will make each record one of a kind. Unique fields are fields that contain data that cannot be repeated in any table record. For

instance, if you already have a record for a John Smith, that record must exist only once in the table. This does not mean, however, that the table cannot contain another John Smith who is a totally different person. We know that some people may have the same first name and/or last name; hence the need for name fields like Middle Name in case a first and/or last name is not unique. Now, proceed in identifying a unique field for your list.

| Student Info | Student Subjects |
|---|---|
| **Student ID** | Subject Name |
| First Name | Subject Grade |
| Last Name | Teacher Name |
| Middle Name | |
| StAddress | |
| City | |
| Phone Number | |
| Adviser Name | |
| Level | |
| Section | |
| Birthdate | |

Notice that the new Student ID field in your list. Since none of the fields before was unique, we need to add a new field that will serve as the table's primary key. Since the Student Info table contains student records, it should contain a unique field to avoid a student's record from being repeated.

Now look closely at the Student Subjects table below. Notice that it was not necessary to add a unique field to it because unlike the Student Info table, more than one record under the Student Subjects can pertain to a single student in the Student Info table.

StudentInfo Table

| StudentID |
|---|
| SD-00001 |
| SD-00002 |

StudentSubjects Table

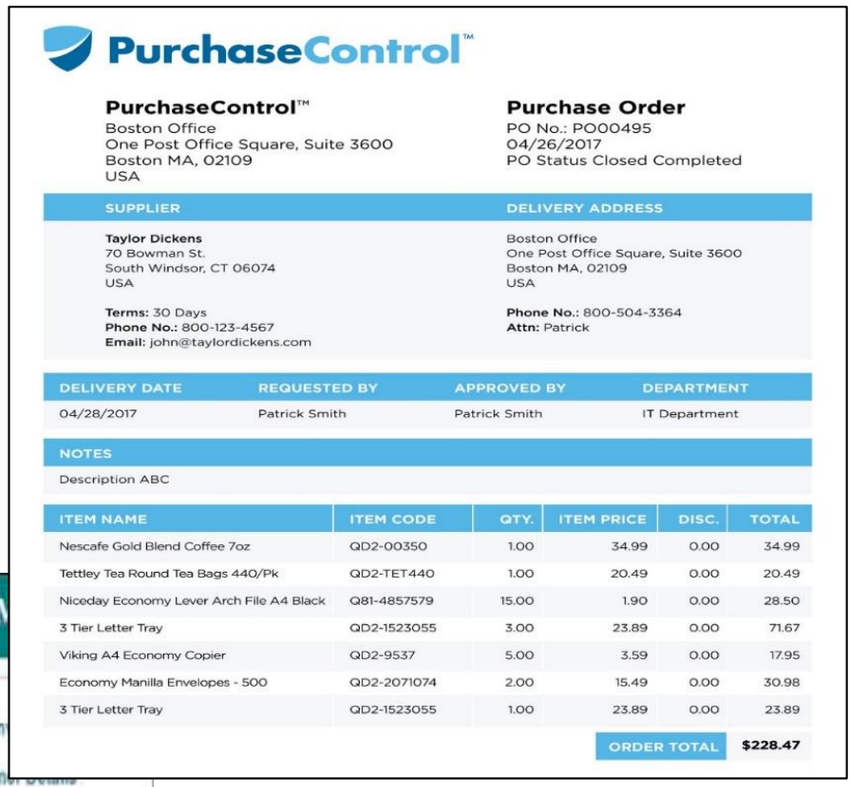
| SubjectName | TeacherName | SubjectGrade |
|---|---|---|
| Math | Mr. Guevarra | 85% |
| Science | Mrs. Guiguinto | 80% |
| English | Ms. Santos | 88% |
| Filipino | Mr. Guevara | 87% |
| Math | Mrs. Guiguinto | 84% |
| Science | Ms. Santos | 86% |
| English | Ms. Santos | 89% |
| Filipino | Ms. Santos | 88% |

In the figure above, notice that each student in the first table has four subjects each in the second table. Therefore, a unique field is not needed in the second table.

## DETERMINATION OF APPROPRIATE DATA OUTPUTS AND CREATION OF CONTROL SYSTEM

**Determination of Appropriate Data Outputs** – after the three stages, you must also consider ways in which information will be extracted from the database. The first step is to design how reports should look like on your computer monitor or when printed on paper. The information that appear on the reports generated by the database should help you determine which fields should be included in your tables.

**Creation of the Control System** – this is where you program the operations of the system to store, sort, change, and combine the data according to the predetermined relationships of phase three. You also need to consider the right application software to build your database.



PurchaseControl™

**PurchaseControl™**
Boston Office
One Post Office Square, Suite 3600
Boston MA, 02109
USA

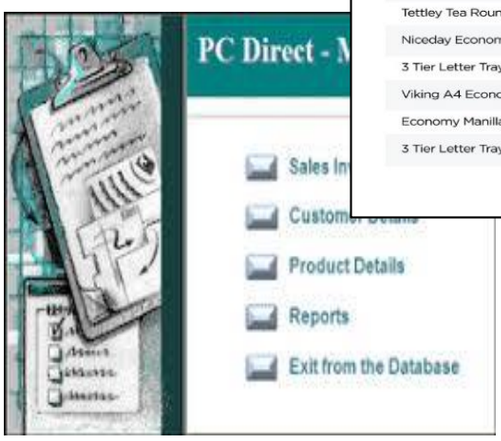
**Purchase Order**
PO No.: PO00495
04/26/2017
PO Status Closed Completed

| SUPPLIER | DELIVERY ADDRESS |
|---|---|
| **Taylor Dickens**<br>70 Bowman St.<br>South Windsor, CT 06074<br>USA<br><br>**Terms:** 30 Days<br>**Phone No.:** 800-123-4567<br>**Email:** john@taylordickens.com | Boston Office<br>One Post Office Square, Suite 3600<br>Boston MA, 02109<br>USA<br><br>**Phone No.:** 800-504-3364<br>**Attn:** Patrick |

| DELIVERY DATE | REQUESTED BY | APPROVED BY | DEPARTMENT |
|---|---|---|---|
| 04/28/2017 | Patrick Smith | Patrick Smith | IT Department |

NOTES
Description ABC

| ITEM NAME | ITEM CODE | QTY. | ITEM PRICE | DISC. | TOTAL |
|---|---|---|---|---|---|
| Nescafe Gold Blend Coffee 7oz | QD2-00350 | 1.00 | 34.99 | 0.00 | 34.99 |
| Tettley Tea Round Tea Bags 440/Pk | QD2-TET440 | 1.00 | 20.49 | 0.00 | 20.49 |
| Niceday Economy Lever Arch File A4 Black | Q81-4857579 | 15.00 | 1.90 | 0.00 | 28.50 |
| 3 Tier Letter Tray | QD2-1523055 | 3.00 | 23.89 | 0.00 | 71.67 |
| Viking A4 Economy Copier | QD2-9537 | 5.00 | 3.59 | 0.00 | 17.95 |
| Economy Manilla Envelopes - 500 | QD2-2071074 | 2.00 | 15.49 | 0.00 | 30.98 |
| 3 Tier Letter Tray | QD2-1523055 | 1.00 | 23.89 | 0.00 | 23.89 |
| | | | ORDER TOTAL | | $228.47 |




**REVISED KNOWLEDGE: Actual answer to the process questions/ focus questions**

1. **What is being done in the first phase?**
   Data Definition – stage where all the necessary fields for the database are gathered and listed. All the data that need to be stored in the database must be considered. These fields are useful in manipulating the data.

