



Assignment Cover Sheet

Qualification		Module Number and Title
HND in Computing and Software Engineering		SEC4205/Database Design Development
Student Name & No.		Assessor
Harshika Chandramoorthy CL/HNDCSE/92/62		
Hand out date		Submission Date
<add date here>		<add date here>
Assessment type	Duration/Length of Assessment Type	Weighting of Assessment
Report 4000 Words		100%

Learner declaration
<p>I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.</p>

Marks Awarded	
First assessor	
IV marks	
Agreed grade	

Signature of the assessor	Date	
---------------------------	------	--

FEEDBACK FORM

INTERNATIONAL COLLEGE OF BUSINESS & TECHNOLOGY

Module/Title: Database Design and Development /Design a Database Management System for a
The Motor Vehicle Insurance Policies

Student:

Assessor:

Assignment:

Strong features of your work:

Areas for improvement:

Marks Awarded:

Learning outcomes covered

- LO1. Understand the database concepts and components
- LO2 Develop a logical database design
- LO3 Implement a physical database for a given requirement
- LO4 Normalization and document databases

Read the following Scenario and assume that you have assigned to **design and implement a robust Database System** for The Motor Vehicle Insurance Policies.

Scenario and the Tasks

The following data model is designed to hold information relating to Motor Vehicle Insurance Policies. For this scenario we need to define the following facts: These facts define the requirements which the Database must meet and should be agreed between the Database User and the Database Designer prior to physical creation. An insurance company writes policies for drivers. One policy can cover many drivers and also many vehicles, but a vehicle can be related to only one policy. Drivers can share one or more vehicles (e.g. a husband and wife own one vehicle and they both drive the same vehicle or a family can have multiple vehicles). The company gets a master list of violations from the Department of Motor Vehicles. These violations are then input into the system and used to determine the price of the policy. A driver may commit more than one violation. One or more drivers can commit the same violation. The system should keep a track of all customers - active (with insurance) and inactive (held in an archive – for cancelled customers). All customers should be able to get a quote, insurance or cancel the insurance.

The draft facts have been defined as:

The Entities required should include:

- Drivers
- Vehicles
- Policies
- Insurance Groups
- Violations
- Link_VehiclesDrivers
- Link_ViolationsDrivers

The Entities are related as follows:

- The relationship between the tblVehicles and tblDrivers tables is Many-To-Many (a vehicle may be driven by one or more drivers; a driver may drive one or more vehicles), so a link table should be created (e.g. tblLink_VehiclesDrivers).

- The relationship between the tblVehicles and tblInsuranceGroups tables is One-To-Many (a vehicle may belong to only one insurance group; many vehicles can belong to the same or different insurance groups).
- The relationship between the tblViolations and tblDrivers tables is Many-To-Many (a driver may commit one or more violations; a violation may be committed by one or more drivers), so a link table should be created (e.g. tblLink_ViolationsDrivers).
- The relationship between the tblPolicies and tblVehicles tables is One-To-Many (a policy can cover one or more vehicles; a vehicle can be covered and related to only one policy).

Note: Mention your valid assumptions clearly.

Students can add any functionality which will enhance the system and make the proposed solution more comprehensive.

Use Harvard referencing to properly acknowledge all the external sources you use.

Tasks:

Design the database with the constraint that the available technology is relational.

1. Explain what is data model, Critically compare different data models and explain why older data models are being replaced by new data models. **(LO1)** (10 Marks)
2. Critically explain the benefits and limitations of different database technologies. **(LO1)** (5 Marks)
3. Analyze and briefly explain the different approaches to database design. **(LO1)** (5 Marks)
4. Draw an entity relationship diagram for given scenario with proper standards. Identify important keys and represent different types of attributes and relationships. **(LO2)** (10 Marks)
5. Draw Relational Schemas. Effectively map conceptual data models with relational database schema according to the mapping algorithm. All the steps should be clearly mentioned. **(LO2)** (10 Marks)

6. Create the database using SQL server. **(LO3) Practical submission and Demonstration** (18 Marks)

NOTE: Make sure to enter at least 10 records for each table.

7. Write sql queries for below requirements. **(LO3) Practical submission and Demonstration** (6 Marks)

Note: You can write any three queries using at least two tables.

8. Explain the purpose and benefits of having queries in the database for decision making process of the furniture shop with the examples. **(LO3)** (10 Marks)

9. Provide the Test plan, test strategy and proper test cases and provide all the test documents. **(LO4)** (10 Marks)

10. Explain how verification and validation has addressed in your data base and the access rights to various types of users. **(LO4)** (6 Marks)

11. Provide a well-structured documentation including proper academic style and Referencing and in-text citation using Harvard Referencing System.(10 Marks)

Assessment Criteria

This submission will be assessed as follows	Depth of the task
TASK 1	Compare Hierarchical , Network and Relational data model and the importance of adapting a new data model
TASK 2	Compare different database technologies SQL server, Oracle, MySQL
TASK 3	Top down design method, Bottom-up design method, Centralized

	design and De centralized design
TASK 4	ERD with cardinality and attributes with proper standard symbols (Proper theories and techniques need to apply in ERD) . Use Chen & Martin notations for ERD
TASK 5	Relational Schema for ERD state the primary key foreign key concept correctly (Proper theories and techniques need to apply in Relational schema)
TASK 6	Create Database using SQL server
TASK 7	Produce SQL queries for manipulation functions
TASK 8	Clearly explain the purpose and benefits of having queries in the database
TASK 9	Correct Test plan and test cases with all test documents
TASK 10	Clearly explain the applied verification and validation process in the data base and explain the Object

5

	permission, Statement permission and Implied permission Statements
--	--

TASK 11	Well-structured Documentation with proper formatting styles and Harvard Referencing System is used for Referencing and in-text citation

This submission will be assessed as follows	Total marks Allocated	Marks obtained by the student for the answer provided
TASK 1	10	
TASK 2	5	
TASK 3	5	
TASK 4	10	
TASK 5	10	
TASK 6	18	
TASK 7	6	
TASK 8	10	
TASK 9	10	
TASK 10	6	
TASK 11	10	
Total Marks	100	

Marking Scheme

Task-1 contains 10 marks

Criteria	Marks
	Out of 10
Fail Not explained the data model. Not compared enough data models .Not given proper reasons for the replacement.	0-4
Pass Explained the data model. Compared minimum 3 data models. Not provided enough/Proper reasons to replace the older data models	4-6
Good Explained the data model. Compared minimum 5 data models. Provided proper reasons to replace the older data models	6-7
Excellent Well explained about the data model. Critically compared different 5 data models. Well explained why the older data models are being replaced by new data models	7-10

Task-2 contains 5 marks

Criteria	Marks
	Out of 5
Fail Provided a poor explanation. Not provided the enough examples	0-1
Pass Provided a brief explanation of different database technologies	1-3
Average Design/ Good Provided a brief explanation of different database technologies with the evaluation also.	3-4
Excellent Design Critically explained the benefits and limitations of different database technologies with proper examples	4-5

Task-3 contains 5 marks

Criteria	Marks
	Out of 5
Fail Not analyse properly the different approaches. Not explain the different approaches	0-1
Pass Analysed the different approaches. Not provided the proper explanation.	1-3
Average / Good	

Analysed the different approaches using enough features. Provided 3-4 the proper explanation with examples

Excellent Design

Well Analyzed and briefly explained the different approaches to 4-5 database design with proper examples and figures also.

Task-4 contains 10 marks

Diagram should be evaluated according to the following criteria.

Criteria	Marks
	Out of 10
Fail Poor logical database design. <ul style="list-style-type: none">• Include incorrect Entities• Include incorrect Attributes and Relationships• Not used proper /standard symbols	0-4
Pass Develop a logical database design. Entity Relationship Diagram <ul style="list-style-type: none">• Identification of correct Entities• Identification of correct Attributes and Relationships	4-6
Average Design/ Good <ul style="list-style-type: none">• Clear identification of Entities, Attributes and Relationships• Correct use of ERD notations with minor mistakes	6-7
Excellent Design <ul style="list-style-type: none">• Highly detailed diagram• Backed by relevant assumptions• ERD with cardinality and attributes with proper standard symbols• Proper theories and techniques need to apply in ERD• Use Chen & Martin notations for ERD	7-10

Task-5 contains 10 marks

Criteria	Marks
	Out of 10
Fail In correct Relational Schema <ul style="list-style-type: none"> Not identified the correct Entities Attributes and Relationships 	0-4
Pass Relational Schema <ul style="list-style-type: none"> Identification of correct Entities Identification of correct Attributes and Relationships 	4-6
Average Design/Good <ul style="list-style-type: none"> Clear identification of Entities, Attributes and Relationships Correct use of Relational Schema notations with minor mistakes 	6-7
Excellent Design <ul style="list-style-type: none"> Relational Schema with proper standard symbols Proper theories and techniques need to apply in Relational schema Relational Schema for ERD state the primary key foreign key concept correctly 	7-10

Task-6 contains 18 marks

Criteria	Marks
	Out of 18
Excellent <ul style="list-style-type: none">• Appropriate use of more sophisticated database features (e.g. use of stored procedures / functions / triggers to implement business rules)• Database design being proposed to facilitate requirements• Provided the relevant assumptions	13-18
Good <ul style="list-style-type: none">• Make a good attempt to follow the Relational Database Model including proper database objects• Provide More accurate database design	9-13
Pass <ul style="list-style-type: none">• Provide a Basic Database Design with minor mistakes including enough database objects (simple design)	4-9
Fail <ul style="list-style-type: none">• Provide a poor Database Design with mistakes• Not work with enough database objects	0-4

Task-7 contains 6 marks**Written SQL Queries (1 mark for each query 3×1 -> 3)**

- Provided the correct SQL queries

Appropriate Data with proper records (3 marks)