2. **How is the second phase created?**
   Data Refinement – in this stage some fields need to be broken down to specific fields. For example, the student's name or borrower's name is often split into three fields: FIRST NAME, MIDDLE INITIAL, and LAST NAME.

3. **What are the stages in database design?**
   - Data definition
   - Data refinement
   - Determine what tables you will need for your database
   - Determine which table each field belongs to
   - Identify fields that have unique values in each record

**FINAL KNOWLEDGE: Generalization/ Synthesis/ Summary**

Database design is an iterative process, which has a starting point and an almost endless procession of refinements. They should be viewed as learning processes. As the designers come to understand the workings of the enterprise and the meanings of its data, and express that understanding in the selected models, the information gained may well necessitate changes to other parts of the design.

# DBES LEARNING ACTIVITY SHEET/GAWAING PAGKATUTO
# COMPUTER TECHNOLOGY 9
**First Quarter - Module 4 - 5**

**MRS. MARY JANE P. DOLLENTE**
*Subject Teacher*

**Name of Learner:** _____ _____ _____ **SCORE:**
*Lastname* *Firstname* *MI*
**Section :** _____ **Date Completed:** _____

**Mini Task.**

**Activity 1: Define the data/field needed in the table.**

Think of the fields needed in the database "SCHOOL" wherein the tables needed are STUDENT, FACULTY and CLASSES. Write the fields that you think are useful to each table.

| 1. Student | 2. Faculty | 3. Classes |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |

Think of the fields needed in the database "LIBRARY" wherein the tables needed are BOOKS, BORROWER, and AUTHOR. Write the fields that you think are useful to each table.

| 4. Books | 5. Borrower | 6. Author |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |

**Activity 2: Data Refinement**

Break down each given field below to allow a database to meet the purposes shown in the following table. Fill out the right column with data related to the major field in the left column.

| Field | Field break down |
|---|---|
| **Last Name** | **1.** |
| | **2.** |
| | **3.** |
| **Birthday** | **4.** |
| | **5.** |
| | **6.** |
| | **7.** |
| **Book** | **8.** |
| | **9.** |
| | **10.** |
| **Teacher** | **11.** |
| | **12.** |
| | **13.** |

**PERFORMANCE TASK.**

**Activity 1.** Examine carefully the fields on the list below. The fields have already been broken down into smaller ones but are still in one list. Determine how the lists can be arranged into two or more tables based on the given information.

A. Supplier's and Customers' Lists
   1. There can be more than one contact person for each supplier
   2. There can only be one record for every supplier
   3. A customer can have more than one contact number such as one cellular phone number and one landline number.
   4. There can only be one record for each customer.

| | |
|---|---|
| SupplierID | CustomerID |
| SupplierNAme | FirstName |
| ContactFirstName | ContactType |
| ContactLastName | LastName |
| ContactPosition | MiddleName |
| Department | StAddress |
| ContactNumber | ContactNumber |
| StAddress | City |
| City | |

Write down the table names and their corresponding fields in the table below.

| | | | |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

B. Authors' List

1. There should only be one record for each author.
2. There should only be one record for each book.
3. There should only be one record for each publisher.
4. An author can write many books.
5. A publisher can publish many books.
6. There can only be one author for each book.

| | |
|---|---|
| AuthorID | AuthorCityAddress |
| BookCode | PublisherCityAddress |
| PublisherID | BookTitle |
| AuthorFirstName | BookGenre |
| AuthorMiddleName | BookSummary |
| AuthorLastName | PublisherPhoneNumber |
| AuthorStAddress | AuthorPhoneNumber |
| PublisherStAddress | |

Write down the table names and their corresponding field on the table below.

| | | | |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

**Rubric for Checking:**

| Criterion | 5 | 4 | 3 | 2 | Score |
|---|---|---|---|---|---|
| Content | The content of the Data Refinement is complete | The content of the Data Refinement is somewhat complete | Some of the content of the Data Refinement is missing | Several content of the Data Refinement is missing. | |
| Organization | The Data Refinement is organized, clean and well presented. | The Data Refinement is somewhat organized, clean and well presented. | The organization, cleanliness of the Data Refinement is acceptable | The organization, cleanliness and presentation of the Data Refinement is not acceptable. | |
| Efficiency | The student finished the Data Refinement in less than the given time limit. | The student is somewhat finished the Data Refinement within the given time limit. | The student is able to finished the Data Refinement on the last minute of the given time limit | The student did not finished the Data Refinement within the given time limit. | |

**Sample Output:**

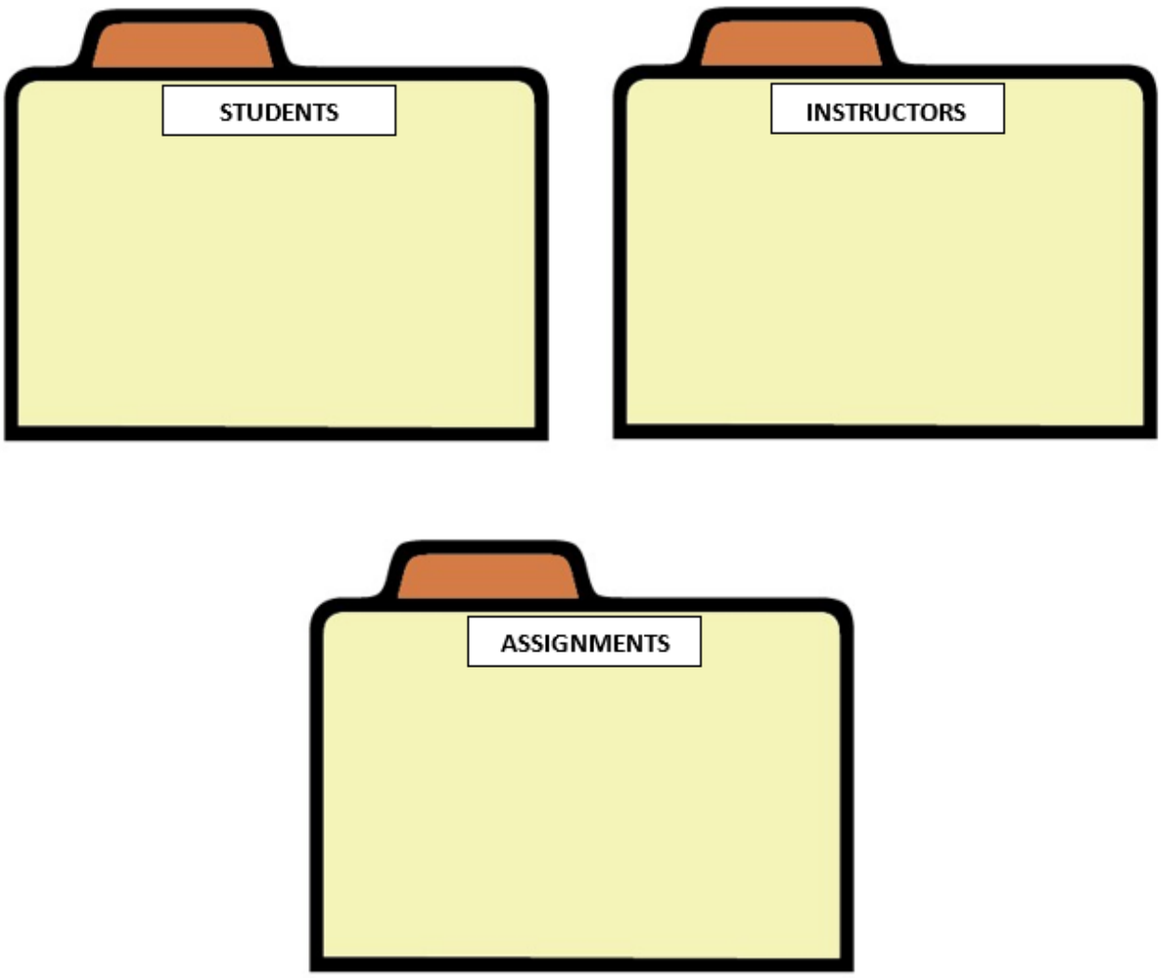
| | |
|---|---|
| FirstName | FourthGrading |
| SubjectGrade | ThirdGrading |
| FirstGrading | SubjectName |
| SecondGrading | MiddleInitial |
| SubjectTeacher | LastName |

| StudentInfo | SubjectInfo | Subject Grades |
|---|---|---|
| LastName | SubjectName | FirstGrading |
| FirstName | SubjectTeacher | SecondGrading |
| MiddleInitial | SubjectGrade | ThirdGrading |
| | | FourthGrading |

**Activity 2:**

Scenario: your teacher assigned you to create a Classroom Database Management. As the first phase in database design, you have to define the fields necessary on each List or Table. Assign the given fields to the appropriate table.

| | | |
|---|---|---|
| Assignment ID | Student ID | Instructor ID |
| First Name | Class ID | Instructor |
| Last Name | Assignemnt Description | Parents' Name |
| Exam | Phone Number | Address |
| Percent of Grade | Maximum Points | Department |



| MODULE 6: RECOGNIZING THE QUALITIES OF A GOOD DATABASE | |
|---|---|
| Content Standard | The learners demonstrate an understanding of key concepts, underlying principles and core competencies in database. |
| Performance Standard | The learners shall be able to independently create/provide quality product database using Microsoft access. |
| 21st Century Learning Skills | Critical Thinking-and-Doing, Computing/ ICT Literacy, Career and Self-reliance. |
| Core Values | Diligence and accuracy |

**REFERENCES:** *(Please be guided with the given references to help you perform the given activities. Click the given links and hyperlinks to access the suggested learning resources.)*


A. *Printed:*
- Bombase, Lilibeth S., et. al. My Computer Microsoft Access XP, Quezon City, Philippines: ABIVA Publishing House Incorporated

B. *Online:*
- Open Textbooks. (n.d.). Siyavula. Retrieved August 9, 2021, from https://intl.siyavula.com/read/
- History of Electronic Database. (n.d.). Https://Www.Instantdb.Co.Uk/History.Html#:~:Text=Databases%20as%20we%20understand%20them,Airlines%20to%20manage%20reservation%20data. Retrieved August 9, 2021, from https://www.instantdb.co.uk/history.html#:%7E:text=Databases%20as%20we%20understand%20them,Airlines%20to%20manage%20reservation%20data./



## TOPIC: Recognizing the Qualities of A Good Database

### Introduction

Data is stored in tables in a database. It can be stored in a single table (called a flat database) or in multiple connected tables. Each table consists of fields and records. Fields are the categories that you want to record data for. For example, the music table shown above, contains fields like Title, Artist, Duration and Album. Records refer to the actual data being captured. With each record containing the data of a single item.

All good databases should begin with valuable metadata and data. The database should be strong enough to store all the relevant data and requirements. Should be able to relate the tables in the database by means of relation.

**Below are the Learning Targets/ Specific Objectives:**

- Understand the history of Electronic Database
- Recognize the qualities of a good database

**Exploration of Prior Knowledge**

**Directions**: Fill in the K-W-H-L Chart below to assess your prior knowledge and understanding of the topic, **Recognizing the Qualities of A Good Database.**

| What I Know | What I Want to Find Out | How I Can Learn More | What I Have Learned |
|---|---|---|---|
| | | | |
| **Skills I expect to use:** | | | |

**PROCESS QUESTIONS/ FOCUS QUESTIONS:**

**Below are the key guide questions that you should remember as you perform all the activities in this lesson. You should be able to answer them at the end of the week.**

1. **What are the qualities of a good database?**
2. **What are the Characteristics and Benefits of a Database?**

**SHORT EXERCISES/DRILLS**

**Activity 1:** Assessing your learning
Write **True** if the statement is correct and **False** if otherwise.

| | |
|---|---|
| | 1. The first electronic database was used in the 1870's. |
| | 2. In 1970, Dr. Edgar F. Codd published an academic paper on relational database. |
| | 3. Edgar's article inspired database transactions called Structured English Query Language. |
| | 4. dBase ran on an old Intel Computer. |
| | 5. SQL stands for Standard Query Language. |

## CONTENT DISCUSSION:

Information generated from a database is an asset, since it provides the user the necessary information for an informed decision. A businessman needs information to decide to decide which direction to take in his business endeavor by looking at sales, expenses, competition and products. In school, the teacher uses information generated from a database to determine the grades of students. A stock broker would use a database to get information to analyze market trends and company information to decide whether to sell or buy a stock for his clients.

## A BRIEF HISTORY OF ELECTRONIC DATABASE

Obviously, databases in one form or another were known to exist throughout the history, even in Ancient times. These were all manual databases either on paper, stone or other materials that people could write on. Databases as we understand them now first appeared in 1960's with the expansion of computers.

As said, it all started in 1960's when IBM introduced first database system that was widely accepted called SABRE. It is today known as a database that was first used at American Airlines to manage reservation data. In the 1970's E.F. Codd wrote a paper that revolutionise the way people think of databases. He was the first to separate database organisation with actual data. It became de facto standard and it is used to this day. In 1980's Structured Query Language or shortly SQL became the standard. Along with increasing popularity of personal computing, databases achieved huge commercial success. IBM was again pioneer with the introduction of DB2. Other IT companies were active as well and products as PARADOX and WATCOM SQL were born as well. Those too were commercially successful. 1990's was the decade of Oracle. They took a nice share of market with their database tools. With the increasing use of Internet in the 1990's and 2000's database systems continued to grow at a steady pace and today we can hardly think of a life without them. They are all around us used in commercial, government and personal setting.

**Short History of Electronic Database Timeline**

- pre Computers Era: Mainly paper was used
- pre 1960: File based flat databases were used
- 1960: SABRE - IBM Database System used by American Airlines
- 1970: E.F.Codd published his database paper
- 1980: SQL became a standard
- 1980: IBM released DB2 which became huge success and a role model for others
- 1990: Electronic Databases became popular and were used at home as well utilising Excel/Access and ODBC
- 1990-2000: With the growing use of Internet and online marketing, databases became the foundation for web sites and web applications. Oracle Developer software was huge success. XML was introduced.