- Derive test data for the Database

Task-8 contains 10 marks

Criteria	Marks
	Out of 10
Excellent <ul style="list-style-type: none">• Detailed Explanation of having queries in Database• Well-structured proper queries to facilitate the requirements• Provided the relevant Examples	7-10
Good <ul style="list-style-type: none">• Well explained the purpose and benefits,• Provided the examples	6-7
Pass Level <ul style="list-style-type: none">• Provide Basic Explanation with the examples including minor mistakes	4-6
Fail <ul style="list-style-type: none">• Provide incorrect queries• Not provide enough examples	0-4

Task-9 contains 10 marks

Criteria	Marks
	Out of 10
Excellent <ul style="list-style-type: none">• Detailed Test Plan• Provided the Test Strategy and Proper test cases• Provided all the test documents with proper test data	7-10
Good <ul style="list-style-type: none">• Provided Correct Test Plan and Test Cases with	6-7

enough test data	.
Pass Level <ul style="list-style-type: none"> • Provided Correct test documents with minor mistakes with enough test data 	4-6
Fail <ul style="list-style-type: none"> • Not provide the proper test documents , Not 	0-4

provide enough test data	
--------------------------	--

Task-10 contains 6 marks

Criteria	Marks
	Out of 6
Excellent Provided and well explained the applied verification and validation process with the relevant examples , Detailed Explanation of Access rights with Suitable Examples for Object permission, Statement permission and Implied permission Statements	4-6
Good Provided the explanation of verification and validation process with the relevant examples Provided the Explanation of Access rights with Suitable Examples	3-4
Pass Provided the process of Verification and Validation Process only. Provided the Access Rights only for the Database without the description and examples	1-3
Fail Not provide the process of Verification and Validation properly and not provide examples to show the Access Rights only for the Database	0-1

Task-11 contains 10 marks

Criteria	Marks
	Out of 10
Excellent Professional standard of documentation with screen shots & good explanation Harvard Referencing System is used for Referencing and in-text citation.	7-10
Good High standard of documentation with screen shots & average explanations	6-7
Pass Acceptable standard of documentation with poor explanations	4-6
Fail Not followed a proper academic style , Not provided the referencing /provided in correct citation.	0-4

Guidelines for the report format

Submission format Report

Paper Size: A4

Words: 4000 words

Page numbers – bottom, right

Line spacing 1.5

Font

Headings 14pt, Bold

Normal 12pt

Font face- Times New Roman

CONTENTS

Acknowledgement.....	05
TASK 01.....	06
What is Data model?.....	07
Reasons of Data model.....	07
Importance of Data model.....	07
Advantages of Data Model.....	08
Disadvantages of Data Model.....	08
Types of Data model.....	08
Hierarchy Data model.....	09
Advantages & Disadvantages of Hierarchy data model.....	09
Network Data Model.....	10
Advantages & Disadvantages of Network Data model.....	11
Comparison of Hierarchy vs Network Data model.....	11
Entity Relational Data Model.....	12
Advantages & Disadvantages of ERD model.....	13
Relational Data Model.....	13
Advantages & Disadvantages of Relational data model.....	14
Comparison of Hierarchy vs Network vs Relational Data Model.....	15
Object-Oriented Data Model	16
Advantages & Disadvantages of Object-Oriented Data model.....	17
Object Relational Data Model.....	18
Explanation for why older data models are being replaced by new data models.....	19
Task 02.....	20
comparison of the different Database Technologies.....	21
SQL Server.....	21
Benefits & Limitations of SQL Server.....	22
Oracle Data Base.....	23
Benefits & Limitations of Oracle Data Base	23
MYSQL.....	24
Benefits & Limitations of MYSQL.....	25
TASK 03.....	26

Different Approaches of Data Base Design.....	27
Bottom-Up Database Approach.....	27
Top-Down Database Approach.....	28
Analyzation Between Bottom-Up and Top-Down Database Approach.....	29
Centralized Database Approach.....	30
De-centralized Database Approach.....	31
Analyzation Between Centralized and Decentralized Data Base Approaches.....	33
Task 04	34
Vehicle Insurance Policy Management System Entities and their Attributes.....	35
Entity Relationship Diagram for scenario.....	36
TASK 05.....	37
Relational schema for the Scenario.....	38
TASK 06.....	41
TASK 07.....	42
TASK 08.....	43
What is Data Base Query?.....	44
The Power of Queries.....	44
The SQL SELECT STATEMENT.....	44
Advantages and Disadvantages of SQL Query.....	45
TASK 09.....	47
Test Plan.....	48
Components of Test Plan.....	48
Test Strategy.....	49
Components of Test Strategy.....	49
Difference Between Test Plan & Test Strategy.....	50
Test Case.....	51
Benefits of Test Case.....	51
Components of Test Case.....	52
TASK 10.....	53
What is Verification?.....	54
Methods of Verification.....	54
Verification methods.....	54

What is Validation?.....	54
Types of Data Validation.....	54
Methods of Validation.....	55
Difference Between Verification and Validation.....	56
What are Access Rights?.....	57
Permissions Types.....	57
References.....	58
Conclusion.....	

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my lecture Mr. Chitahura sir who gave me the golden opportunity to do this wonderful project on the topic Data base Design & Development, which also helped me in doing a lot of Research and I came to know about so many new things, Iam really thankful to them.

Secondly, I would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

Task 01

WHAT IS DATAMODEL?

This referred as it specifies how the logical structure of an information domain is modeled, how data is related to each other, how they are processed and stored within the system, the method of formulating data in a very exceeding system in a standardized format. It helps to meet requirements in business.

Reasons for using a Data Model

- ✧ Visual representation of a knowledge helps to spice up the data analysis and it provides a holistic picture of the data which could be utilized by developers to make a physical data base.
- ✧ All-important data of an enterprises are accurately presented within the model and it reduces the likelihood of omission. Data omission can cause incorrect results and faulty reports
- ✧ The data model portrays a far better understanding of business requirements.
- ✧ It helps within the creation of a sturdy design that brings the full data of an organization on the identical platform. It assists in identifying the redundant, duplicate & missing data similarly.
- ✧ Knowledgeable data model helps in providing better consistency across all projects of an enterprise.
- ✧ It improves data quality.
- ✧ It helps project managers with a stronger scope and quality management & also it improves performance to the core.
- ✧ It defines relational tables, stored procedures, and first & foreign keys.

Importance of Data Model

- ✧ Are a communication tool.
- ✧ Give an overall view of the data base.
- ✧ Organize data for various users.
- ✧ Are an abstraction for the creation of wonderful database.

Advantages of Data model






- ✧ Data objects provided by the functional team are presented accurately with data modeling.
- ✧ Data modeling allows you to question data from the information base and derive report supported the knowledge. It indirectly contributes to data analysis with the help of reports, these reports are going to be used for improving the quality and the productivity of the projects.
- ✧ Business have many data form of format and provides a structured system for such unstructured sorts of data.
- ✧ Data modeling improves business intelligence by making data modelers work closely with the underside realities of the project which include gathering data from multiple unstructured sources, reporting requirements, spending patterns, etc...
- ✧ It improves communication across the organization.

Disadvantages of data model

- ✧ Development of an information model is also an awfully tedious job; one should bear in mind of the physical characteristics of the information storage.
- ✧ The system involves complex application development and knowledge of biographical truth.
- ✧ The model is not quite user-friendly small challenges induced within the system require major modification within the whole application.

Types of Data Model

There are many sorts of data models being employed in nowadays. during this data model's variety of them are here,

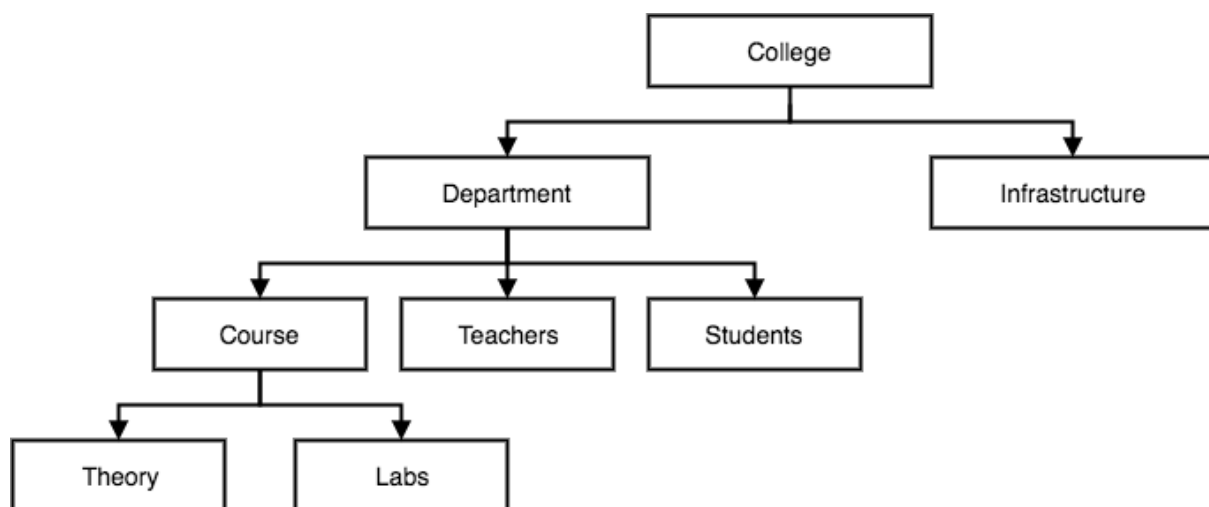
-  **Hierarchical model**
-  **Network model**
-  **Entity-Relationship model**
-  **Relational model**
-  **Object-Oriented model**

- ✚ **Object-Relational model**
- ✚ **Flat Data model**
- ✚ **Semi-Structured Data model**
- ✚ **Associative Data model**
- ✚ **Context Data model**

In these data models I would like to explain about some data models.

HIERARCHICAL DATA MODEL

In the hierarchical model, data is organized into a tree-like structure with one parent and plenty of children in each record. The main drawback of this model is that it can only have one to many node relations.



Advantages and disadvantages of Hierarchical model

Advantages

- ✧ Promotes sharing of data.
- ✧ The relationship between parent and child promotes conceptual simplicity and integrity of data.
- ✧ The protection of the database is provided and implemented by the design and development of the database.
- ✧ Active to 1: Many relationships

Disadvantages

- * Requires knowledge of the characteristics of physical data storage.
- * System requires hierarchical path knowledge.
- * Changes in implementation include changes in application systems altogether.
- * Implementation, limitation.
- * No definition of data.
- * Missing criteria.