## RECOGNIZING THE QUALITIES OF A GOOD DATABASE

As outlined above having the right information would lessen the chances of failure in any endeavor; therefore, it is important that information generated from a database system will have the following characteristics.

1. Accurate information – means the reports generated from the database program are error free.
2. Organized information – means the reports maybe arranged to suit the need of the decision maker.
3. Timely information – means the report should be useful at the moment it is generated.
4. Verifiable information - means the report can be proven to be correct or incorrect.
5. Accessible information – means the database report is available to the person who needs it at any time.
6. Economical information- means information generated from a database program should be cost-effective to produce. Generating reports from a database costs money and the means of sharing these to the right people in the organization is crucial in cutting cost to a minimum.
7. Useful information – means the report should have meaning to the people who get them. Information is important to a lot of people, and some information are only relevant to some.

**Characteristics and Benefits of a Database**

Managing information means taking care of it so that it works for us and is useful for the tasks we perform. By using a DBMS, the information we collect and add to its database is no longer subject to accidental disorganization. It becomes more accessible and integrated with the rest of our work. Managing information using a database allows us to become strategic users of the data we have.

We often need to access and re-sort data for various uses. These may include:

- Creating mailing lists
- Writing management reports
- Generating lists of selected news stories
- Identifying various client needs

The processing power of a database allows it to manipulate the data it houses, so it can:

- Sort
- Match
- Link
- Aggregate
- Skip fields
- Calculate
- Arrange

Because of the versatility of databases, we find them powering all sorts of projects. A database can be linked to:

- A website that is capturing registered users
- A client-tracking application for social service organizations
- A medical record system for a health care facility
- Your personal address book in your email client
- A collection of word-processed documents
- A system that issues airline reservations

**Characteristics and Benefits of a Database**

There are a number of characteristics that distinguish the database approach from the file-based system or approach. This chapter describes the benefits (and features) of the database system.

**Self-describing nature of a database system -** A database system is referred to as self-describing because it not only contains the database itself, but also metadata which defines and describes the data and relationships between tables in the database. This information is used by the DBMS software or database users if needed. This separation of data and information about the data makes a database system totally different from the traditional file-based system in which the data definition is part of the application programs.

**Insulation between program and data -** In the file-based system, the structure of the data files is defined in the application programs so if a user wants to change the structure of a file, all the programs that access that file might need to be changed as well.
On the other hand, in the database approach, the data structure is stored in the system catalogue and not in the programs. Therefore, one change is all that is needed to change the structure of a file. This insulation between the programs and data is also called program-data independence.

**Support for multiple views of data -** A database supports multiple views of data. A view is a subset of the database, which is defined and dedicated for particular users of the system. Multiple users in the system might have different views of the system. Each view might contain only the data of interest to a user or group of users.

**Sharing of data and multiuser system -** Current database systems are designed for multiple users. That is, they allow many users to access the same database at the same time. This access is achieved through features called concurrency control strategies. These strategies ensure that the data accessed are always correct and that data integrity is maintained. The design of modern multiuser database systems is a great improvement from those in the past which restricted usage to one person at a time.

**Control of data redundancy** - In the database approach, ideally, each data item is stored in only one place in the database. In some cases, data redundancy still exists to improve system performance, but such redundancy is controlled by application programming and kept to minimum by introducing as little redundancy as possible when designing the database.

**Data sharing** - The integration of all the data, for an organization, within a database system has many advantages. First, it allows for data sharing among employees and others who have access to the system. Second, it gives users the ability to generate more information from a given amount of data than would be possible without the integration.

**Enforcement of integrity constraints** - Database management systems must provide the ability to define and enforce certain constraints to ensure that users enter valid information and maintain data integrity. A database constraint is a restriction or rule that dictates what can be entered or edited in a table such as a postal code using a certain format or adding a valid city in the City field.

There are many types of database constraints. Data type, for example, determines the sort of data permitted in a field, for example numbers only. Data uniqueness such as the primary key ensures that no duplicates are entered. Constraints can be simple (field based) or complex (programming).

**Restriction of unauthorized access** - Not all users of a database system will have the same accessing privileges. For example, one user might have read-only access (i.e., the ability to read a file but not make changes), while another might have read and write privileges, which is the ability to both read and modify a file. For this reason, a database management system should provide a security subsystem to create and control different types of user accounts and restrict unauthorized access.

**Data independence** - Another advantage of a database management system is how it allows for data independence. In other words, the system data descriptions or data describing data (metadata) are separated from the application programs. This is possible because changes to the data structure are handled by the database management system and are not embedded in the program itself.

**Transaction processing** - A database management system must include concurrency control subsystems. This feature ensures that data remains consistent and valid during transaction processing even if several users update the same information.

**Provision for multiple views of data** - By its very nature, a DBMS permits many users to have access to its database either individually or simultaneously. It is not important for users to be aware of how and where the data they access is stored

**Backup and recovery facilities** - Backup and recovery are methods that allow you to protect your data from loss. The database system provides a separate process, from that of a network backup, for backing up and recovering data. If a hard drive fails and the database stored on the hard drive is not accessible, the only way to recover the database is from a backup.
If a computer system fails in the middle of a complex update process, the recovery subsystem is responsible for making sure that the database is restored to its original state. These are two more benefits of a database management system.

- **KEY TERMS**
- concurrency control strategies: features of a database that allow several users access to the same data item at the same time
- data type: determines the sort of data permitted in a field, for example numbers only
- data uniqueness: ensures that no duplicates are entered
- database constraint: a restriction that determines what is allowed to be entered or edited in a table
- metadata: defines and describes the data and relationships between tables in the database
- read and write privileges: the ability to both read and modify a file
- read-only access: the ability to read a file but not make changes

- self-describing: a database system is referred to as self-describing because it not only contains the database itself, but also metadata which defines and describes the data and relationships between tables in the database

**REVISED KNOWLEDGE: Actual answer to the process questions/ focus questions**

1. **What are the qualities of a good database?**

    1. Accurate information – means the reports generated from the database program are error free.
    2. Organized information – means the reports maybe arranged to suit the need of the decision maker.
    3. Timely information – means the report should be useful at the moment it is generated.
    4. Verifiable information - means the report can be proven to be correct or incorrect.
    5. Accessible information – means the database report is available to the person who needs it at any time.
    6. Economical information- means information generated from a database program should be cost-effective to produce. Generating reports from a database costs money and the means of sharing these to the right people in the organization is crucial in cutting cost to a minimum.
    7. Useful information – means the report should have meaning to the people who get them. Information is important to a lot of people, and some information are only relevant to some.

2. **What are the Characteristics and Benefits of a Database?**
    - Self-describing nature of a database system
    - Insulation between program and data
    - Support for multiple views of data
    - Sharing of data and multiuser system
    - Control of data redundancy
    - Data sharing
    - Enforcement of integrity constraints
    - Restriction of unauthorized access
    - Data independence
    - Transaction processing
    - Provision for multiple views of data
    - Backup and recovery facilities

**FINAL KNOWLEDGE: Generalization/ Synthesis/ Summary**

Managing information means taking care of it so that it works for us and is useful for the tasks we perform. By using a DBMS, the information we collect and add to its database is no longer subject to accidental disorganization. It becomes more accessible and integrated with the rest of our work. Managing information using a database allows us to become strategic users of the data we have.

# DBES LEARNING ACTIVITY SHEET/GAWAING PAGKATUTO
# COMPUTER TECHNOLOGY 9
**First Quarter - Module 6**

_____
*Subject Teacher*

| | |
|---|---|
| **Name of Learner:** _____ _____ _____<br>***Lastname*** ***Firstname*** ***MI***<br>**Section :** _____ **Date Completed:** _____ | **SCORE:** |

**A. Written Work/s:**

Activity 1: Determine what quality of a good database is being emphasized in the given situations below.

_____ 1. Government agencies in many countries can verify identification of their citizens and confirm whether the passport or identification card presented is really that person or not.

_____ 2. The people and the commission on election officials would have access to the result of the electoral process so that the winner would be declared immediately.

_____ 3. The government would decide to build more power plants in a certain period of time upon analyzing a recent report of the increasing trend of electrical power use in the next years.

_____ 4. For example, your grade is important to you and to your parents but no to others who do not know you.

**Activity 2. Answer the questions briefly but substantially.**

1. How is a DBMS distinguished from a file-based system?

2. What is data independence and why is it important?

3. What is the purpose of managing information?

4. Discuss the uses of databases in a business environment.

| MODULE 7: ESTABLISHING RELATIONSHIPS IN BETWEEN TABLES | |
|---|---|
| Content Standard | The learners demonstrate an understanding of key concepts, underlying principles and core competencies in database. |
| Performance Standard | The learners shall be able to independently create/provide quality product database using Microsoft access. |
| 21st Century Learning Skills | Critical Thinking-and-Doing, Computing/ ICT Literacy, Career and Self-reliance. |
| Core Values | Diligence and accuracy |

**REFERENCES:** *(Please be guided with the given references to help you perform the given activities. Click the given links and hyperlinks to access the suggested learning resources.)*

*A. Printed:*

- Bombase, Lilibeth S., et. al. My Computer Microsoft Access XP, Quezon City, Philippines: ABIVA Publishing House Incorporated

*B. Online:*

- Database Relationships. (n.d.) Https://Condor.Depaul.Edu/Gandrus/240IT/Accesspages/Relationships.Htm. Retrieved August 9, 2021, from https://condor.depaul.edu/gandrus/240IT/accesspages/relationships.htm


## TOPIC: Establishing Relationships In Between Tables

### Introduction

In a relational database, relationships enable you to prevent redundant data. For example, if you are designing a database that will track information about books, you might have a table named "Titles" that stores information about each book, such as the book's title, date of publication, and publisher. There is also information that you might want to store about the publisher, such as the publisher's telephone number, address, and ZIP Code/Postal Code. If you were to store all this information in the "Titles" table, the publisher's telephone number would be duplicated for each title that the publisher prints.

A better solution is to store the publisher's information only one time, in a separate table that we will call "Publishers." You would then put a pointer in the "Titles" table that references an entry in the "Publishers" table.

To make sure that your data stays synchronized, you can enforce referential integrity between tables. Referential integrity relationships help make sure that information in one table matches information in another. For example, each title in the "Titles" table must be associated with a specific publisher in the "Publishers" table. A title cannot be added to the database for a publisher that does not exist in the database.

Logical relationships in a database enable you to efficiently query data and create reports.

**Below are the Learning Targets/ Specific Objectives:**

- Identify and discuss the types of database relationship.
- Distinguish and explain the differences and functions of database relationship.
- Explain when a database is said to be well-designed

**Exploration of Prior Knowledge**

**Directions**: Fill in the K-W-H-L Chart below to assess your prior knowledge and understanding of the topic, **Establishing Relationships In Between Tables**

| What I Know | What I Want to Find Out | How I Can Learn More | What I Have Learned |
|---|---|---|---|
| | | | |
| **Skills I expect to use:** | | | |

**PROCESS QUESTIONS/ FOCUS QUESTIONS:**

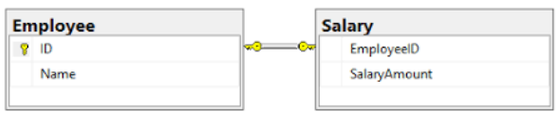
**Below is the key guide question that you should remember as you perform all the activities in this lesson. You should be able to answer them at the end of the week.**

1. **What are the kinds of database relationships?**

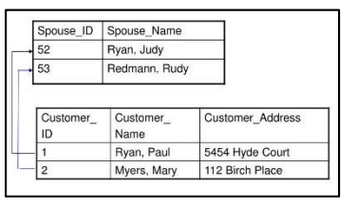
**SHORT EXERCISES/DRILLS**

Identify what kind of database relationship is shown below.

**1.**



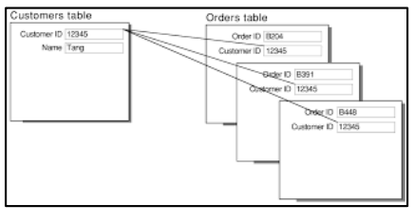

**2.**



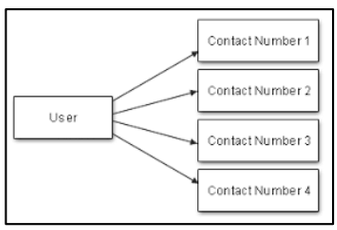
| Spouse_ID | Spouse_Name |
|---|---|
| 52 | Ryan, Judy |
| 53 | Redmann, Rudy |

| Customer_ ID | Customer_ Name | Customer_Address |
|---|---|---|
| 1 | Ryan, Paul | 5454 Hyde Court |
| 2 | Myers, Mary | 112 Birch Place |

**3.**




**4.**



**5.**

CUSTOMERS

| customer_id | customer_name |
|---|---|
| 101 | John Doe |
| 102 | Bruce Wayne |

ORDERS

| order_id | customer_id | order_date | amount |
|---|---|---|---|
| 555 | 101 | 12/24/09 | $156.78 |
| 556 | 102 | 12/25/09 | $99.99 |
| 557 | 101 | 12/26/09 | $75.00 |

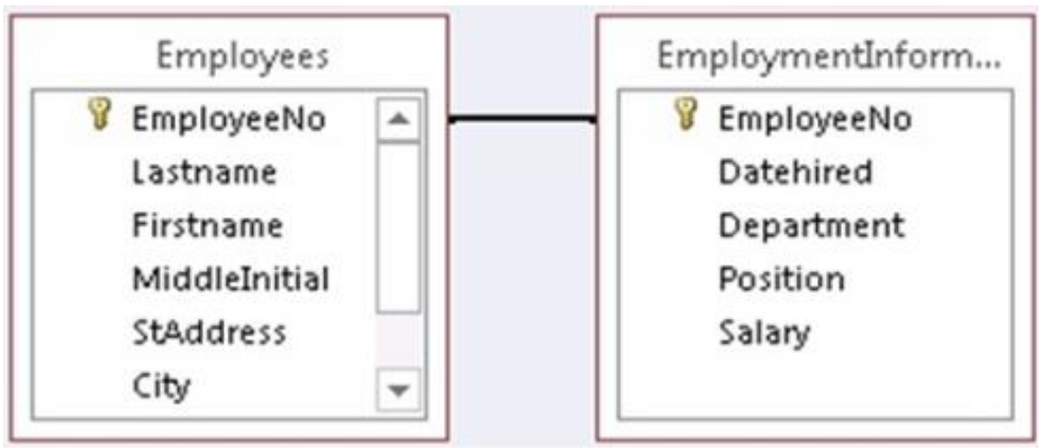
## CONTENT DISCUSSION:

A relationship is a link between two or among three or more tables in a database. This relationship allows you to simultaneously view or access related records from two or more tables. Before you start creating relationships between/among tables in your database, you first need to understand the types of relationships that you can use.