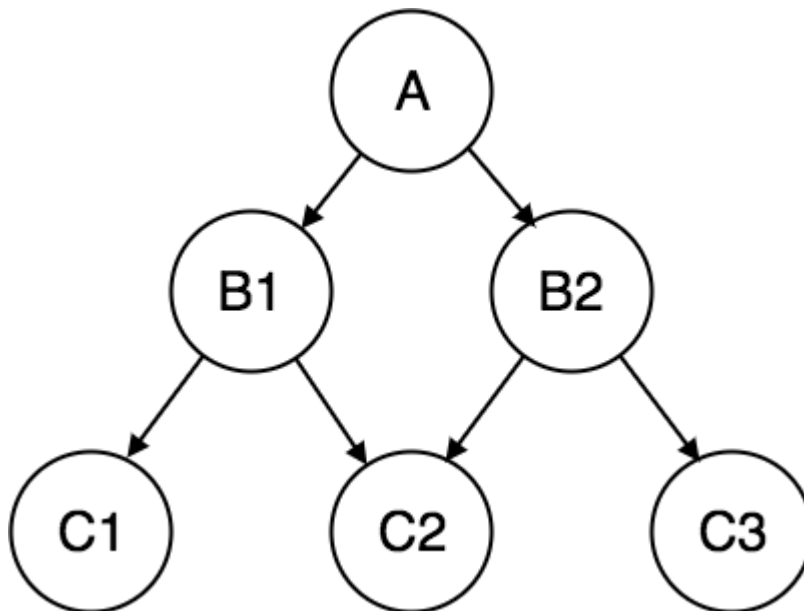
Network Data Model

This is an extension of the model in Hierarchy. Data is structured in this model.

More graph type, and more than one parent node can be required. Data is more relevant in this database model, as more relationships exist.

Established under this model database. As the information is more relevant, access to the information is also additional.

Faster and simpler. This model for the database was a standard map.



Advantages and disadvantages of Network Model

Advantages

- * Conceptual, simplicity.
- * Handles more relationship types.
- * Data access is flexible.
- * Data owner/member relationship promotes data integrity.
- * Conformance to standards.
- * Includes data definition language (DDL) and data manipulation language (DML).

Disadvantages

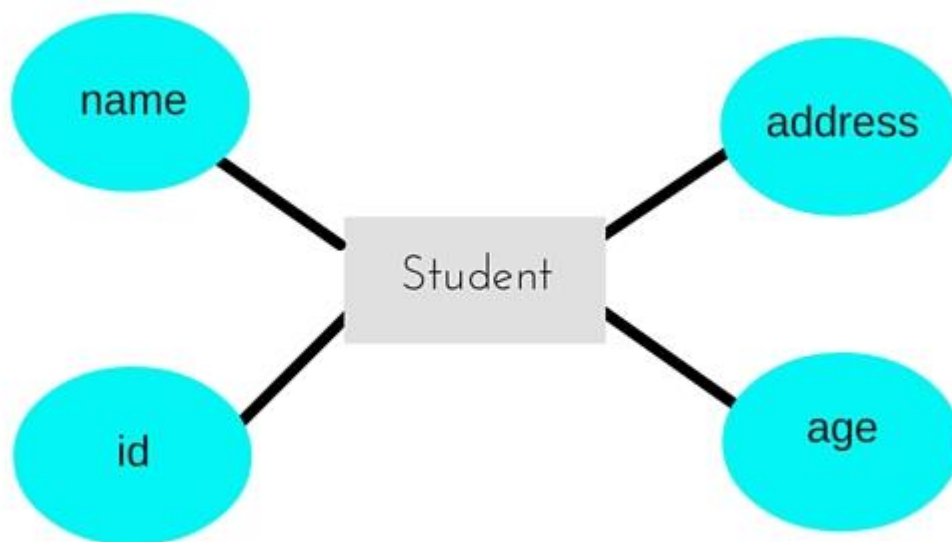
- * System complexity limits efficiency
- * Navigation system yields complex implementation, application development and management.
- * Structural changes require altogether application programs.

Comparison Between Hierarchical and Network Data Models

Hierarchical Data Model	Network Data Model
Manage large amount of data for complex manufacturing projects.	Represents complex data relationships
Represented by an upside-down tree which contains segment.	Improve database performance and impose a database standard.
Segment: Equivalent of a file system's record type.	
Depicts a set of one-to-many (1:M) relationships.	Depicts both One-to-Many (1:M) and Many -to-Many (M: N) relationships.

Entity Relationship Model

Graphical representation of entities and their relationships in a very data base structure.



Entity Relationship Diagram (ERD)

Uses graphic representations to model database components.

COMPONENT OF (ERD)

ENTITY

Ex: Person, Places, Objects, Events & Concepts

ATTRIBUTES

Each entity can have number of characteristics and these characteristics are called as attributes.

KEY ATTRIBUTES

The key attributes are an attribute that uniquely identifies an entity in the entity set.

RELATIONSHIP

There are three types of relationships they are

One to One (1:1)

One to Many (1:M)

Many to Many (M: N)

Entity instances or Entity occurrence

Rows in relational tables.

Connectivity

Term accustomed label the link types.

Advantages and Disadvantages of Entity Relationship model

Advantages

- ✧ Visual modeling yields conceptual simplicity.
- ✧ Visual representation makes it an efficient communication tool.
- ✧ It is integrated with the dominant relational model.

Disadvantages

- ✧ Limited constraint representation.
- ✧ Limited relationship representation.
- ✧ No data manipulation language.
- ✧ Loss of data content occurs when attributes are away from entities to avoid crowded displays

Relational Data model

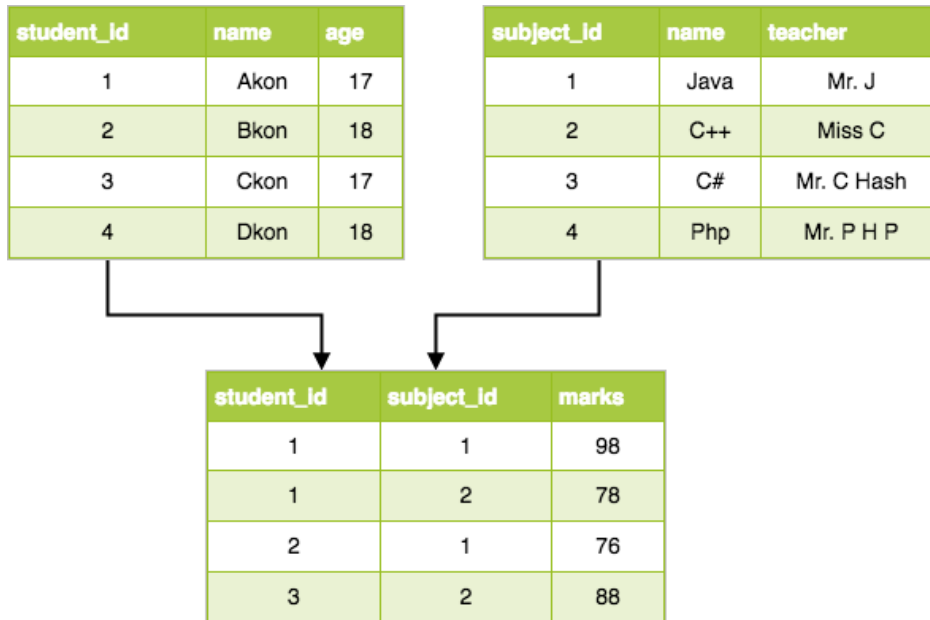
Produced an automatic transmission database that replaced standard transmission databases.

BASED ON RELATION

Relation or Table: Matrix composed of intersecting tuple and attribute.

Tuple: Rows

Attributes: Column's Describes a precise of data manipulation constructs.



Advantages and Disadvantages of relational Model

Advantages

- ✧ Structural independence is promoted using independent tables.
- ✧ Tabular views improve conceptual simplicity.
- ✧ Ad hoc query capability is based on SQL.
- ✧ Isolates the end user from physical- level details.
- ✧ Improves implementation and management simplicity.

Disadvantages

- ✧ Requires substantial hardware system and software overhead.
- ✧ Conceptual simplicity gives untrained people the tools to use a good system poorly.
- ✧ May promote information problems.

Relational Database Management System

Performs basic functions provided by the hierarchical and network DBMS systems.

Makes the relational data model easier to understand and implement.

Hides the complexities of the relational model from the user.

Comparison Between Hierarchy vs Network vs Relational Model

Hierarchical Model	Network Model	Relational Data Model
Relationship between records is of the parent child type.	Relationship is a record in the form of references or connections	Relationship between records is represented by a relationship that contains a key for each record that is involved.
Relationship between records is the child type of the parent.	Within this model, too, many to many partnerships can be introduced.	Most partnerships can be quickly put into practice
Some of the relationships in this model cannot be articulated.	Application of record relationships is complicated due to the use of pointers.	Implementation of relationships using a key or composite key field is quite easy.
This is a simple straightforward and natural method of record relationship implementation.	Network model is useful to represent records of many to many relationships	Relational model is useful to represent and relate most real-world objects.
This type of data model is only useful when the database includes a certain hierarchical character.	Search for an easy record as there are multiple paths to access a data element	To search for a data element, a unique indexed key field is used.
It is difficult to search for a record because one can recover it	The record relationship in the network model is physical	Relational model does not maintain physical connections between records, data is logically organized in rows and columns and stored in table form.
During updating or deletion process, chances of data inconsistency is involved.	There is no inconsistency problem in the network model	Consistency methods such as the standardization process are adopted for data integrity

Object-Oriented Data Model

OBJECT-ORIENTED DATABASE(OODB)

Are dependent on the object-oriented programming paradigm. It enables us to create classes, organize objects, structures an inheritance hierarchy and call methods of represents class-subclass link. The dash (_ _ _) represents attribute-domain link.