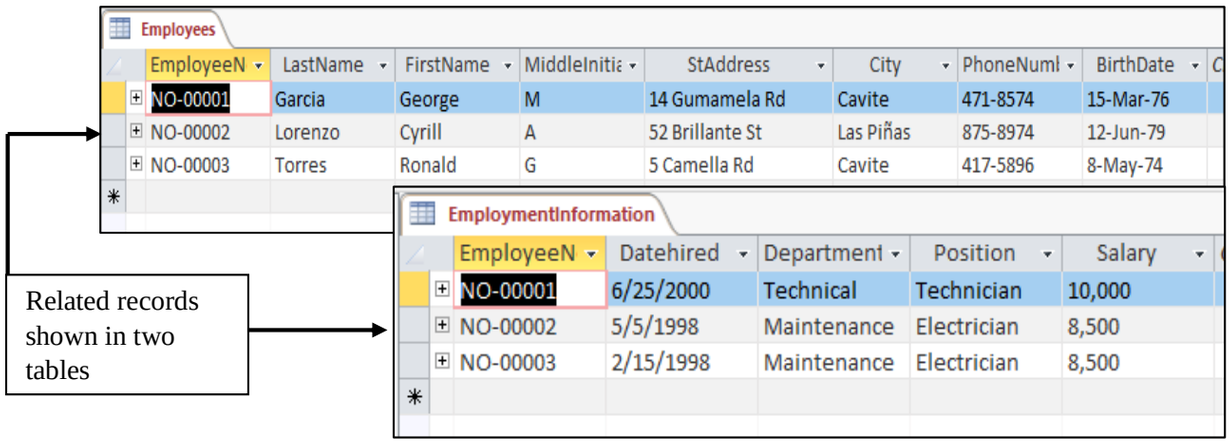
## ESTABLISHING RELATIONSHIPS

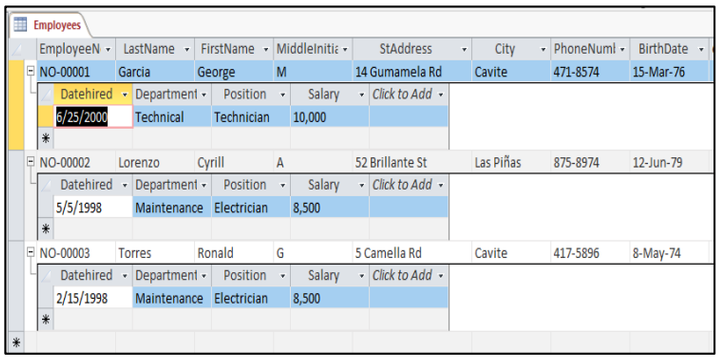
### Kinds of relationships

- **One to one relationship –** this relationship means that a record in the first table can only have one related record in the other table. This means that both tables must have a unique field to avoid repetitive record entries. This type of relationship is seldom used when creating a database. Most of the time, this relationship separates one table into two tables when there are too many fields. For instance, if you have 20 fields in a table, you may consider putting the field in two tables to help improve a user's speed in searching through records because the number of fields that MS Access needs to search through has been reduced.




The tables below show the sample data that illustrate how the one to one relationship works. The first figure shows the records in two tables. Notice that both tables have the Employee No field that is used to state the relationship between the tables. The other figure shows the related records from both tables in a sub datasheet.




A one-to-one (1:1) relationship means that each record in Table A relates to one, and only one, record in Table B, and each record in Table B relates to one, and only one, record in Table A. Look at the following example of tables from a company's Employees database:

| PERSONAL | | | | | | |
|---|---|---|---|---|---|---|
| EmployeeID | FirstName | LastName | Address | City | State | Zip |
| EN1-10 | Carol | Schaaf | 2306 Palisade Ave. | Union City | NJ | 07087 |
| EN1-12 | Gayle | Murray | 1855 Broadway | New York | NY | 12390 |
| EN1-15 | Steve | Baranco | 742 Forrest St. | Kearny | NJ | 07032 |
| EN1-16 | Kristine | Racich | 416 Bloomfield St. | Hoboken | NJ | 07030 |
| EN1-19 | Barbara | Zumbo | 24 Central Ave. | Ritchfield Park | NJ | 07660 |
| EN1-20 | Daniel | Gordon | 2 Angelique St. | Weehawken | NJ | 07087 |
| EN1-22 | Jacqueline | Rivet | 3600 Bergeline Ave. | Union City | NJ | 07087 |
| EN1-23 | Betsy | Rosyln | 1800 Boulevard East | Weehawken | NJ | 07086 |
| EN1-25 | Will | Strick | 2100 91st St. | North Bergen | NJ | 07047 |
| EN1-26 | Susan | Shipe | 240 Fifth Ave. | New York | NY | 10018 |

| PAYROLL | |
|---|---|
| EmployeeID | PayRate |
| EN1-10 | $25.00 |
| EN1-12 | $27.50 |

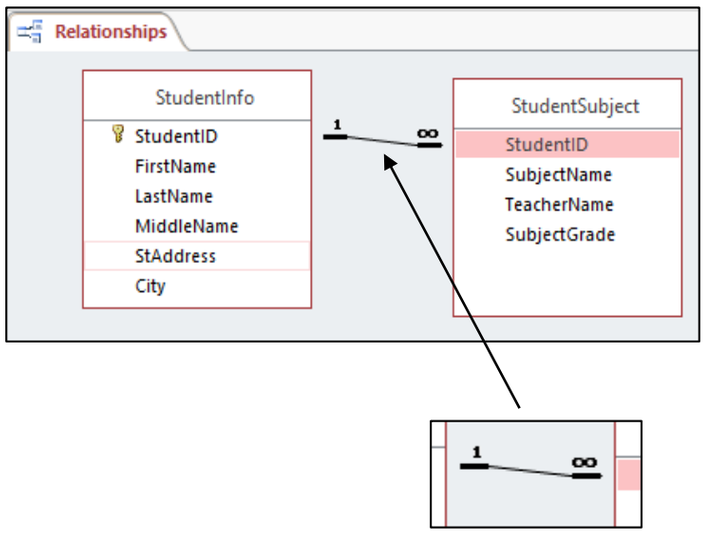
| | |
|---|---|
| EN1-15 | $20.00 |
| EN1-16 | $19.00 |
| EN1-19 | $22.75 |
| EN1-20 | $23.00 |
| EN1-22 | $22.50 |
| EN1-23 | $19.50 |
| EN1-25 | $12.50 |
| EN1-26 | $14.00 |

Above, tables with a one-to-one relationship from a database of information about employees

Each record in the Personal table is about one employee. That record relates to one, and only one, record in the Payroll table. Each record in the Payroll table relates to one, and only one, record in the Personal table. (This is what looking at it from both directions means.)

In a one-to-one relationship, either table can be considered to be the primary or parent table.

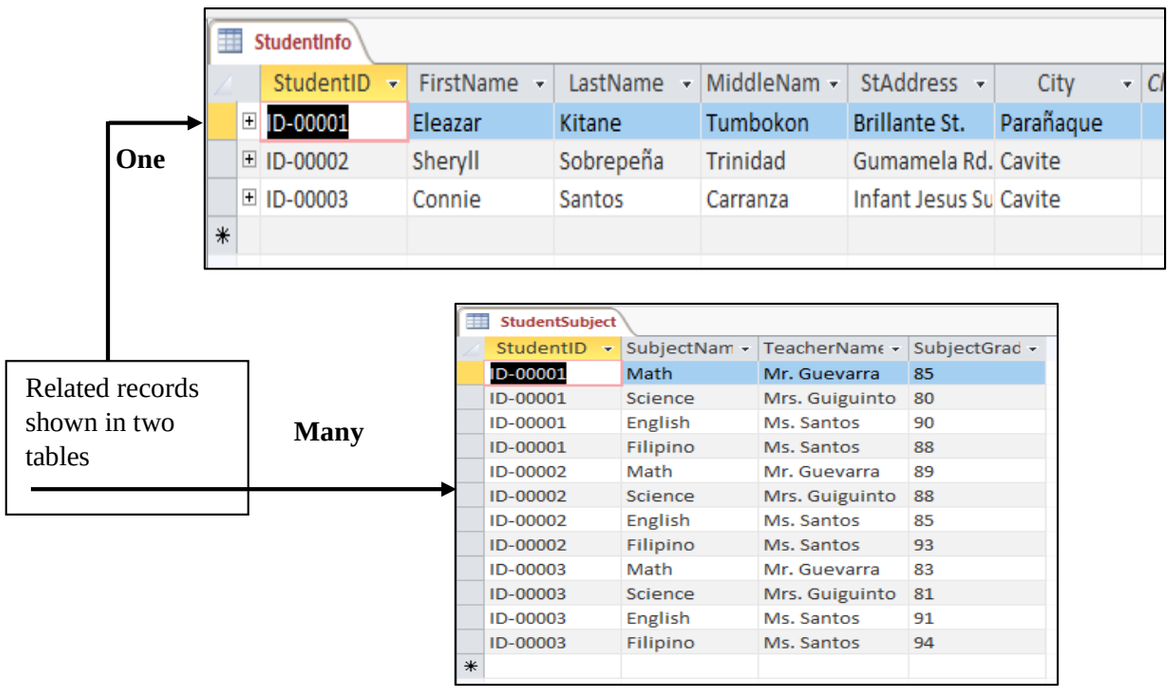
- **One to many relationship –** in this relationship, one record in the first table can have many related records in the other table. This means that only one of the tables will have a unique field. A good example would be the two tables in your design, the **Student Info** and **Student Subjects** tables.

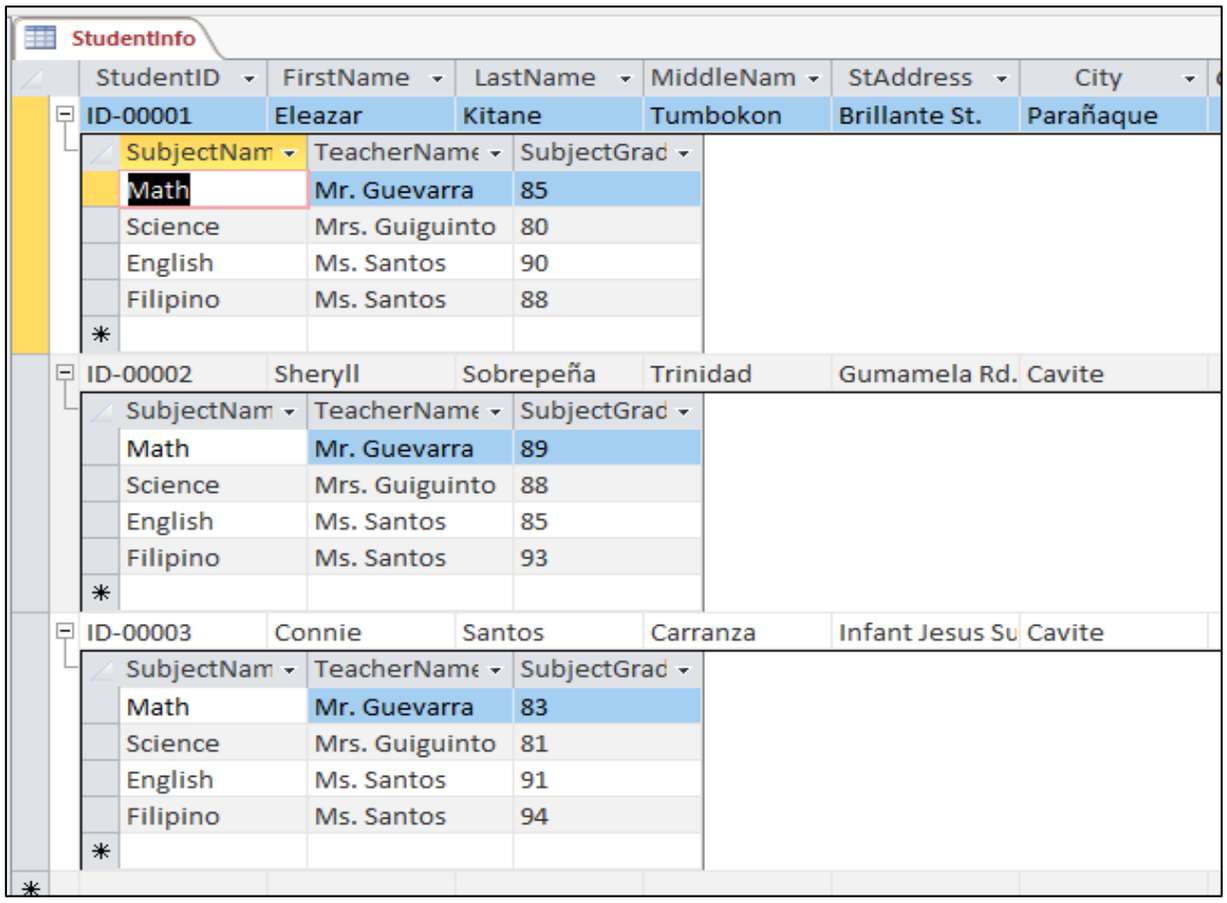



This figure, the line that represents the relationships between the tables show a (one side of the relationship) and an infinite symbol

The infinite symbol tells us that the table contains the many side of the relationships