CLASS

Collection of similar objects with shared structure and behavior organized in a class hierarchy.

CLASS HIERARCHY

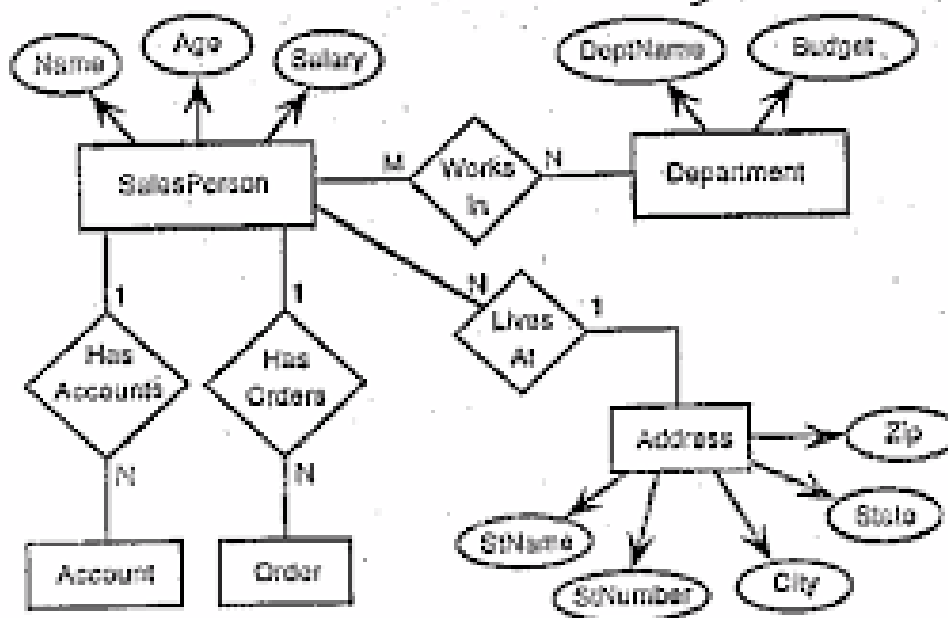
Resembles an upside-down tree in which each class has only one parent.

INHERITANCE

Object inherit methods and attributes of parent class.

UNIFIED MODELING LANGUAGE (UML)

Describes set of diagrams and symbols to graphically model a system.



Advantages and Disadvantages of OODB

Advantages

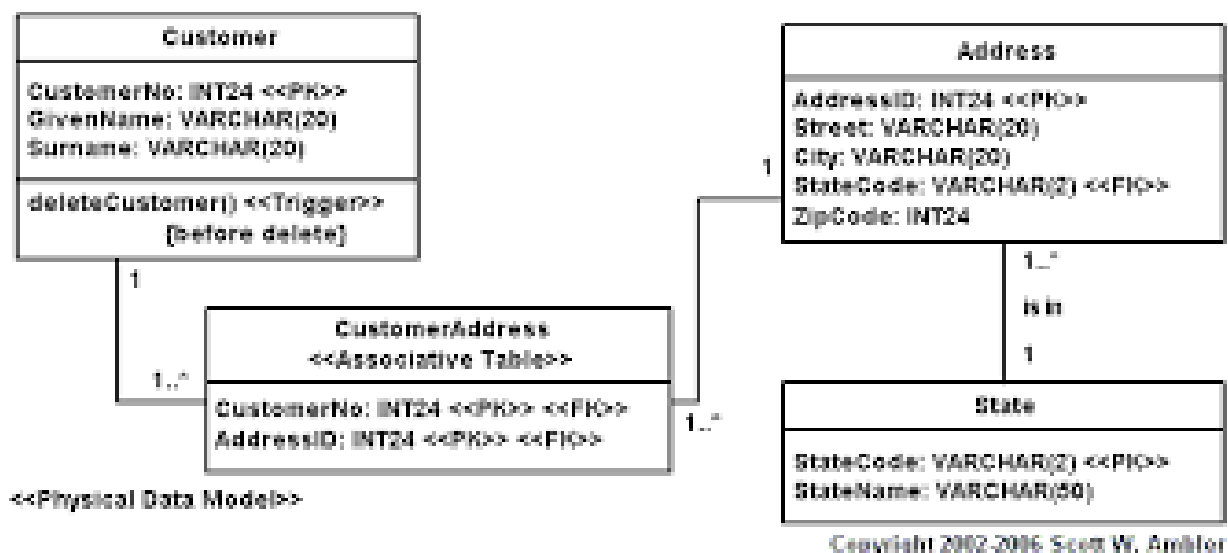
- ✳ OODB allows nested structure so it is easy to navigate through these complex structures in the form of objects rather than in the form of tables and record.
- ✳ OODB allows arbitrary data types and sub-classing with inheritance.
- ✳ OODB are favored in those applications where the relationship among data elements are more important than data items themselves.

Disadvantages

- ✳ Slow development of standards caused vendors to supply their own enhancement.
- ✳ Comprised widely accepted standards.
- ✳ Complex navigational system.
- ✳ Learning curve is steep.
- ✳ High system overhead slows transactions.

Object Relational Data Model

An object-relational database (ORD) could be a management system (DBMS) that is composed of both an on-line database (RDBMS) and an on-line database (OODBMS). ORD supports the fundamental components of any electronic information service model in its schemas and the command language used, like objects, classes, and inheritance. An object-relational database may additionally be referred to as an object computer database management systems (ORDBMS)



Advantages and Disadvantages of ORDM

Advantages

- ✧ Reuse and sharing.
- ✧ Increased productivity.
- ✧ Use of experience in developing RDMS

Disadvantages

- ✧ The ORDBMS approach has the obvious disadvantages of complexity and associated increased costs. Further, there are the proponents of relational approach that believe the essential simplicity and purity of the relational model are lost with these types of extension.

Explanation for why older data models are being replaced by new data models.

The high-level answer is things change. Business has changed, and what is required from technology is usually changing. Systems and their Data support the business. so, means because the business evolves, the information model evolves with it. Also, technology itself changes and requires different configurations of information to figure with the newer data management infrastructure.

For example, A designer built his first data models to model a straightforward register that ran on one IBM mainframe. once they switched to Client-Server distributed processing, the information had to vary to produce data at each layer of the architecture. therefore, the model changed. And on and on.

Think of modern analytics. we wish information from data that was not available a decade ago. Not to mention the sad incontrovertible fact that most data bases were not modeled the least bit and are junk. And even those who were modeled were modeled WRONG (usually because they relied on an enormous vendor who wanted it done their proprietary way). Again, things change, and that we must change with them.

Task 02

comparison of the different Database Technologies

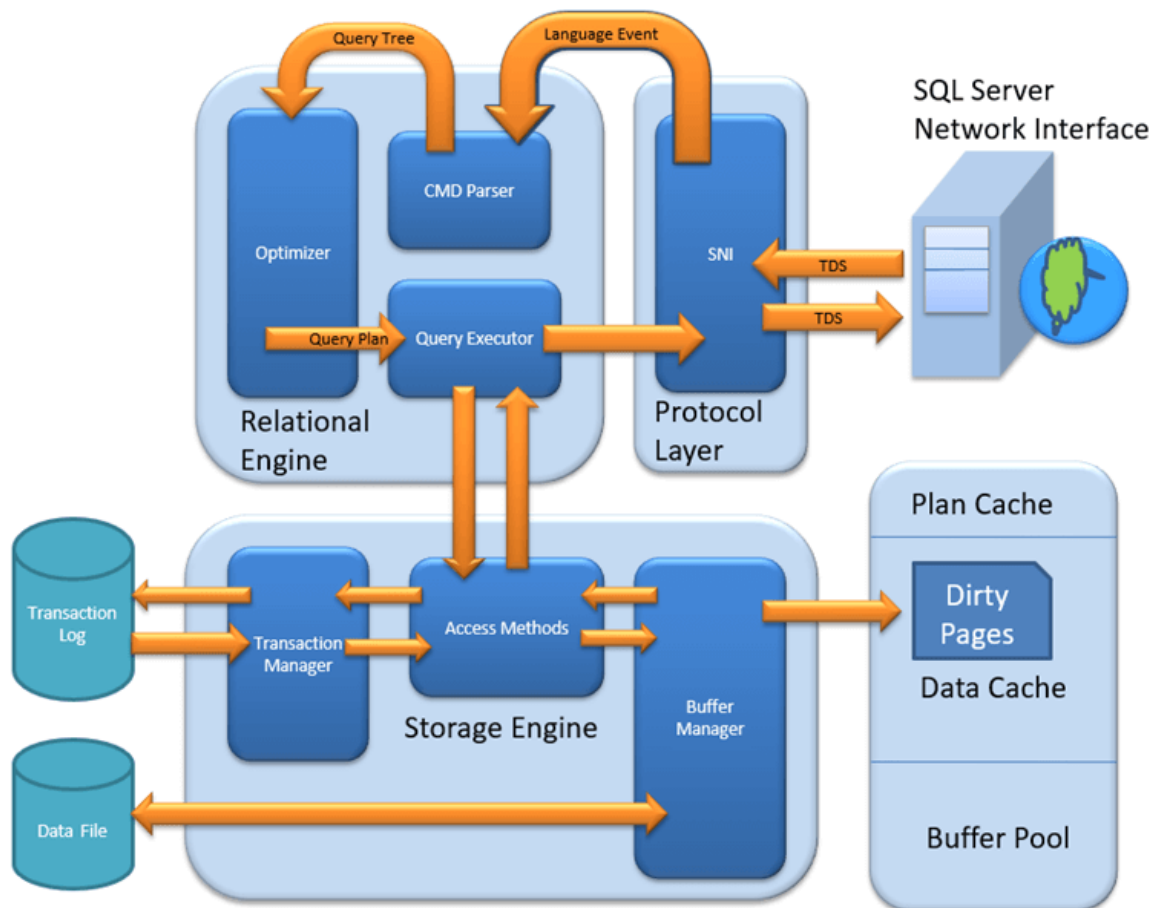
Property	SQL Server	MySQL	ORACLE
Purpose	A Relational database of Microsoft	A Widely used open source database	Hugely used RDBMS
The Basic model of Database	RDBMS	RDBMS	RDBMS
Secondary model of Database	Graph DBMS Key-value store Document Store	Key-value store Document store	Graph DBMS Key-value store Document Store RDF store
Website	www.microsoft.com/en-us/sql-server	www.mysql.com	www.oracle.com/database/index.html
Developer	Microsoft	Oracle	Oracle
Initial Release Year	1989	1995	1980
License	Commercial	Open Source	Commercial
Cloud Support	No	No	No
Implementation Language	C++	C and C++	C and C++
Data Scheme	Yes	Yes	Yes

SQL Server

SQL SERVER is a relational database management system (RDBMS) developed by Microsoft.

It is primarily designed and developed to compete with MySQL and Oracle database.

SQL Server supports ANSI SQL, which is the standard SQL (Structured Query Language) language.



Benefits & Limitations of SQL Server

Benefits

- ✧ High Speed:
- ✧ SQL Queries can be used to retrieve large amount of records from a database quickly and efficiently.
- ✧ Well Defined standards exists:
- ✧ SQL databases use long established standards, which is being adopted by ANSI & ISO. Non-SQL databases do not adhere to any clear standards.
- ✧ No coding required:
- ✧ Using standards SQL, it is easier to manage database systems without having to write substantial amount of code.

- ✳ Emergence of ORDBMS:
- ✳ Previously SQL databases were synonymous with relational database. With the emergence of Object-Oriented DBMS, object storage capabilities are extended to relational databases.

Limitations

- ✳ Difficulty in Interfacing:
- ✳ Interfacing an SQL database is more complex than adding few lines in codes.
- ✳ More Feature Implemented in Proprietary Way:
- ✳ Although SQL database conform to ANSI & ISO standards, some databases go for proprietary extensions to standard SQL to ensure vendor lock-in.

Oracle Data Base

Oracle database (Oracle DB) is additionally an electronic information service management system (RDBMS) from the Oracle Corporation. Originally developed in 1977 by Lawrence Ellison and other developers, Oracle DB is one in all the foremost trusted and widely used electronic information service engines. The system is made around an electronic information service framework within which data objects could even be directly accessed by users (or an application front end) through structured language (SQL). Oracle is additionally a very scalable electronic information service architecture and is usually employed by global enterprises, which manage and process data across wide and native area networks. The Oracle database has its own network component to permit communications across networks. Oracle also defined as a single use tool based on the client-server system that allows the management of database known as a more complete database system.

Benefits & Limitations of Oracle Data Base

Benefits

- ✳ Centralized control and management system:
- ✳ This allows the data to be controlled completely from a tabular exchange since it is responsible for assigning, adding, deleting records, and modifying them.
- ✳ Standardization:
- ✳ Allows standardization between different implementation of SQL.

- ✳ Grouping of transaction:
- ✳ it allows to group several transactions and divide each activity into segments and thus achieve a better performance in less time possible.
- ✳ Performance methods:
- ✳ Applies methods to improve the database through a cluster application.

Limitations

- ✳ Incompatibility and Complexity:
- ✳ This is presented as a limitation in the areas of time, data syntax, and character sensitivity
- ✳ Management of the Structure:
- ✳ Oracle often tends to be complex or sometimes difficult to manage for certain activities, so it is recommended to install a basic version and configure with a minimal customization.
- ✳ The Price:
- ✳ Many times, there is no price or a specific budget for the Oracle license as it usually changes over time depending on the change of policies, patches, and updates by the company.

MYSQL

MySQL is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL).

SQL is the most popular language for adding, accessing, and managing content in a database. It is most noted for its quick processing, proven reliability, ease, and flexibility of use. MySQL is an essential part of almost every open source PHP application. Good examples for PHP & MySQL-based scripts are [WordPress](#), [Joomla](#), [Magento](#) and [Drupal](#).



Benefits & Limitations of MYSQL

BENEFITS

- ✧ It is easy to use.
- ✧ It is open source.
- ✧ It is incredibly inexpensive.
- ✧ It is an industry standard (And Still Extremely Popular)

Limitation

- ✧ It is got a few stability issues.
- ✧ It suffers from relatively poor performances.
- ✧ Development is not community driven-and hence has lagged.
- ✧ Its functionality tends to be heavily dependent on addons.
- ✧ Developers may find some of its limitations to be frustrating.

TASK 03

Different Approaches of Data Base Design

There are different types of approaches in data base design and I have mentioned some of them in here and explained about it. They are

Bottom-Up Database Approach

Top-Down Database Approach

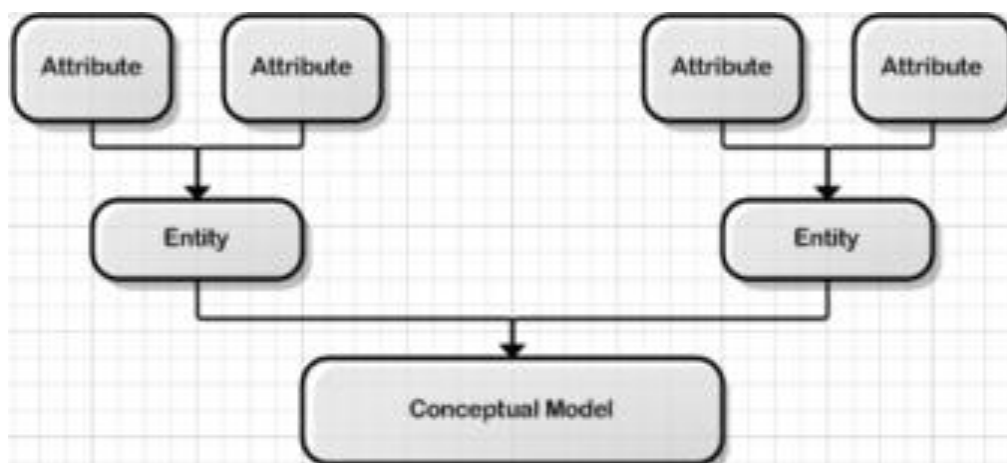
Centralized Database Approach

De-centralized Database Approach

Bottom-Up Data Base Approach

In this design, individual parts of the system are laid out in details. The parts are the linked to make larger components, which are successively linked until a whole system is made. Object oriented language like C++ or java uses bottom up approach where each object is identified first.

Make decisions about reusable low-level utilities then decide how there will be put together to make high level construct. Contrast between Top down design and bottom up design.



Top-Down Data Base Approach

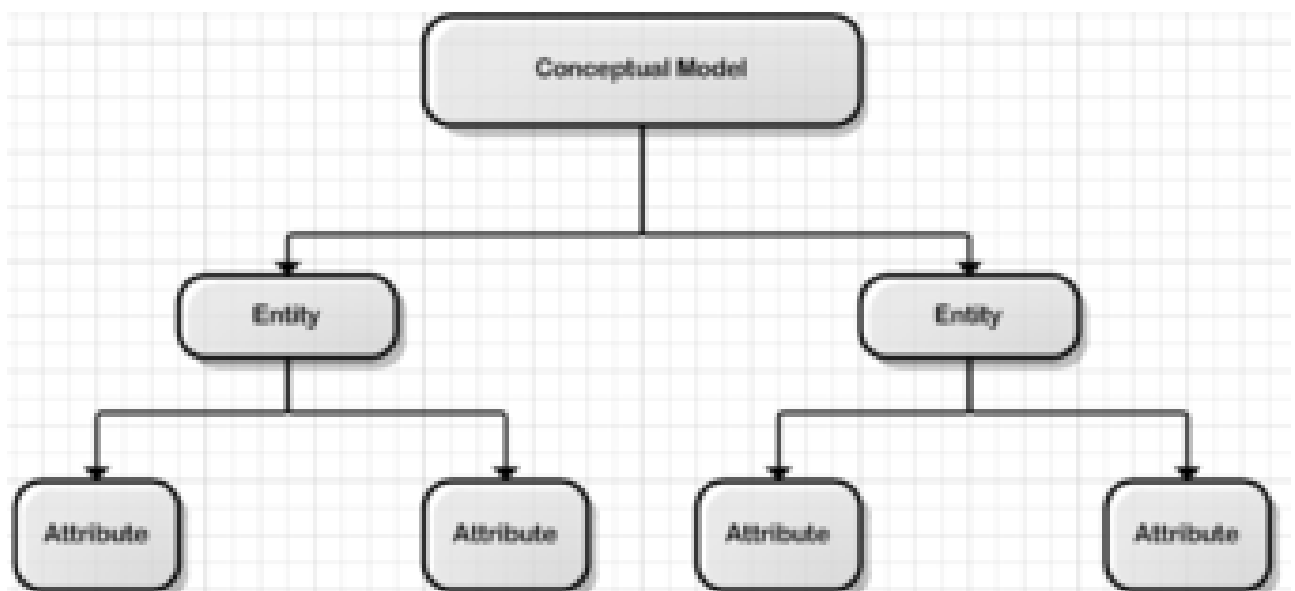
In top down model, an outline of system is formulated without going into details for any a part of it. Each a part of it then refined into more details, defining it in yet more details until the complete specification is detailed enough to validate the model. if we glance at a drag as a full, it is visiting appear unfeasible because of it is so complicated for example: Writing a University programmed, writing a word processing system. Complicated issues could also be resolved victimization high down style, conjointly remarked as Stepwise refinement where, we break the matter into parts, then break the parts into parts soon and now each of part are easy to try and do.

Breaking problems into parts help us to identify what needs to be done.

At each step of refinement new parts will become less complex and therefore easier to solve.

Parts of solution may turn out to be reusable.

Breaking problems into parts allows more than one person to solve the problem.



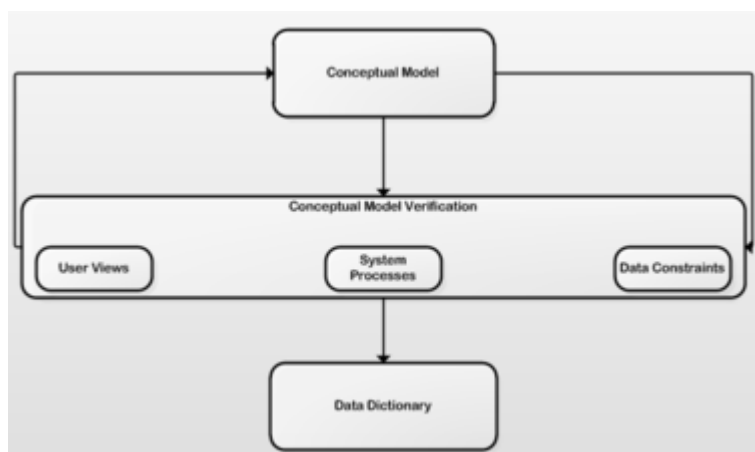
Analyzation Between Bottom-Up and Top-Down Database Approach

TOP DOWN APPROACH	BOTTOM UP APPROACH
In this approach We focus on breaking up the problem into smaller parts.	In bottom up approach, we solve smaller problems and integrate it as whole and complete the solution.
Mainly used by structured programming language such as COBOL, Fortran, C etc.	Mainly used by object-oriented programming language such as C++, C#, Python.
Each part is programmed separately therefore contain redundancy.	Redundancy is minimized by using data encapsulation and data hiding.
In this the communications are less among modules.	In this module must have communication.
It is used in debugging, module documentation, etc.	It is basically used in testing.
In top down approach, decomposition takes place.	In bottom up approach composition takes place.
In this top function of system might be hard to identify.	In this sometimes we cannot build a program from the piece we have started.
In this implementation details may differ.	This is not natural for people to assemble.

Centralized Data Base Approach

In a centralized database model, a core unit acts during a way that serves the whole company. This description is usually wont to describe an analytics model which has been completely centralized, though it is going to also talk over with shared information database which is accessible to everyone.

Companies that are large and have a single-business organization have a requirement for information, analytics, and applications to be able to cross various boundaries. This enhances the communication within the organization, providing more practical leadership from the C-Suite as specific needs are accurately associated with the individuals tasked with a response. Choosing a centralized system is about funding some way for a novel business strategy to own the foremost potential impact. in keeping with reporting from Deloitte, 42% of companies have established a centralized approach.



Advantages

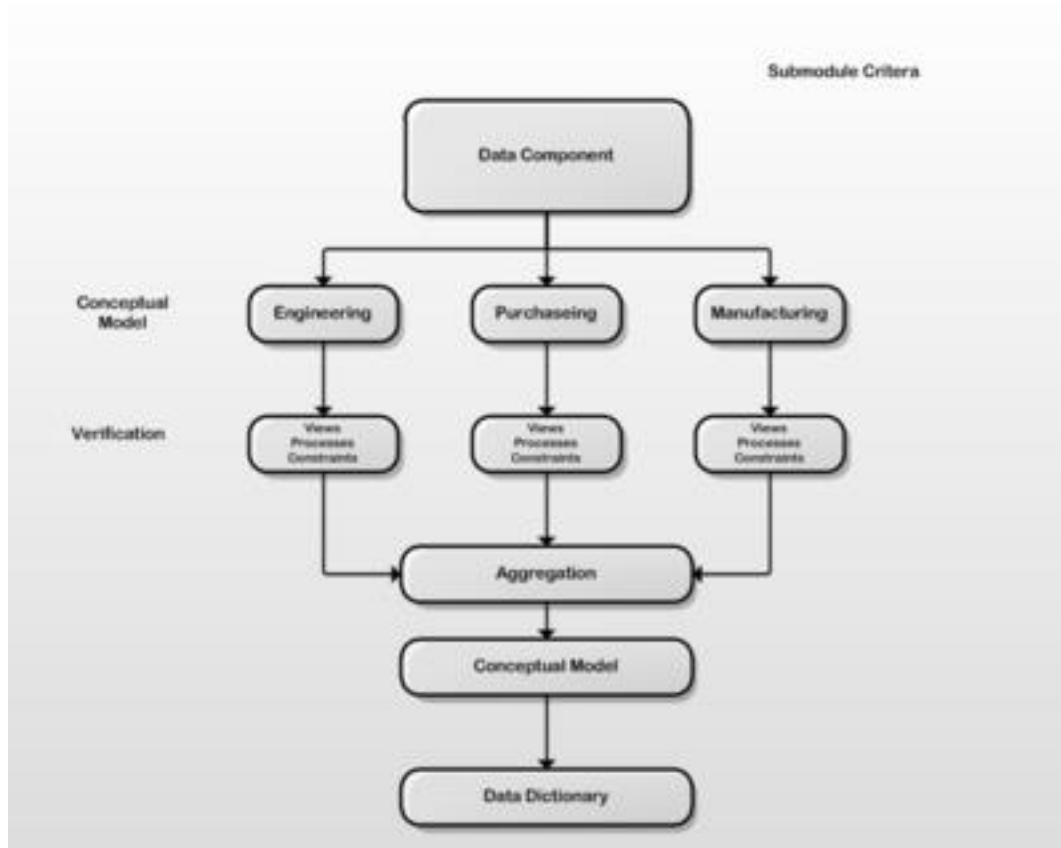
- ✳ It allows for performing on cross-functional projects.
- ✳ It is easier to share ideas across analysts.
- ✳ Analysts are often assigned to specific problems or projects centrally.
- ✳ Higher levels of security are additionally obtained.
- ✳ Higher levels of dependability are present within the system.
- ✳ It reduces conflict.
- ✳ Organizations can act with greater speed.
- ✳ It helps an organization stay near a focused vision.

Disadvantages

- ✳ It can become unresponsive to the wants of the business.
- ✳ There are lower levels of location-based adaptability.
- ✳ It can have a negative impact on local morale.
- ✳ Succession planning is restricted with a centralized database.
- ✳ It reduces the amount of legitimate feedback received.
- ✳ It is visiting increase costs. There is a risk of loss.

Decentralization Data Base Approach

As its name implies, decentralized systems do not have one central owner. Instead, they use multiple central owners, each of which usually stores a replica of the resource's users can access. A decentralized system is often at the same time as liable to crashes as a centralized one. However, it is by choice more tolerant to faults. That is because when one or more central owners or servers fail, the others can still provide data access to users. Resources remain active if a minimum of 1 all told the central servers still operate. Usually, this means that system owners can repair faulty servers and address the opposite problems while the system itself continues to run as was common. Server crashes in an exceedingly very decentralized system may affect the performance and limit access to some data. But in terms of overall system uptime, this method offers an infinite improvement over a centralized system. Another advantage of this design is that the interval to the data is typically faster. That is because owners can create nodes in several regions or areas where user activity is high. However, decentralized systems are still in danger of the identical security and privacy risks to users as centralized systems. And while their fault tolerance is higher, this comes at a price. Maintaining a decentralized system is often dearer.



Advantages

- ✧ Less likely to fail than a centralized system.
- ✧ Better performance.
- ✧ Allows for a more diverse and more flexible system.

Disadvantages

- ✧ Security and privacy risks to users
- ✧ Higher maintenance costs
- ✧ Inconsistent performance when not properly optimized

Analysis Between Centralized and Decentralized Data Base Approaches

Centralized Design Approach	Decentralized Design Approach
A type of data base that contains a single data base located at one location in the network.	a type of a data base that contains two or more data base files located at different locations in the network.
Managing, updating in taking in backups of data is easier because there is only one data file.	As there are multiple database files in a distributed database it requires time to synchronize data.
Requires time for accessing data because multiple users access the database file.	Speed in accessing the data is higher because the data is retrieved from the nearest data base file.
If the database fails, the users do not have access to a database.	If one database fails, the users can still access other database files.
Has more data consistency and it provides the complete view to the user.	Can have data replications, and there can be some data inconsistency.

Task 04

Vehicle Insurance Policy Management System Entities and their Attributes

Driver Entity: Driver ID, Name, DOB, Date Licensed, License Reference, Address, Contact.

Vehicles Entity: Vehicle ID, Vehicle Chassis No, Vehicle Engine No, Vehicle Engine Size, Vehicle Make, Vehicle Model, Registration Year, Registration No, Vehicle Value, Insurance Groups, Policy Reference No.

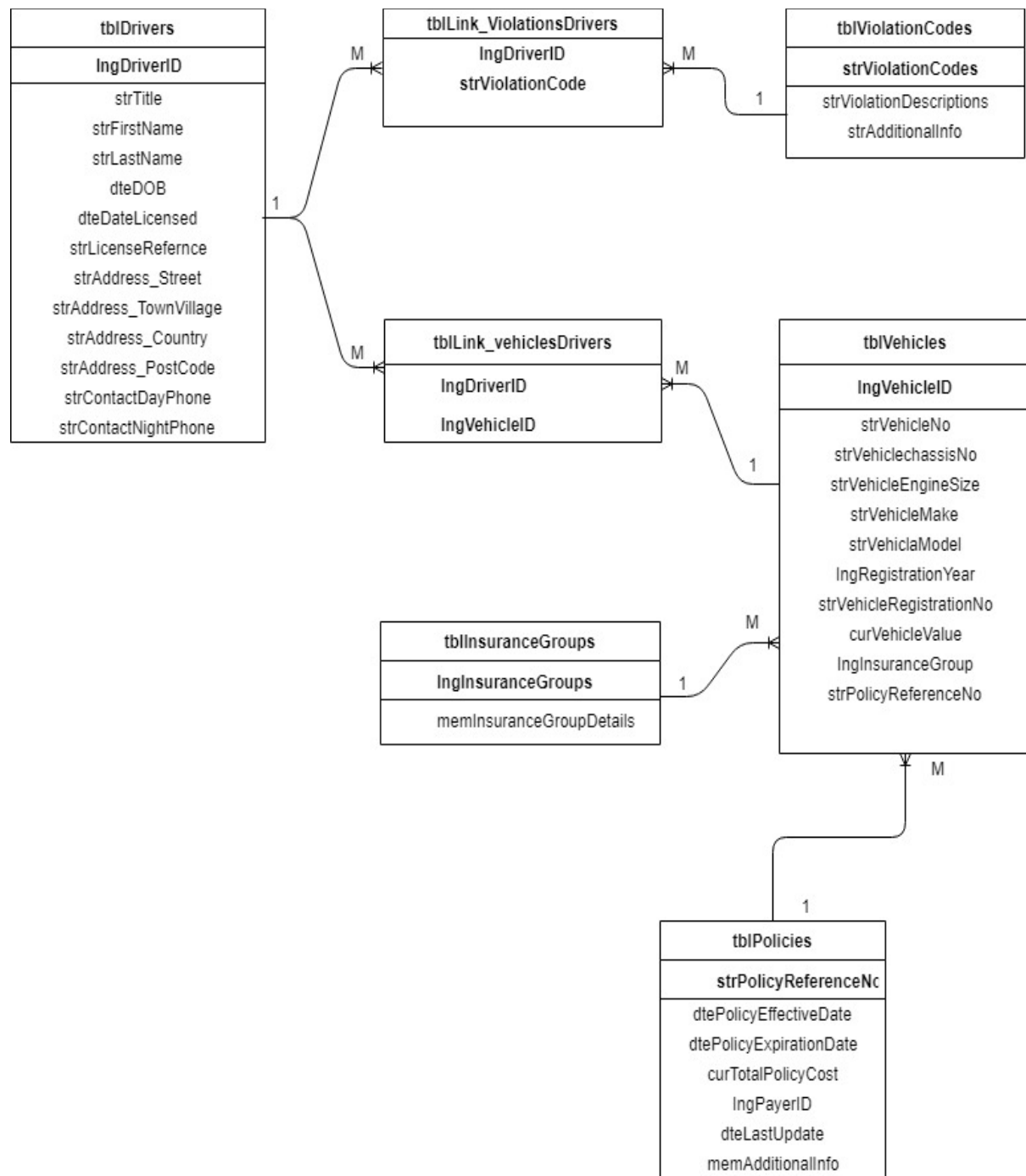
Insurance Groups Entity: Insurance Groups, Members Details.

Violations Entity: Violation Codes, Violation Description, Additional Info.

Link Vehicles Drivers Entity: Driver ID, Vehicle ID.

Link Violations Drivers Entity: Driver ID, Violation Codes.

ENTITY RELATIONSHIP DIAGRAM FOR THE SCENERIO



TASK 05

Relational schema for the Scenario

Mapping Many to Many relationship

TblVehicles (Vehicle ID, vehicle chassis No, engine No, Engine size, Vehicle make, model, registration year, registration No, vehicle value, insurance group, policy reference No, Driver Id)

TblDrivers (Driver Id, first name, last name, title, DOB, date licensed, license reference No, street, town village, country, postal code, day phone, night phone, vehicle ID)

FK/NM

DRIVER (Driver ID, first name, last name, title, DOB, date licensed, license reference No, street, town village, country, postal code, day phone, night phone, vehicle ID)

VEHICLE (Vehicle ID, vehicle chassis no, vehicle engine No, vehicle engine size, vehicle make, model, registration year, registration No, vehicle value, insurance group, policy reference No, driver ID)

FK/NM

TblLink Vehicle Drivers (Driver ID, vehicle Id)

Mapping One to Many relationship

FK

INSURANCE GROUPS (insurance_group, members insurance group details)

VEHICLE (Vehicle ID, vehicle chassis no, vehicle engine No, vehicle engine size, vehicle make, model, registration year, registration No, vehicle value, policy reference No, insurance group)

Mapping Many to Many Relationship

Tblviolation (violation codes, violation description, additional info)

TblDrivers (Driver Id, first name, last name, title, DOB, date licensed, license reference No, street, town village, country, postal code, day phone, night phone)

TblLink violationDrivers (violation codes, Driver ID)

FK/NM

Violations (violation codes, violation description, additional info)

Drivers (Driver Id, first name, last name, title, DOB, date licensed, license reference No, street, town village, country, postal code, day phone, night phone)

FK/NM

TblLink violationDrivers (violation codes, Driver ID)

Mapping One to Many Relationship

FK

Policies (policy reference No, policy effective date, policy expiry date, total policy cost, payer ID, last update, mem additional info)

Vehicles (Vehicle ID, vehicle chassis No, engine No, Engine size, Vehicle make, model, registration year, registration No, vehicle value, insurance group, policy reference)

TASK 06

TASK 07

TASK 08

What is Data Base Query?

A query about a database is a request for data from a database. Usually the request is for data retrieval; however, data can be manipulated using queries as well. The data may come from one or more tables. Queries can also perform data calculations or automate the management tasks of your data. You can also check your updates before you add them to the database., or maybe from other queries.

The Power of Queries

A database can show complicated patterns and events, but only by using the query. This capacity is used. Most tables store a large amount of data are part of a complex database. A query enables the data to be stored in a single table and you can conveniently analyze it.

Queries can also carry out data measurements or automate data processing activities. Before committing it to the database you can also review updates of your data.

The SQL SELECT STATEMENT

The SELECT statement retrieves data from a database.

The data is returned in a table-like structure called a result-set.

SELECT is the most frequently used action on a database

Used for queries on single or multiple tables.

Clauses of SELECT statement.

SELECT

list the columns (and expressions) that should be returned from the query.

FROM

Indicate the table(s) or view(s) from which data will be obtained.

WHERE

Indicate the conditions under which a row will included in the result.

GROUP BY

Indicate categorization of results.

HAVING

Indicate the conditions under which a category(group)will be included.

ORDER BY

Sorts the result according to specified criteria.

Advantages and Disadvantages of SQL Query

Advantages of SQL Query

Not necessary coding

The database system can be handled very easily without the use of regular SQL to write a considerably large amount of code.

Standards well established

SQL databases used by ISO and ANSI are used for a long time. The non-SQL databases are not compatible with any requirements.

Potability

In PCs, servers, laptops and even some cell phones, SQL can be used in the software.

Interactive Language

You will use this domain language to communicate with databases and provide answers in seconds to complex questions.

Various views of data

User can interpret the database structure and databases differently with the aid of the SQL language

Disadvantages of SQL Query

Difficult Interface

SQL has a complex interface, which makes access difficult for some users.

Partial Control

Due to hidden business rules, the programmers who use SQL have no control over the database.

Implementation

Those databases use regular SQL proprietary extensions to ensure that the client is protected.

Cost

Many programmers have difficulties in accessing certain SQL versions because of the running costs.

TASK 09

Test Plan

Test planning, the most important activity to ensure that a list of tasks and milestones initially appears in a baseline plan to track the project's progress. This also sets out the scale of the research effort.

It is the key document also named as a master test plan or a project test plan and usually established during the project's early phase.

COMPONENTS OF TEST PLAN

No.	Parameter	Description
1.	Test plan identifier	Unique identifying reference.
2.	Introduction	A brief introduction about the project and to the document.
3.	Test items	A test item is a software item that is the application under test.
4.	Features to be tested	A feature that needs to test on the test ware.
5.	Features not to be tested	Identify the features and the reasons for not including as part of testing.
6.	Approach	Details about the overall approach to testing.
7.	Item pass/fail criteria	Documented whether a software item has passed or failed its test.
8.	Test deliverables	The deliverables that are delivered as part of the testing process, such as test plans, test specifications and test summary reports.
9.	Testing tasks	All tasks for planning and executing the testing.
10.	Environmental needs	Defining the environmental requirements such as hardware, software, OS, network configurations, tools required.

11.	Responsibilities	Lists the roles and responsibilities of the team members.
12.	Staffing and training needs	Captures the actual staffing requirements and any specific skills and training requirements.
13.	Schedule	States the important project delivery dates and key milestones.
14.	Risks and Mitigation	High-level project risks and assumptions and a mitigating plan for each identified risk.
15.	Approvals	Captures all approvers of the document, their titles, and the sign off date.

Test Strategy

The research technique is also known as the test method, which determines how tests should be conducted. Study method consists of two techniques:

Proactive-An approach that initiates the prototype design process as early as possible to identify and repair the defects before the build is produced.

Reactive-An approach where research is not undertaken until after design and coding has been completed.

Components of Test Strategy

Scope and Objectives

Business issues

Roles and responsibilities

Communication and status reporting

Test deliverables

Industry standards to follow

Test automation and tools

Testing measurements and metrics
 Risks and mitigation
 Defect reporting and tracking
 Change and configuration management
 Training plan.

Difference Between Test Plan & Test Strategy

TEST PLAN	TEST STRATEGY
It is derived from software requirement specification (SRS).	It is derived from the Business Requirement document (BRS).
It is prepared by the test lead or manager.	It is developed by the project manager or the Business analyst.
Test plan id, features to be tested, test techniques, testing tasks, features pass or fail criteria, test deliverables, responsibilities, and schedule, etc. are the components of the test plan.	Objectives and scope, documentation formats, test processes, team reporting structure, client communication strategy, etc. are the components of test strategy.
If there is a new feature or a change in the requirement that is happened, then the test plan document gets updated.	Test strategy maintains the standards while preparing the document. It is also called as Static document.
We can prepare the test plan individually.	In smaller projects, test strategy is often found as a section of a test plan.
We can prepare a Test plan at the project level.	We can use Test strategy at multiple projects.
It describes how to test, when to test, who will test and what to test.	It describes what type of technique to follow and which module to test.

TEST PLAN

We can describe about the specifications by using a Test Plan.

TEST STRATEGY

Test strategy describes about the general approaches.

Test Case

A test case is a series of activities carried out on a system to assess if it meets the needs and functions of software correctly. A test case is intended to assess whether various features within the system meet their needs and to ensure that the system complies with all applicable specifications, directives, and customer requirements. Any flaws or faults within the program may also be identified by writing a test case.

Test cases are normally written by Quality Assurance Team members or research manager and can be used for any device research as step-by - step instructions. Tests start when a system feature or collection of features is completed by the development team. A sequence or collection of test cases is called a test suite. A test case document includes test steps, test data, preconditions and the postconditions that verify requirements.

Benefits of Test Case

Guaranteed good test coverage.

Reduced maintenance and software support costs.

Reusable test cases.

Confirmation that the software satisfies end-user requirements.

Improved quality of software and user experience.

Higher quality products lead to more satisfied customers.

More satisfied customers will increase company profits.

Components of Test Case

Test name: A title that describes the functionality or feature that the test is verifying.

Test ID: Typically, a numeric or alphanumeric identifier that QA engineers and testers use to group test cases into test suites.

Objective: Also called the description, this important component describes what the test intends to verify in one to two sentences.

References: Links to user stories, design specifications or requirements that the test be expected to verify.

Prerequisites: Any conditions that are necessary for the tester or QA engineer to perform the test.

Test setup: This component identifies what the test case needs to run correctly, such as app version, operation system, date and time requirements and security specifications.

Test steps: Detailed descriptions of the sequential actions that must be taken to complete the test.

Expected results: An outline of how the system should respond to each test step.

TASK 10

What is Verification?

The Project Manager identifies customer requirements and prioritizes them during the planning phase. A product is developed during the execution stage. It is essential to ensure that the product supplied does what it was supposed to do before it is handed over to the customer and the project can be shut down. An internal method is called a test for determining the product's fulfillment with consumer requirements.

Methods of Verification

Verification methods are classified in to 4 types they are

Formal verification techniques rely on mathematical proof of correctness.

Informal verification techniques rely on subjective human reasoning.

Static verification techniques assess the system by using the source code without executing it.

Dynamic verification techniques assess the system by executing it first

Verification methods

Inspections

Demonstration

Testing

Analysis

What is Validation?

Data validation is a process that ensures the delivery of clean and clear data to the programs, applications and services using it. It checks for the integrity and validity of data that is being inputted to different software and its components. Data validation ensures that the data complies with the requirements and quality benchmarks.

Data validation is also known as input validation.

Types of Data Validation

Code validation

Data type validation

Data range validation

Constraint validation

Structured validation

Methods of Validation

Type – If you make a specific field numeric then it will not allow you to input any letters or other non-numeric characters. Be wary when using the numeric data type. If you use it for fields like phone numbers, it will not allow you to enter spaces, or other human-friendly forms of formatting. Some data types can carry out an extra type check.

Presence – This is sometimes called Allow Blank or Mandatory. This type of validation compels the user to enter data in the required field.

Unique Identifier – It is essential that one record can be plainly recognized from another record. Each record has one field that functions as a unique identifier for a record. An easy validation check can be done to make sure that a value occurs only once in this field—it doesn't matter if there are thousands of records in the database, the check can be carried out just the same.

Range Check – Range check is a validation check which can be applied to numeric fields. This is done to ensure that only numbers within a certain domain can be entered into a field. Remember that this does not necessarily mean that the data entered will be correct. But it will certainly lie within reasonable limits.

Format – This is used for a field that requires an entry in a specific format.

Restricted Choice – There are times that fields in a database have a definite amount of data that can be entered into them.

Programming a database to accept only one of a series of valid choices can prevent errors and can also serve to lessen the time it takes to input data.

This has different forms like a list box, combo box, or radio button.

Benefits of Restricted Choice:

Faster data entry, because it is typically much quicker to select from a list than to type each individual entry.

Enhanced accuracy because it lessens the risk of spelling mistakes.

Limits the options to choose from by only displaying the essential choices.

Referential Integrity – If you are using a relational database, then you can impose referential integrity to validate inputs. You can check data inputs in certain fields against values in other tables. For example, in the job database, when a new hire is entered, you could check the supervisor name against the employee table, just like you could check the department name against the department table.

Difference Between Verification and Validation

VERIFICATION	VALIDATION
Verification is a process, to ensure that weather we are building the product right i.e., to verify the requirements which we have and to verify whether we are developing the product accordingly or not.	Validation is the process, weather we are building the correct product i.e., to validate the product which we have developed is right or not.
As per IEEE-STD-610: the process of evaluation software to determine whether the products of given development phase satisfy the conditions imposed at the beginning of that phase.	As per IEE-STD-610: the process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements [IEEE-STD-610]
It the answers the question, Am I building the product, right?	It answers the question, Am I building a right product?
Activities involved here are inspections, Reviews Walk throughs.	Activities involved in this testing the software application by mean of Whitebox, Gray box, & Blackbox testing.
It is a Low-Level Activity.	It is a High-Level Activity
It is a static method of checking documents and files.	It is a dynamic process of testing the real product.
It does not involve code execution	It involves code execution
Low cost compared to validation test	Costly composed to verification test.

What are Access Rights?

Access Rights are the permissions an individual user or a computer application holds to read, write, modify, delete, or otherwise access a computer file; change configurations or settings, or add or remove applications. The level of access rights often depends on the user's position or supervisory role in the company.

Permissions Types

Permission can be granted at the database, schema, or object level. ... The default permissions that are granted to system objects at the time of ... or the permissions can be an implied or covering permission GRANT or DENY.

Standard Permissions provide the least amount of control over access to databases and are provided for backward compatibility purposes and to make the assignment of permissions extremely easy to manage.

Object Level Permissions are new to and provide object level user access controls (i.e. table, view, stored procedure, etc.) allowing the administrator to grant or revoke permissions to individual objects, users, and groups of users.

Implied permission, in the context of real property, means conduct or words or both that imply that an owner or occupant of land has agreed to another person's use of or ability to enter land.

References

- alida.d, n.d. *study.com*. [Online]
Available at: <https://study.com/academy/lesson/project-verification-methods-techniques.html>
- Anon., n.d. *berty. tech*. [Online]
Available at: <https://berty.tech/blog/decentralized-distributed-centralized>
[Accessed 20 june 2019].
- Anon., n.d. *Janbask training*. [Online]
Available at: <http://www.janbasktraining.com/blog/oracle-vs-sqlserver-vs-mysql/>
- Anon., n.d. *studytonight*. [Online]
Available at: <http://www.studytonight.com/dbms/database-models.php>
- Anon., n.d. *techopedia*. [Online]
Available at: <http://www.techopedia.com/definition/8711/oracle-database>
- Anon., n.d. *techopedia*. [Online]
Available at: <http://www.techopedia.com/definition/8714/object-relational-database-ord>
- Anon., n.d. *techopedia*. [Online]
Available at: <https://www.techopedia.com/definition/10283/data-validation>
[Accessed 9 february 2017].
- Anon., n.d. *us legal*. [Online]
Available at: <https://definitions.uslegal.com/i/IMPLIED-permission/#:~:text=Implied%20permission%2C%20in%20the%20context,or%20ability%20to%20enter%20land>
- Anon., n.d. *webeduclick.com-programming language tutorial*. [Online]
Available at: <http://www.webeduclick.com/comparision-between-hierarchical-model-network-model-relational-model/>
- Arteaga, A., n.d. *Broadcom*. [Online]
Available at: <http://www.community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=e116f73b-2a43-45b9-b433-38c33518e81a&CommunityKey=bc9cf16a-865b-4cd5-8b97->

[9d33656e43b1&tab=librarydocuments](#)

[Accessed 02 february 2019].

chapple, m., n.d. *life ware*. [Online]

Available at: <https://www.lifewire.com/query-definition-1019180>

[Accessed 18 november 2019].

do factory, n.d. *dofactory*. [Online]

Available at: <https://www.dofactory.com/sql/select>

Gaille, B., n.d. *brandongaille small business & marketing advice*. [Online]

Available at: <https://brandongaille.com/15-centralized-database-advantages-and-disadvantages/>

[Accessed 16 july 2018].

Gananathan, V., n.d. *Structured Query Language*. [Online]

Available at: <http://www.cs.iit.edu/~cs561/cs425/VenkatashSQLIntro/default.htm>

Geeks for geeks, n.d. *Geeks for Geeks*. [Online]

Available at: <https://www.geeksforgeeks.org/difference-between-bottom-up-model-and-top-down-model/>

ghahari, a., n.d. *devqa.io*. [Online]

Available at: <https://devqa.io/test-strategy-and-test-plan/>

[Accessed 1 june 2019].

intellipaat, 2015. *intellipaat tutorials*. [Online]

Available at: <http://intellipaat.com/blog/data-modeling-tutorial-for-beginners/>

Mack, J., n.d. *DATAREALM*. [Online]

Available at: <https://www.datarealm.com/blog/five-advantages-disadvantages-of-mysql/>

[Accessed 28 MAY 2014].

point, t., n.d. *tutorialspoint*. [Online]

Available at: http://www.tutorialspoint.com/dbms/dbms_data_models.htm

rajkumar, n.d. *softwaretestingmaterial*. [Online]

Available at: <https://www.softwaretestingmaterial.com/verification-and-validation/>

[Accessed 7 february 2020].

rouse, m., n.d. *software testing help*. [Online]

Available at: <https://www.softwaretestinghelp.com/difference-between-test-plan-test-strategy->

[test-case-test-script-test-scenario-and-test-condition/](#)

[Accessed 16 april 2020].

shingh, c., 2015. *beginnersbook*. [Online]

Available at: <http://www.bignnersbook.com/2015/04/hierarchical-model-in-dbms/>

soffront, n.d. *soffront*. [Online]

Available at: <https://soffront.com/glossary/access-rights/>

techtarget, n.d. *techtarget*. [Online]

Available at: <https://whatis.techtarget.com/definition/test-case>

Thakur, s., n.d. *whatisdbms*. [Online]

Available at: <https://whatisdbms.com/what-is-sql-applications-advantages-and-disadvantages/>

[Accessed 2 june 2017].

tutorial points, n.d. *titorial points*. [Online]

Available at: https://www.tutorialspoint.com/software_testing_dictionary/test_plan.htm

tutorials, w.-p. l., n.d. *webeduclick.com*. [Online]

Available at: <http://websduclick.com/types-of-database-models/>

Conclusion

By completing this course work I get knowledge about extended Entity Relationship diagram, master detail form and report. this course work developed my developing skills and my research and creative ability. Now I am more confident about Data base design and Development. At this moment I am so much happy by completing this course work successfully.