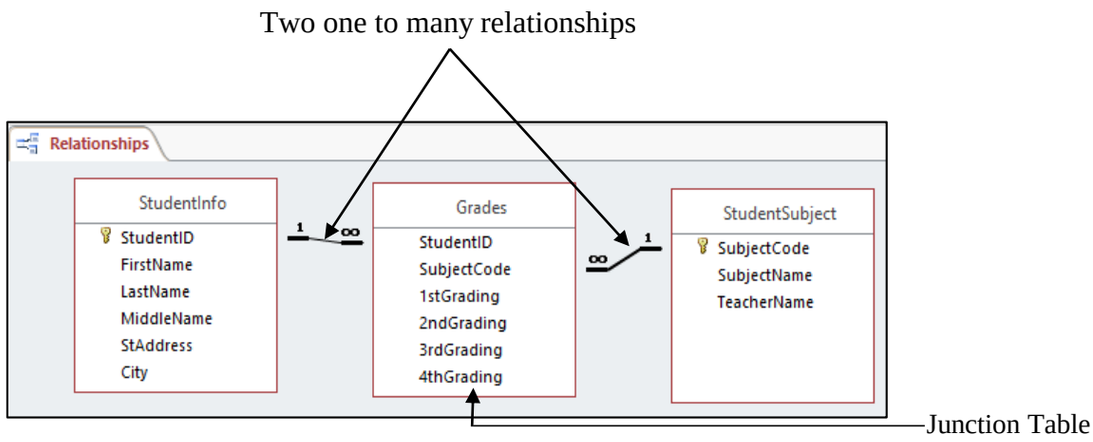
The infinite symbol on the many side of the relationship means that the table can have many records that are related to only one record in the first table. The following tables show how related records in the two tables will appear in a table with sub datasheet.




Related records shown in a table with a sub datasheet

- **Many to many relationships** – this relationship means that many records in one table can have many related records in another table. Unlike the first two types, this relationship cannot be directly defined between two tables because you will need a third table called the **junction table**. Basically, this type of relationship is formed by two one to many relationships that are both related to a third table as shown in the following figure.




In this relationship, the two tables are joined together by another table, the **junction table**. These two one-to-many relationships make up a many to many relationships.

The three tables below show records in a many-to-many relationship. Observe the Student ID and Subject Code fields. Notice that under both the Student Info and Subject Info tables, the Student ID and Subject Code fields only appear once in each record (unique fields) while in the Subject Grades table (Junction table), both fields contain repetitive records because the table represents the many side of the relationship.

StudentInfo

| StudentID | FirstName | LastName | MiddleNam | StAddress | City |
|---|---|---|---|---|---|
| ID-00001 | Eleazar | Kitane | Tumbokon | Brillante St. | Parañaque |
| ID-00002 | Sheryll | Sobrepeña | Trinidad | Gumamela Rd. | Cavite |
| ID-00003 | Connie | Santos | Carranza | Infant Jesus Su | Cavite |

One

Many

StudentSubject

| SubjectCode | SubjectNam | TeacherName |
|---|---|---|
| SUB-00001 | Math | Mr. Guevarra |
| SUB-00002 | Science | Mrs. Guiguinto |
| SUB-00003 | English | Ms. Santos |
| SUB-00004 | Filipino | Ms. Santos |

One

Many

Grades

| StudentID | SubjectCod | 1stGrading | 2ndGrading | 3rdGrading | 4thGrading |
|---|---|---|---|---|---|
| ID-00001 | SUB-00003 | 80 | 85 | 95 | 95 |
| ID-00001 | SUB-00004 | 85 | 83 | 89 | 88 |
| ID-00002 | SUB-00001 | 80 | 80 | 83 | 88 |
| ID-00002 | SUB-00004 | 85 | 85 | 90 | 91 |

Now, it is time to establish the relationship between the tables for your Students Database. You already have a unique field (Student ID) for the Student Info table. As mentioned earlier, this unique field means that the table will serve as the one side of the relationship. As for the Student Subjects table, a unique field is not necessary since it will contain more than one record (many side) that is

related to a single record in the first table. This also means that you should create one-to-many relationships.

**Common field**

To build a relationship, both tables should have a common field. A common field refers to a field in both tables that contains the same data or value. In this case, the Student Info table uses the Student ID field to uniquely identify a record. To connect the Student Subjects table to the Student Info table, you need to add a field to it that will contain the Student ID data. This means you need to modify the Student Subjects table shown below.

| StudentInfo | StudentSubject |
|---|---|
| **StudentID** | SubjectName |
| FirstName | SubjectGrade |
| LastName | TeacherName |
| MiddleName | **StudentID** |
| StAddress | |
| City | |
| PhoneNumber | |
| AdviserName | |
| Level | |
| Section | |
| Birthdate | |

Common Field

Both tables should now have the Student ID field as a common field. This tells MS Access that the records under the Student Subjects table whose Student ID fields under the Student Info table are related.

**REVISED KNOWLEDGE: Actual answer to the process questions/ focus questions**

1. **What are the kinds of database relationships?**
   - **One – to – one**
   - **One – to – many**
   - **Many – to – many**

**FINAL KNOWLEDGE: Generalization/ Synthesis/ Summary**

A relationship works by matching data in key columns, usually columns (or fields) that have the same name in both tables. In most cases, the relationship connects the primary key, or the unique identifier column for each row, from one table to a field in another table. The column in the other table is known as the "foreign key." For example, if you want to track sales of each book title, you create a relationship between the primary key column (let's call it title_ID) in the "Titles" table and a column in the "Sales" table that is named title_ID. The title_ID column in the "Sales" table is the foreign key.

# DBES LEARNING ACTIVITY SHEET/GAWAING PAGKATUTO
# COMPUTER TECHNOLOGY 9
**First Quarter - Module 7**

_____
*Subject Teacher*

**Name of Learner:** _____ _____ _____ **SCORE:**
***Lastname*** ***Firstname*** ***MI***
**Section :** _____ **Date Completed:** _____

**A. Written Work/s:**

**Activity 1**: Accessing your learning

A. Identify common fields, if any exist, or create common fields, if necessary, between the given tables based on how they should be related for the following purposes:
   1. Should show which class each student belongs to
   2. Should show the adviser for each class
   3. Should show the subjects each student is taking and his/her corresponding grades
   4. Should show the students' teacher per subject

| STUDENTS | CLASSES | TEACHERS | GRADES | SUBJECTS |
|---|---|---|---|---|
| FistName | Section | TeacherID | 1stGrading | SubjectCode |
| MiddleName | Level | FirstName | 2ndGrading | SubjectName |
| LastName | RoomNumber | MiddleName | 3rdGrading | Level |
| StAddress | TeacherID | LastName | 4thGrading | |
| Barangay | | StAddress | | |
| City | | Barangay | | |
| PhoneNumber | | City | | |
| | | PhoneNumber | | |
| | | | | |
| | | | | |

Rewrite the fields for each table, including the common fields you determined in the table above.

| | | | | |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

**B.** Identify and encircle the common fields, if any exist, or create common fields, if necessary, between the given tables base on how they should be related for the following purposes:

| CUSTOMERS | PRODUCTS | TRANSACTIONS | TRANSACTIONDETAILS |
|---|---|---|---|
| CustomerID | ProductName | TransactionNumber | ProductName |
| FirstName | Description | TransactionDate | QuantityPurchased |
| MiddleName | Brand/Model | | |
| LastName | UnitPtice | | |
| StAddress | | | |
| City | | | |
| PhoneNumber | | | |

Rewrite the fields for each table, including the common fields you have determined in the table above

| | | | |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |