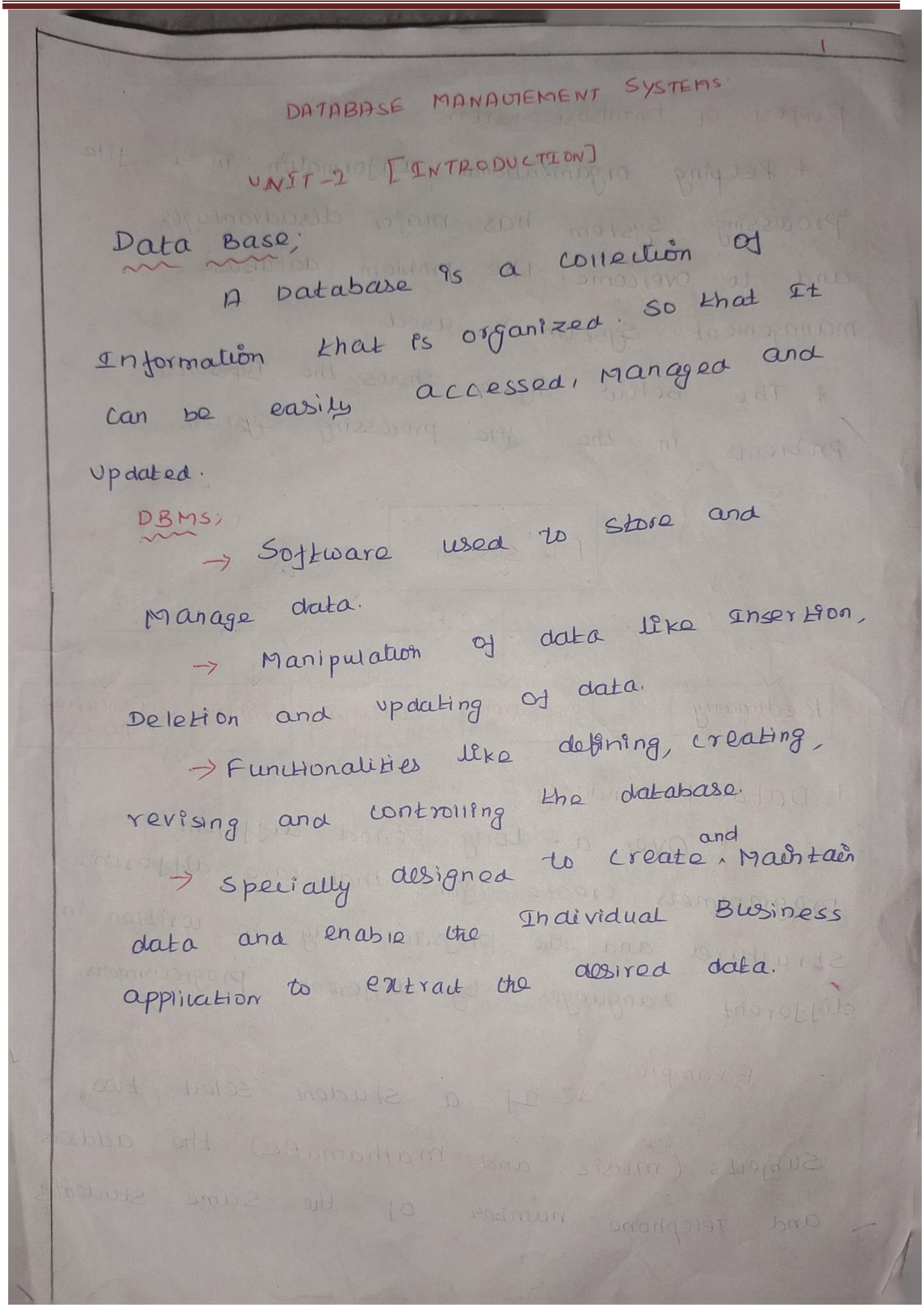


DATABASE MANAGEMENT SYSTEMS

UNIT -2 [INTRODUCTION]

Data Base;

A Database is a collection of Information that is organized so that It can be easily accessed, Managed and Updated.

DBMS;

→ Software used to store and Manage data.

→ Manipulation of data like Insertion, Deletion and updating of data.

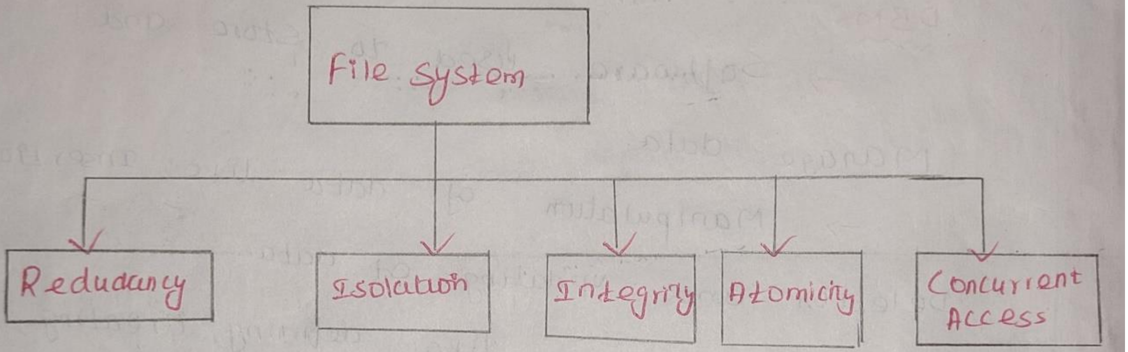
→ Functionalities like defining, creating, revising and controlling the database.

→ Specially designed to create and Maintain data and enable the Individual Business application to extract the desired data.

PURPOSE OF DATABASE SYSTEMS:

* Keeping organizational information in a file processing system has major disadvantages and to overcome the problem database management system is used.

* The below figure shows the Types of problems in the file processing system.

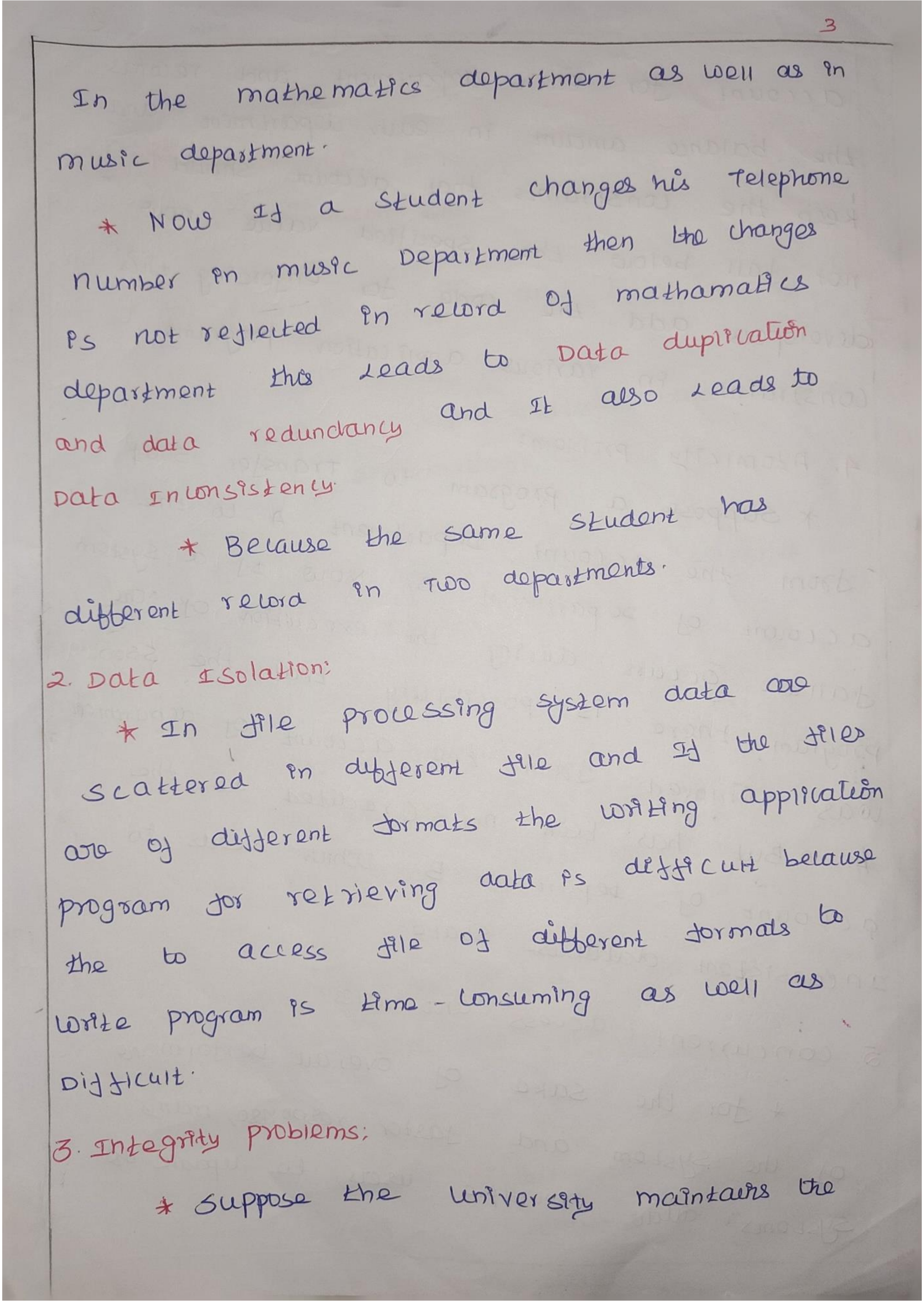


1. Data Redundancy:

* Over a long period, different programmers create files that have different structures and the program may be written in different languages by different programmers.

Example;

* If a student select two subjects (music and mathamatics) the address and Telephone number of the same students



3

In the mathematics department as well as in music department.

* Now If a Student changes his Telephone number in music Department then the changes is not reflected in record of mathamatics department this leads to Data duplication and data redundancy and It also leads to Data Inconsistency.

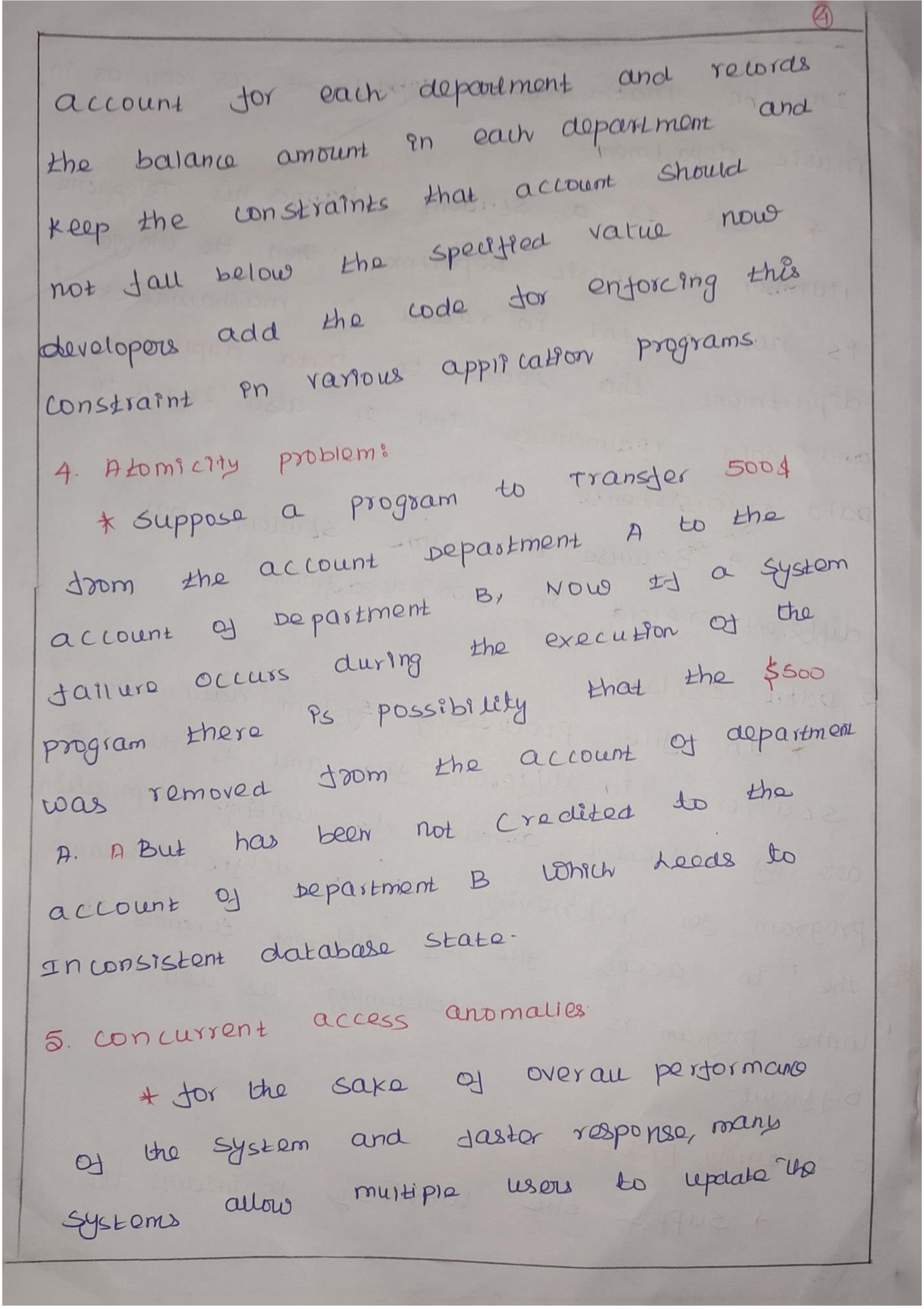
* Because the same student has different record in two departments.

2. Data Isolation:

* In file processing system data are scattered in different file and If the files are of different formats the writing application program for retrieving data is difficult because the to access file of different formats to write program is time - consuming as well as Difficult.

3. Integrity problems;

* Suppose the university maintains the



4

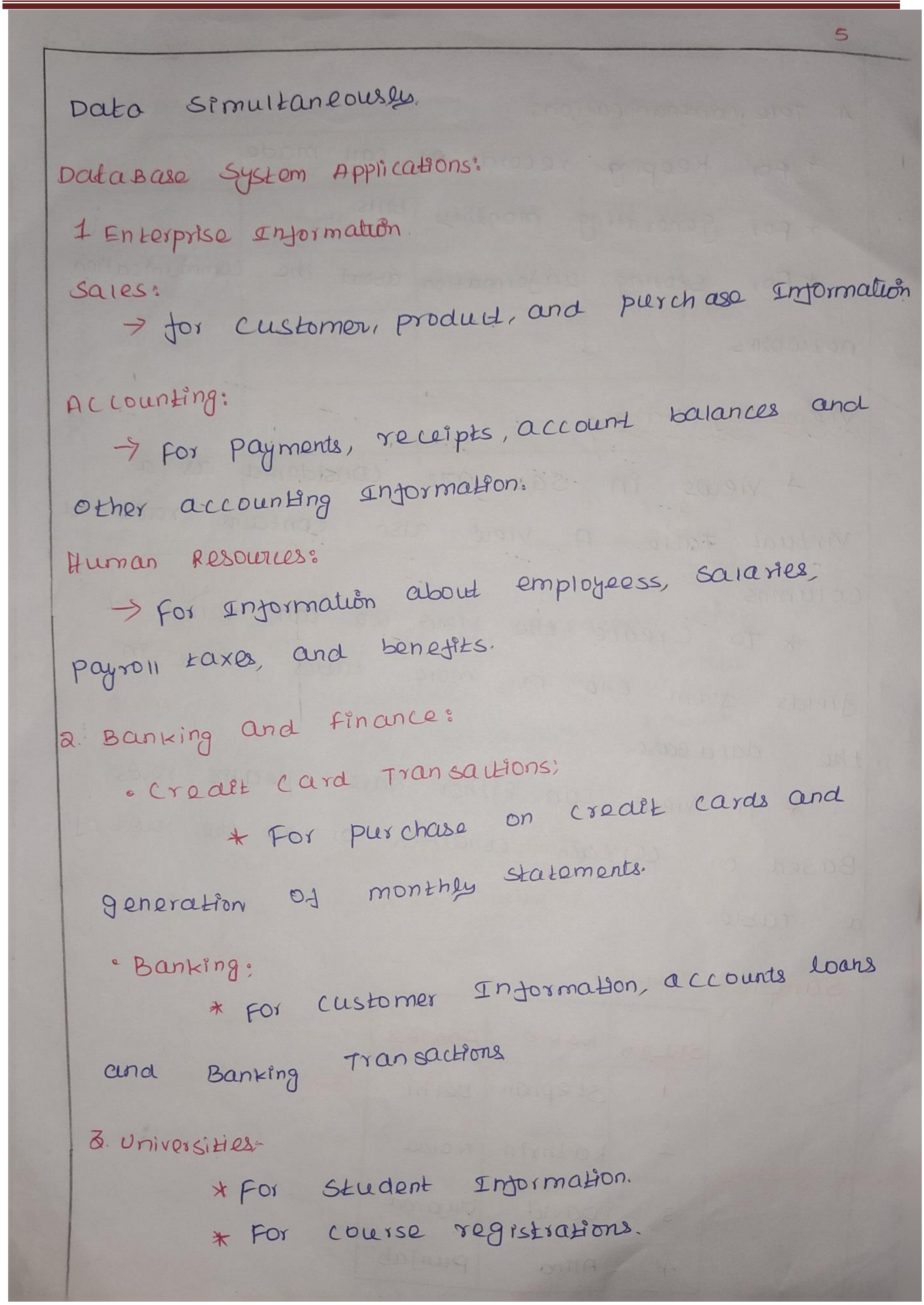
account for each department and records the balance amount in each department and keep the constraints that account should not fall below the specified value now developers add the code for enforcing this constraint in various application programs.

4. Atomicity problem:

* suppose a program to Transfer 500$ from the account Department A to the account of Department B, Now If a system failure occurs during the execution of the program there is possibility that the $500 was removed from the account of departmen A. A But has been not credited to the account of Department B which Leeds to Inconsistent database state.

5. concurrent access anomalies

* for the sake of overall performance of the system and faster response, many systems allow multiple users to update the



Data simultaneously.

Database System Applications:

1 Enterprise Information

Sales:

→ for customer, product, and purchase Information

Accounting:

→ for payments, receipts, account balances and other accounting information.

Human Resources:

→ For information about employeess, salaries, payroll taxes, and benefits.

2. Banking and finance:

• Credit Card Transactions;

* For purchase on credit cards and generation of monthly statements.

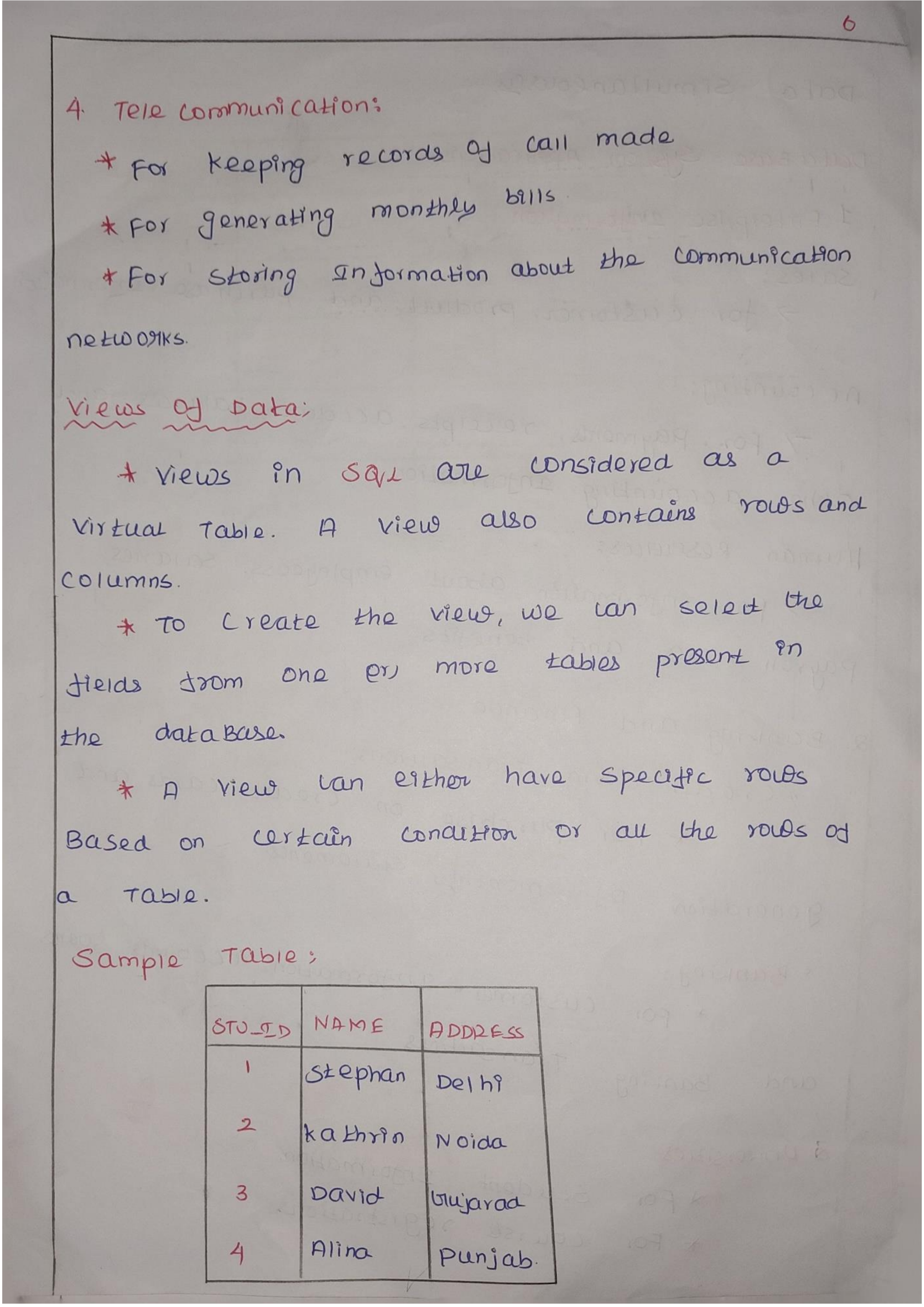
• Banking:

* For customer Information, accounts loans and Banking Transactions

3. Universities-

* For Student Information.

* For course registrations.



4. Tele communication:

- * For keeping records of call made
- * For generating monthly bills
- * For storing Information about the communication networks.

Views of Data:

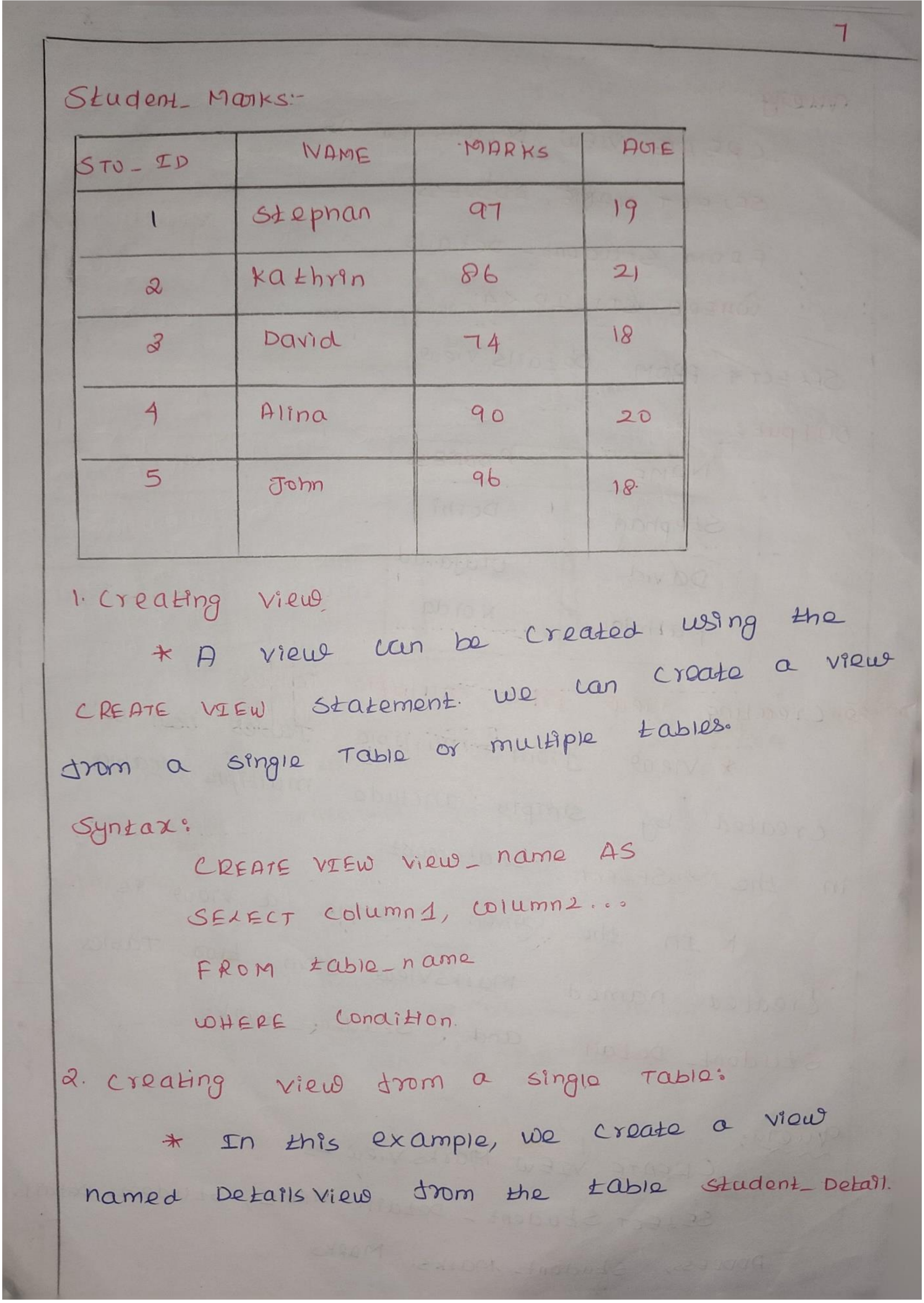
* Views in SQL are considered as a virtual Table. A view also contains rows and columns.

* To create the view, we can select the fields from one or more tables present in the database.

* A view can either have specific rows Based on certain condition or all the rows of a Table.

Sample Table:

STU_ID	NAME	ADDRESS
1	Stephan	Delhi
2	kathrin	Noida
3	David	Gujarad
4	Alina	Punjab.



Student_ Marks:-

STU_ID	NAME	MARKS	AGE
1	Stephan	97	19
2	kathrin	86	21
3	David	74	18
4	Alina	90	20
5	John	96	18.

1. Creating view

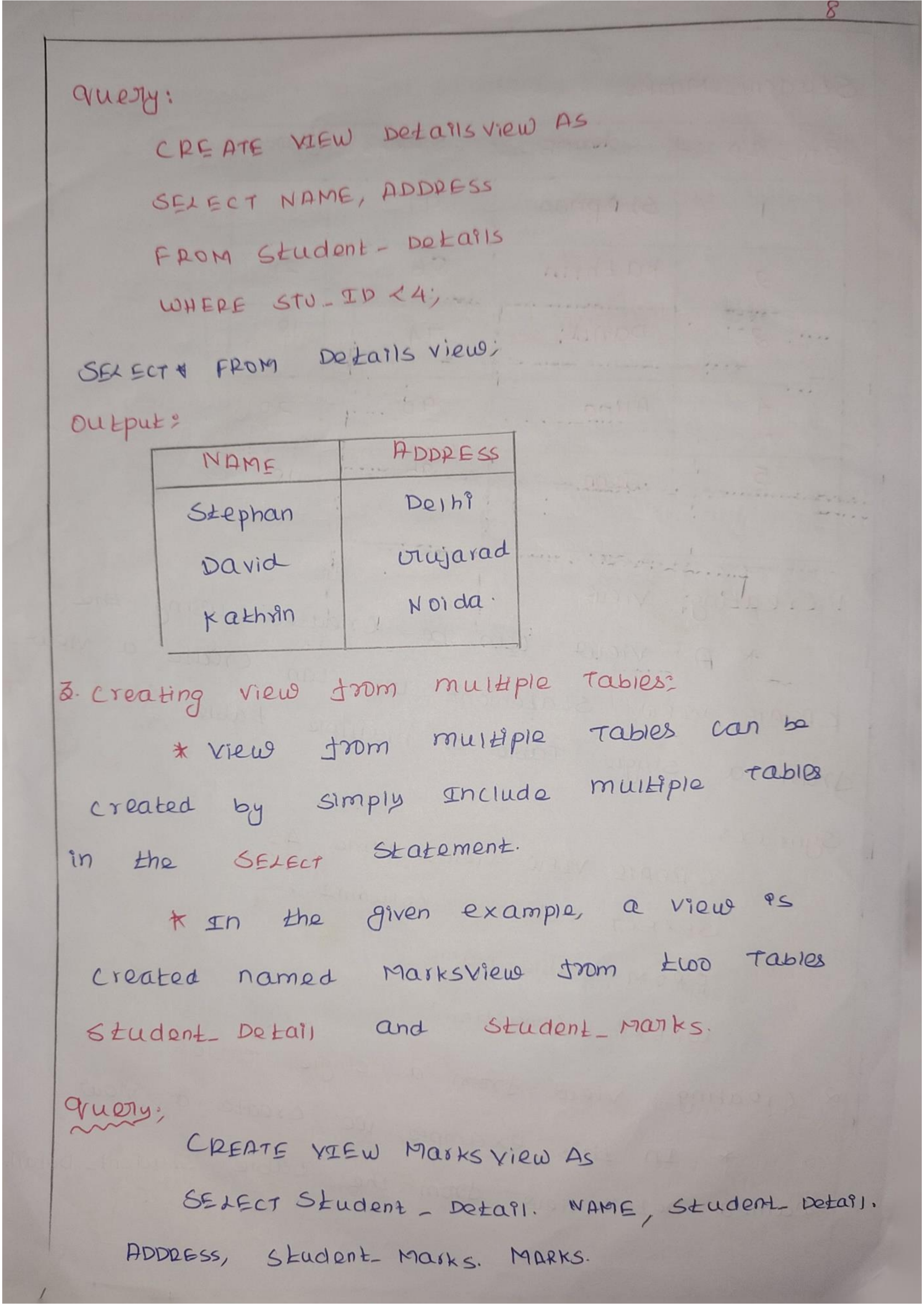
* A view can be created using the CREATE VIEW statement. we can create a view from a single Table or multiple tables.

Syntax:

```
CREATE VIEW view_ name AS
SELECT column1, column2...
FROM table_name
WHERE condition.
```

2. creating view from a single Table:

* In this example, we create a view named Details view from the table Student_Detail.



query:

```
CREATE VIEW Details view AS
SELECT NAME, ADDRESS
FROM Student_Details
WHERE STU_ID <4;
```

```
SELECT * FROM Details view;
```

output:

NAME	ADDRESS
Stephan	Delhi
David	Gujarad
Kathrin	Noida.

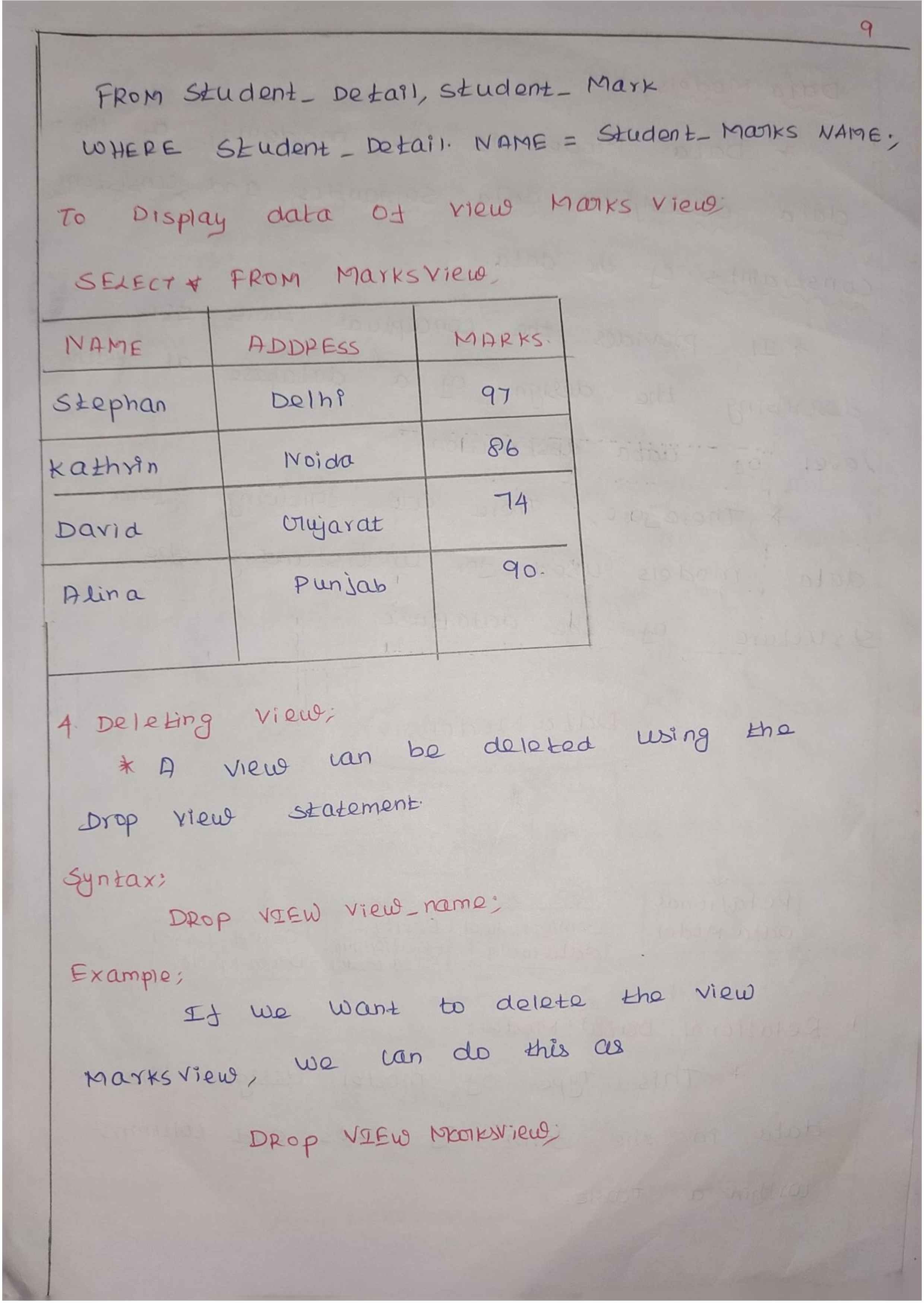
3. Creating view from multiple Tables:

* View from multiple Tables can be created by simply Include multiple tables in the SELECT statement.

* In the given example, a view is created named MarksView from two Tables Student_Detail and Student_Marks.

query;

```
CREATE VIEW MarksView AS
SELECT Student_Detail.NAME, Student_Detail.
ADDRESS, Student_Marks.MARKS.
```



9

```
FROM Student_Detail, student_Mark
WHERE Student_Detail. NAME = Student_Marks NAME;
```

To Display data of view Marks view

```
SELECT * FROM MarksView;
```

NAME	ADDRESS	MARKS.
Stephan	Delhi	97
kathrin	Noida	86
David	Gujarat	74
Alina	Punjab	90.

4. Deleting view;

* A view can be deleted using the Drop view statement.

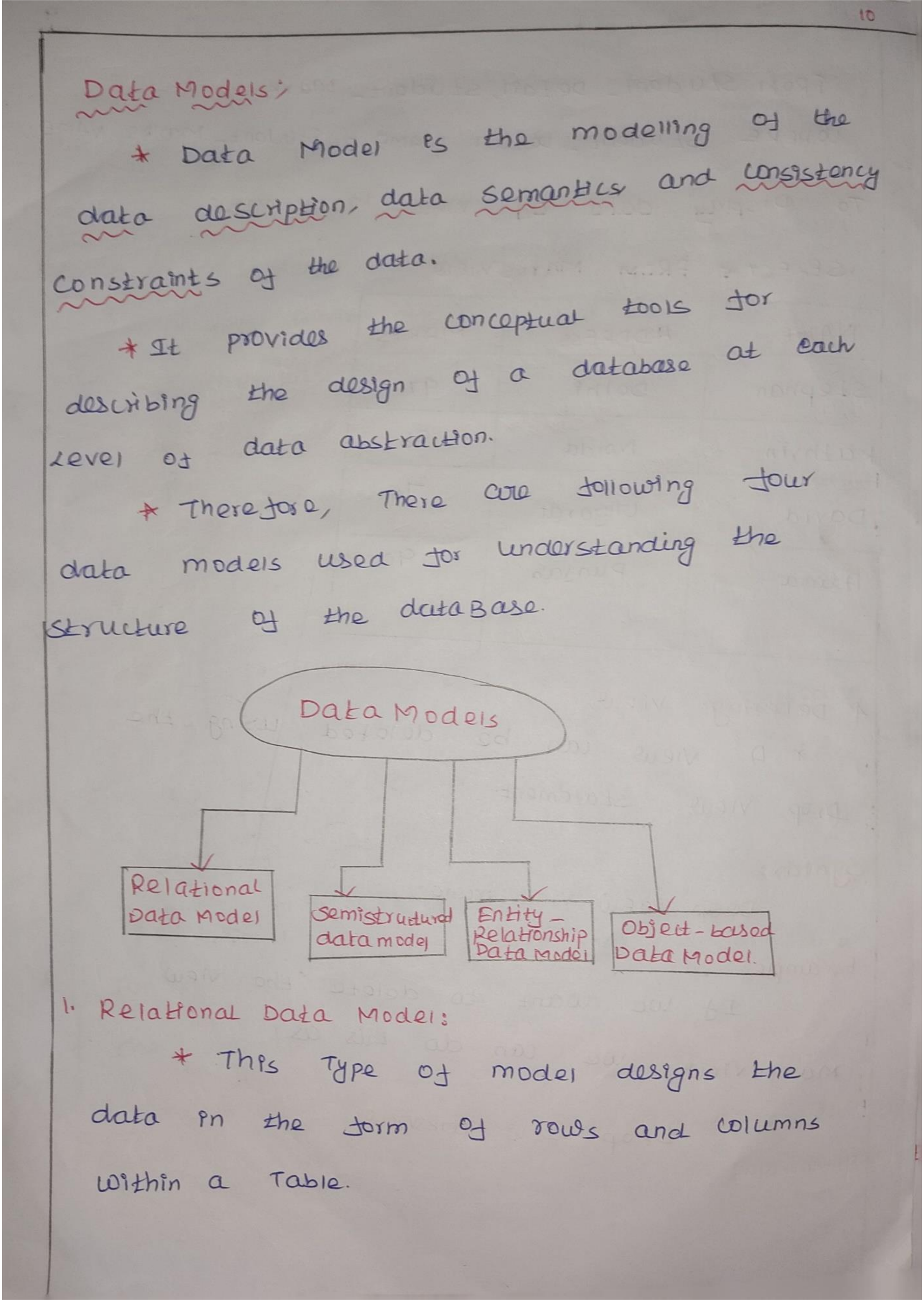
Syntax;

```
DROP VIEW view_name;
```

Example;

If we want to delete the view MarksView, we can do this as

```
DROP VIEW MarksView;
```



Data Models:

* Data Model is the modelling of the data description, data semantics and consistency constraints of the data.

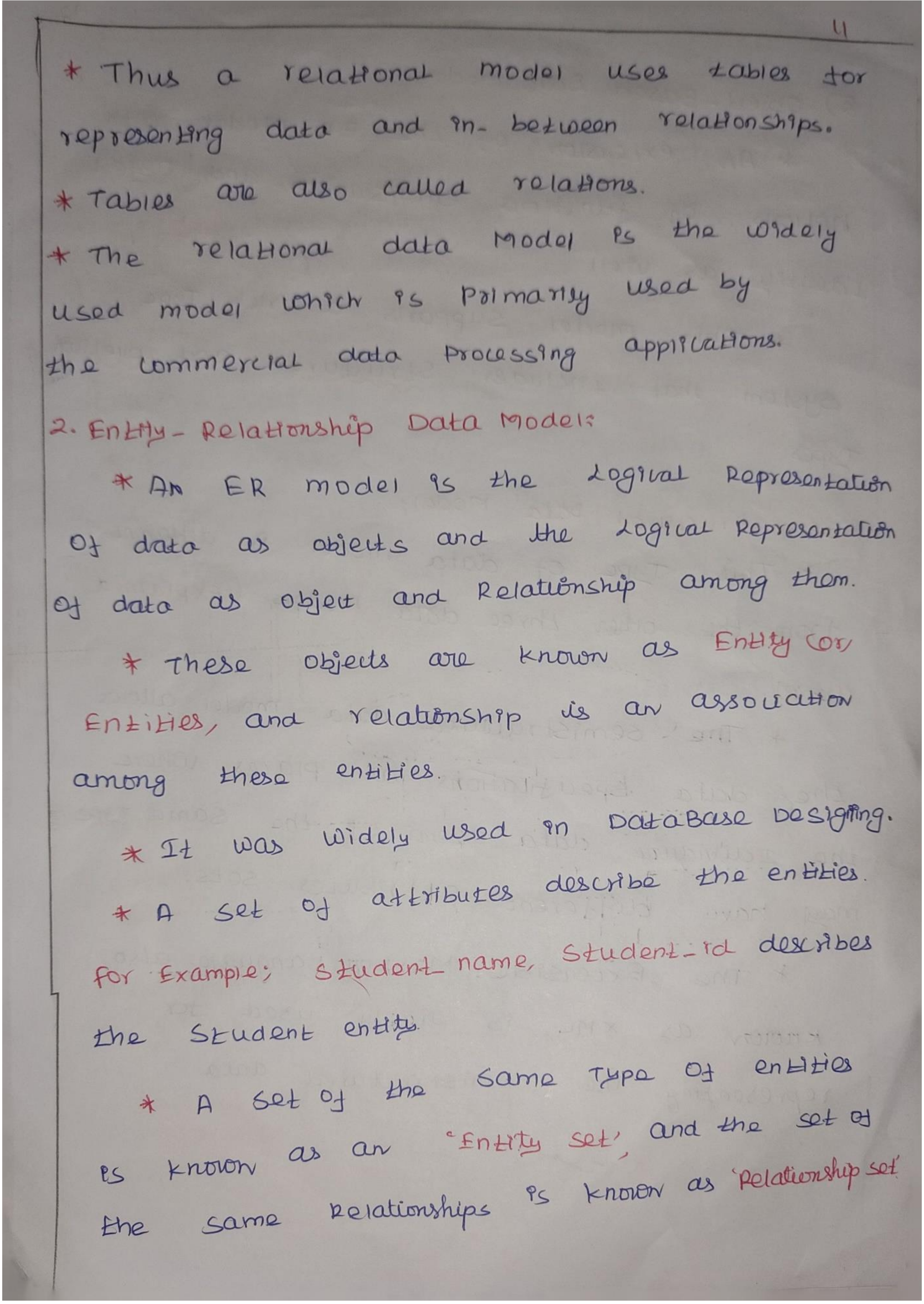
* It provides the conceptual tools for describing the design of a database at each level of data abstraction.

* Therefore, There are following four data models used for understanding the structure of the dataBase.

Data Models
Relational Data Model
Semistructured data model
Entity - Relationship Data Model
Object - based Data Model.

1. Relational Data Model:

* This Type of model designs the data in the form of rows and columns within a Table.



* Thus a relational model uses tables for representing data and in- between relationships.

* Tables are also called relations.

* The relational data Model is the widely used model which is primarily used by the commercial data processing applications.

2. Entity- Relationship Data Model:

* An ER model is the logical Representation of data as objects and the logical Representation of data as object and Relationship among them.

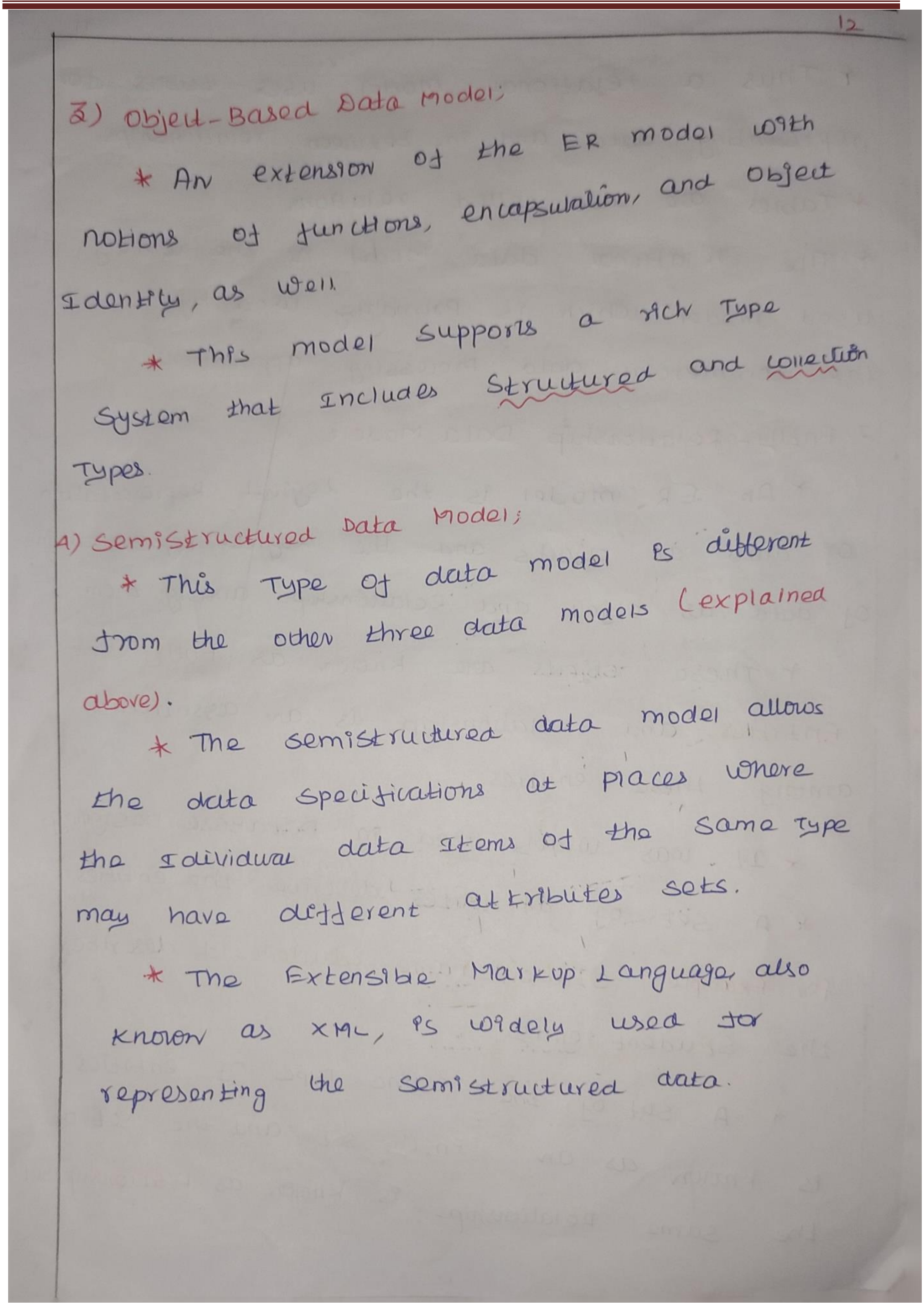
* These objects are known as Entity (or) Entities, and relationship is an association among these entities.

* It was widely used in DataBase Designing.

* A set of attributes describe the entities.

for Example; student_name, student_id describes the student entity.

* A set of the same Type of entities is known as an 'Entity set', and the set of the same Relationships is known as 'Relationship set'



3) Object-Based Data Model;

* An extension of the ER model with notions of functions, encapsulation, and object Identity, as well.

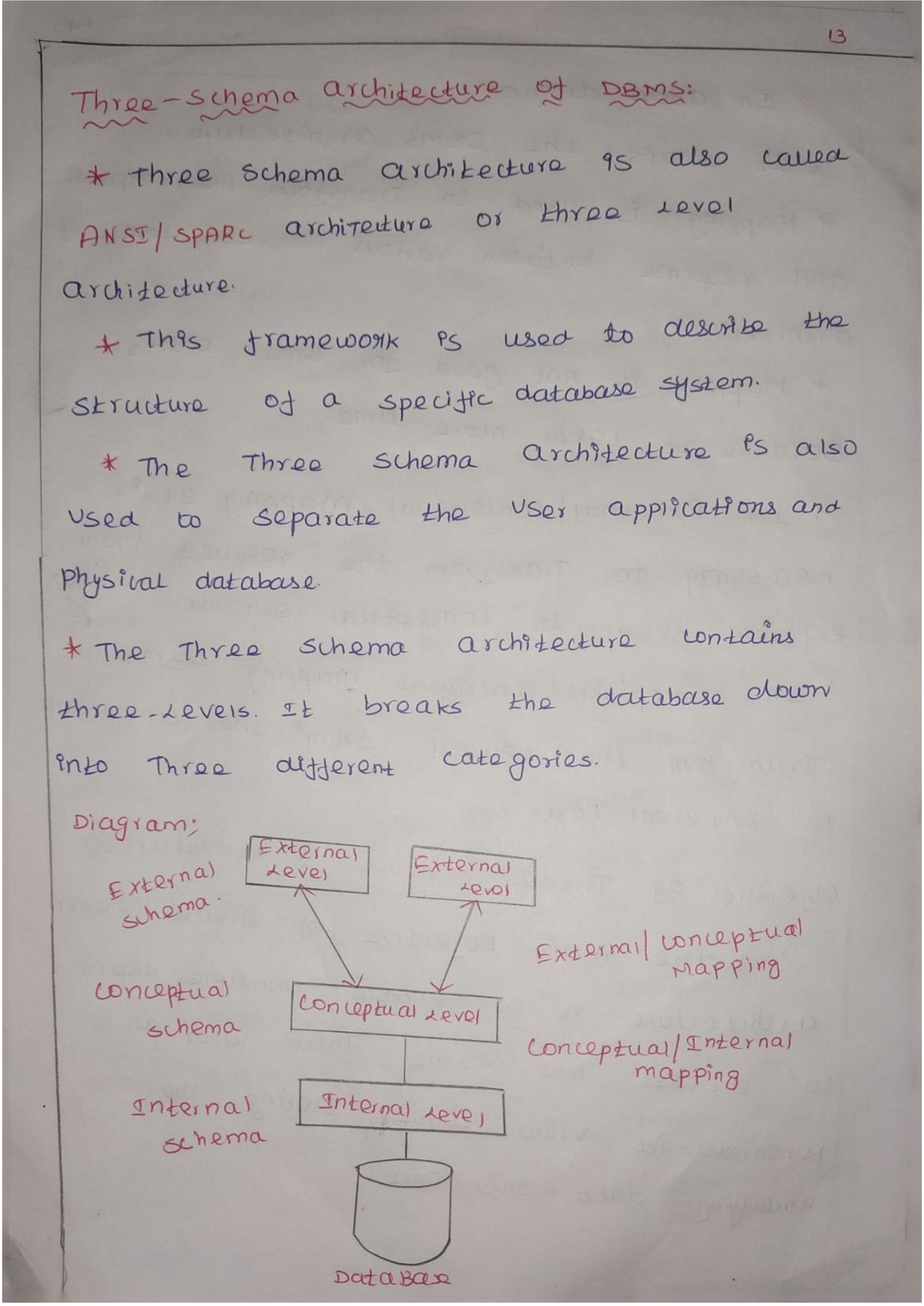
* This model supports a rich Type system that Includes Structured and collection Types.

4) Semistructured Data Model;

* This Type of data model is different from the other three data models (explained above).

* The semistructured data model allows the data specifications at places where the Idividual data Items of the same Type may have different attributes sets.

* The Extensible Markup Language, also known as XML, is widely used for representing the semistructured data.



Three-schema architecture of DBMS:

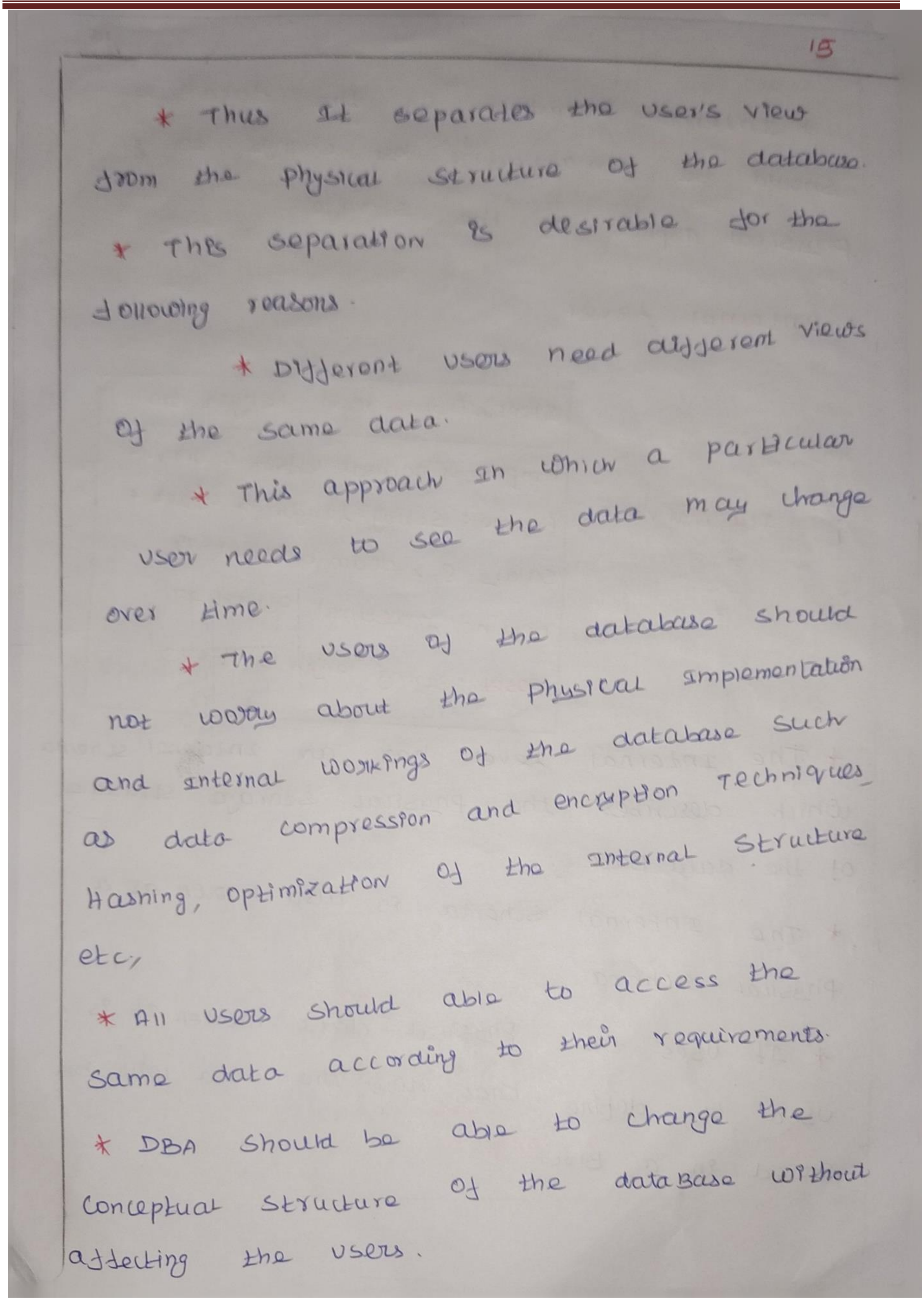
* Three Schema architecture is also called ANSI/SPARC architecture or three level architecture.

* This framework is used to describe the structure of a specific database system.

* The Three schema architecture is also used to separate the user applications and physical database.

* The Three schema architecture contains three-levels. It breaks the database down into Three different categories.

Diagram:



15

- * Thus it separates the user's view from the physical structure of the database.

- * This separation is desirable for the following reasons.

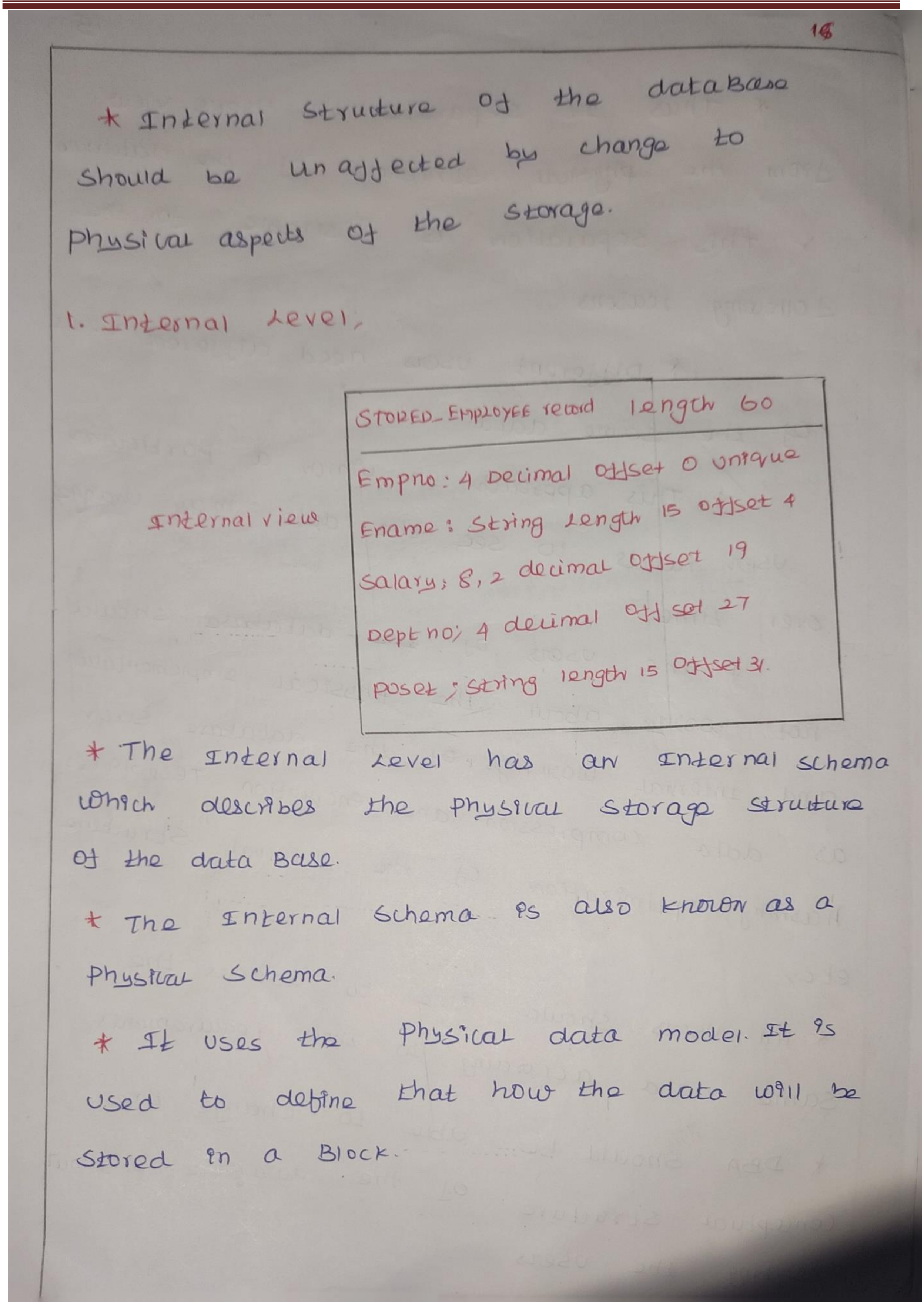
  - * Different users need different views of the same data.

  - * This approach in which a particular user needs to see the data may change over time.

  - * The users of the database should not worry about the physical implementation and internal workings of the database such as data compression and encryption techniques Hashing, optimization of the internal structure etc,

- * All users should able to access the same data according to their requirements.

- * DBA should be able to change the conceptual structure of the database without affecting the users.



* Internal Structure of the database should be unaffected by change to physical aspects of the storage.

1. Internal Level,

Internal view

STORED_EMPLOYEE record length 60

Empno: 4 Decimal offset 0 unique

Ename: String length 15 offset 4

Salary; 8, 2 decimal offset 19

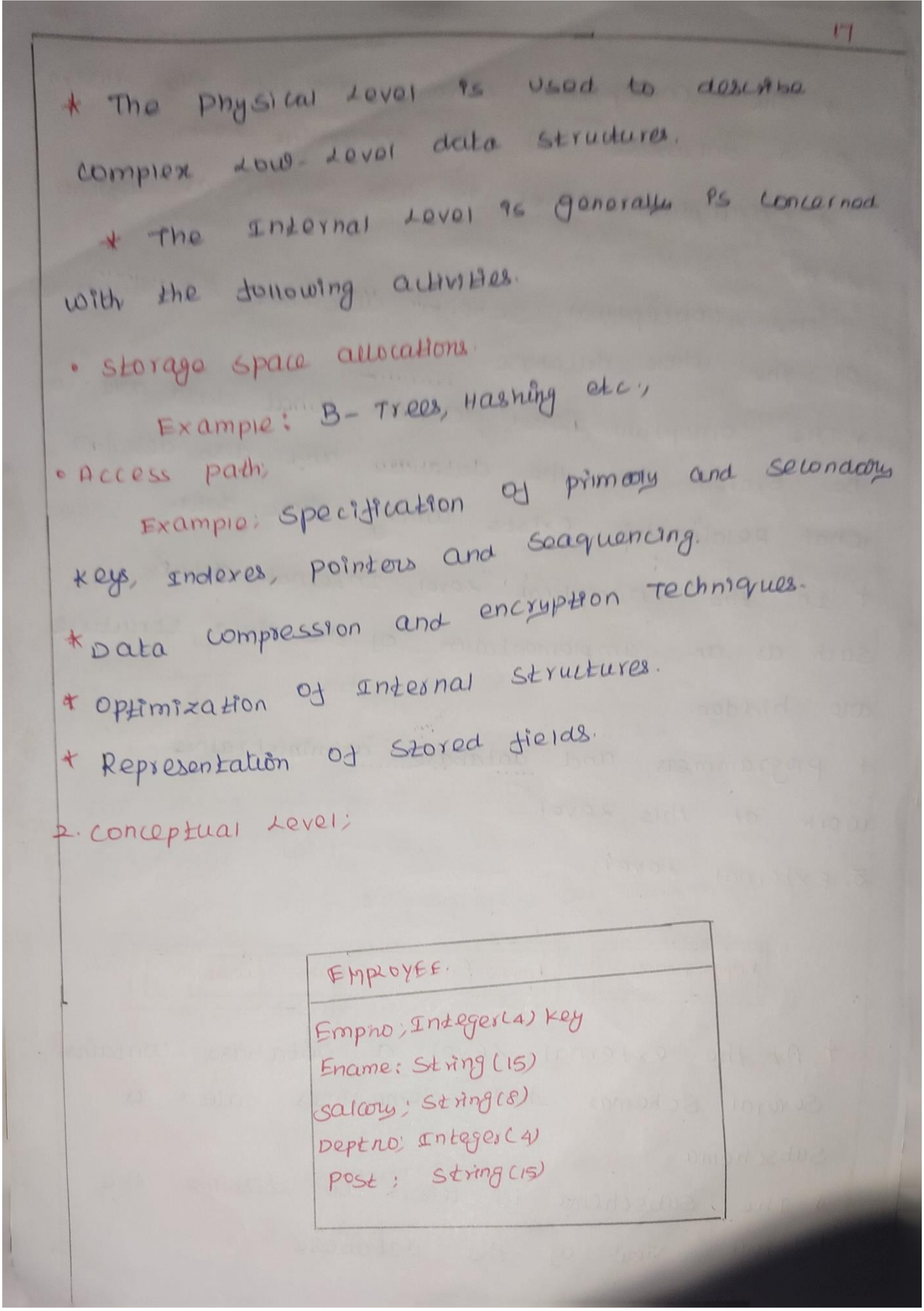
Dept no; 4 decimal offset 27

poset ; String length 15 offset 31.

* The Internal Level has an Internal schema which describes the physical storage structure of the data Base.

* The Internal Schema is also known as a physical Schema.

* It uses the physical data model. It is used to define that how the data will be stored in a Block.



17

* The physical level is used to describe complex low-level data structures.

* The internal level is generally is concerned with the following activities.

- Storage space allocations

  Example: B- Trees, Hashing etc.,

- Access path;

  Example; Specification of primary and secondary keys, indexes, pointers and seaquencing.

* Data compression and encryption techniques.

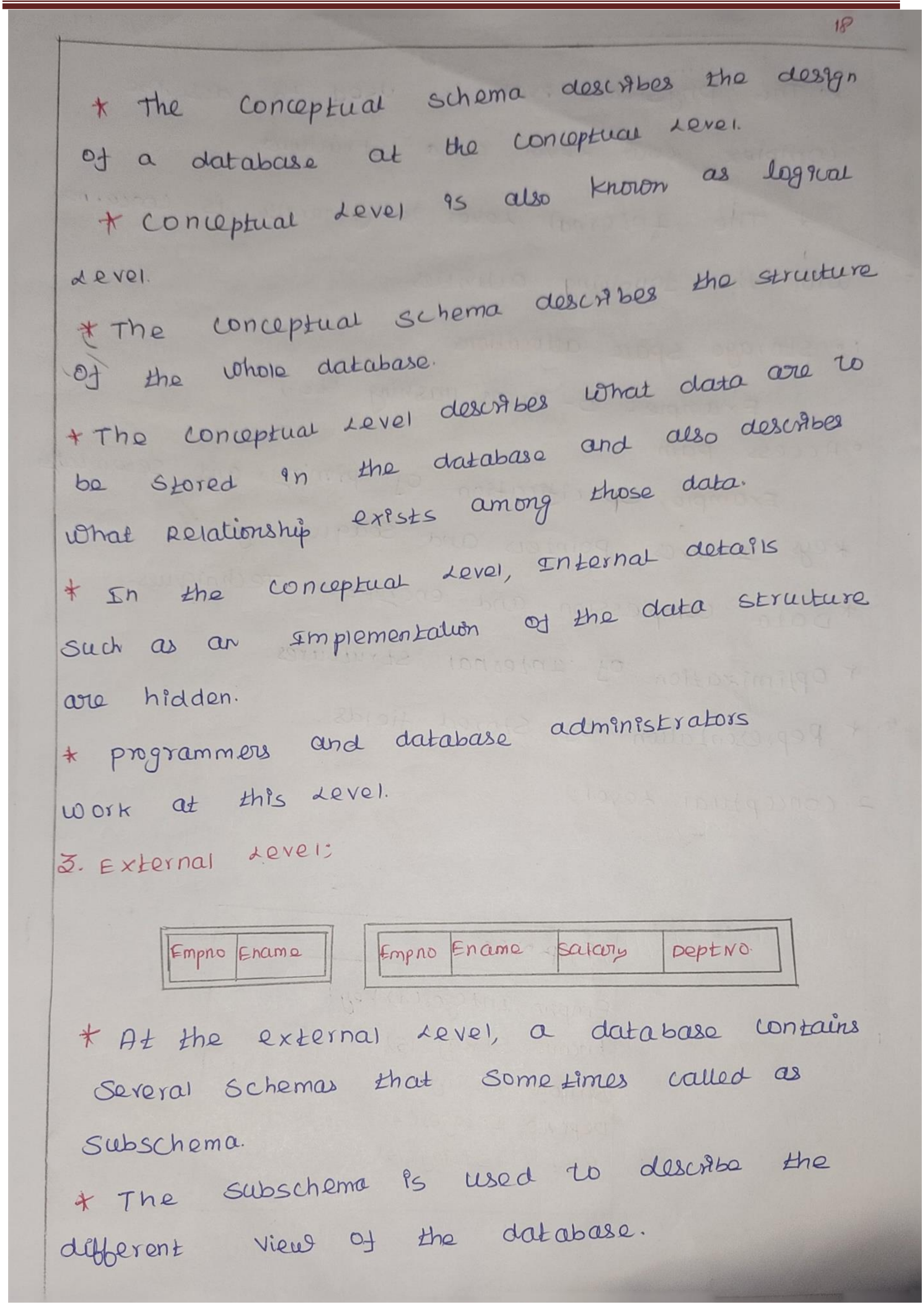
* Optimization of internal structures.

* Representation of stored fields.

2. Conceptual Level;

EMPLOYEE.

Empno; Integer(4) key
Ename: String (15)
salary; String(8)
Deptno; Integer(4)
post ; String (15)



* The conceptual schema describes the design of a database at the conceptual level.

* Conceptual level is also known as logical level.

* The conceptual schema describes the structure of the whole database.

* The conceptual level describes what data are to be stored in the database and also describes what relationship exists among those data.

* In the conceptual level, internal details such as an implementation of the data structure are hidden.

* programmers and database administrators work at this level.

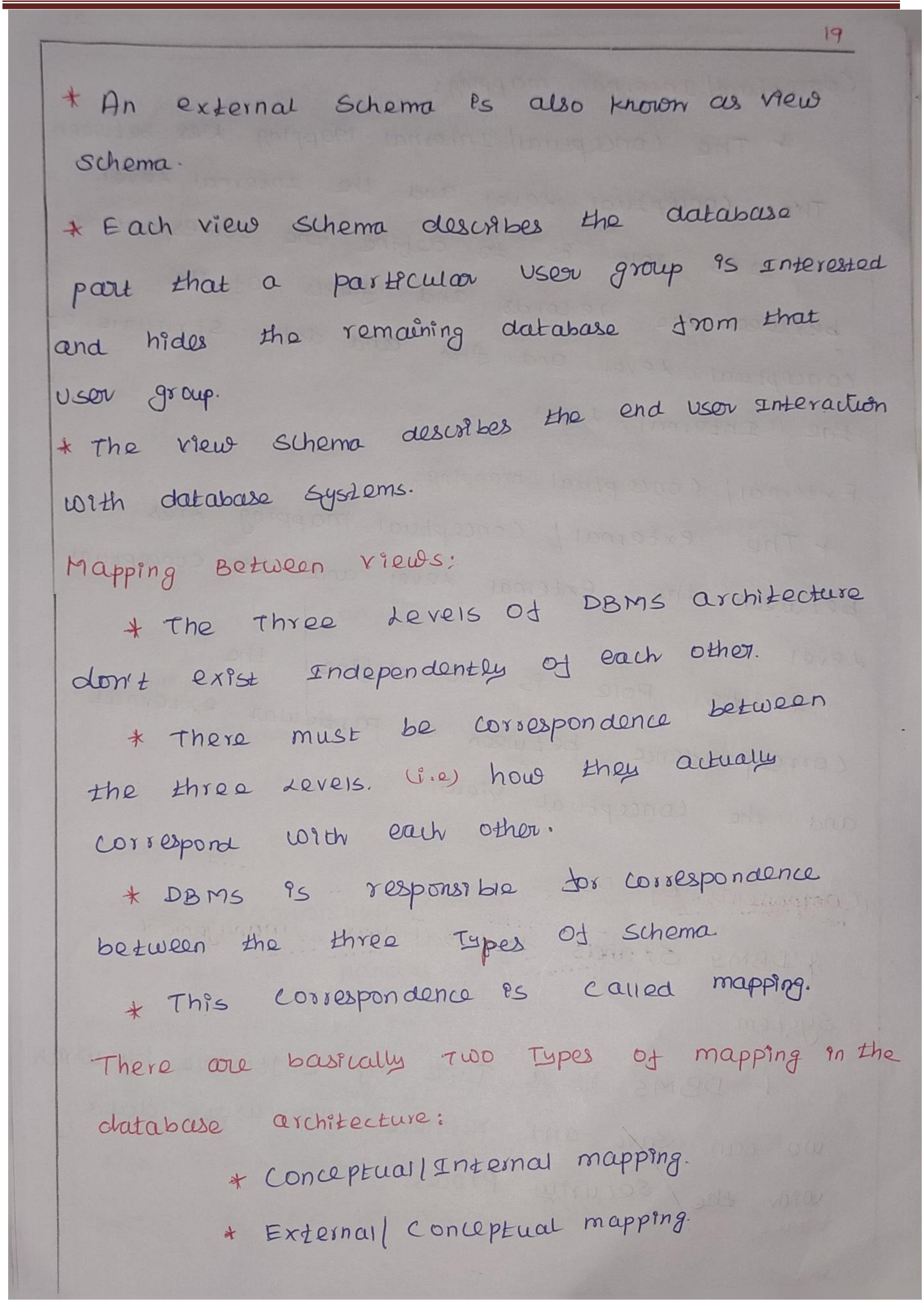
3. External level:

Empno	Ename

Empno	Ename	Salary	DeptNo.

* At the external level, a database contains several schemas that sometimes called as subschema.

* The subschema is used to describe the different view of the database.



* An external Schema is also known as view schema.

* Each view schema describes the database part that a particular user group is interested and hides the remaining database from that user group.

* The view schema describes the end user interaction with database systems.

Mapping Between views:

* The Three Levels of DBMS architecture don't exist Independently of each other.

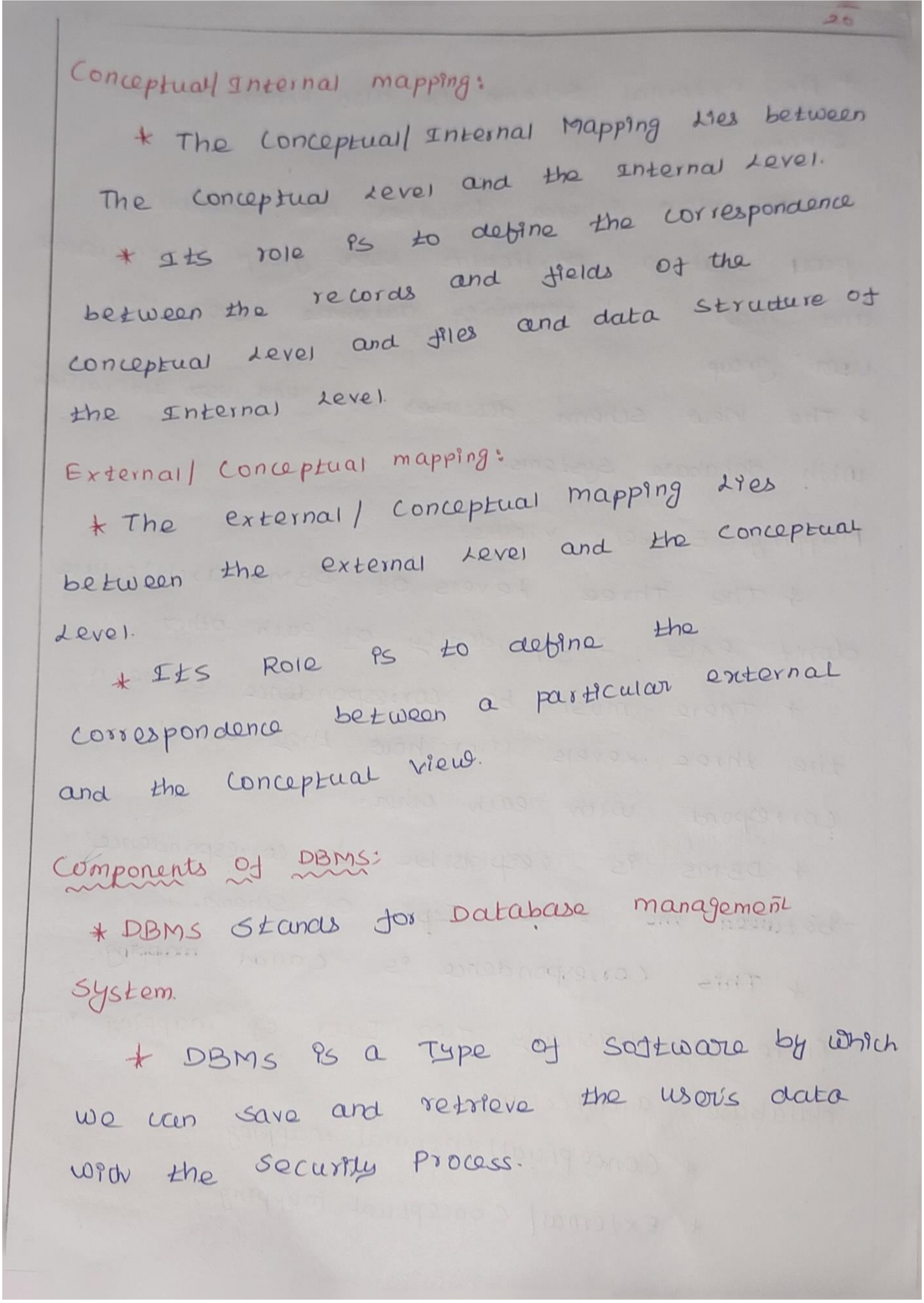
* There must be correspondence between the three levels. (i.e) how they actually correspond with each other.

* DBMS is responsible for correspondence between the three Types of schema

* This correspondence is called mapping.

There are basically Two Types of mapping in the database architecture:

* Conceptual/Internal mapping.
* External/ Conceptual mapping.



Conceptual/Internal mapping:

* The Conceptual/ Internal Mapping lies between The conceptual level and the internal level.

* Its role is to define the correspondence between the records and fields of the conceptual level and files and data structure of the Internal level.

External/ Conceptual mapping:

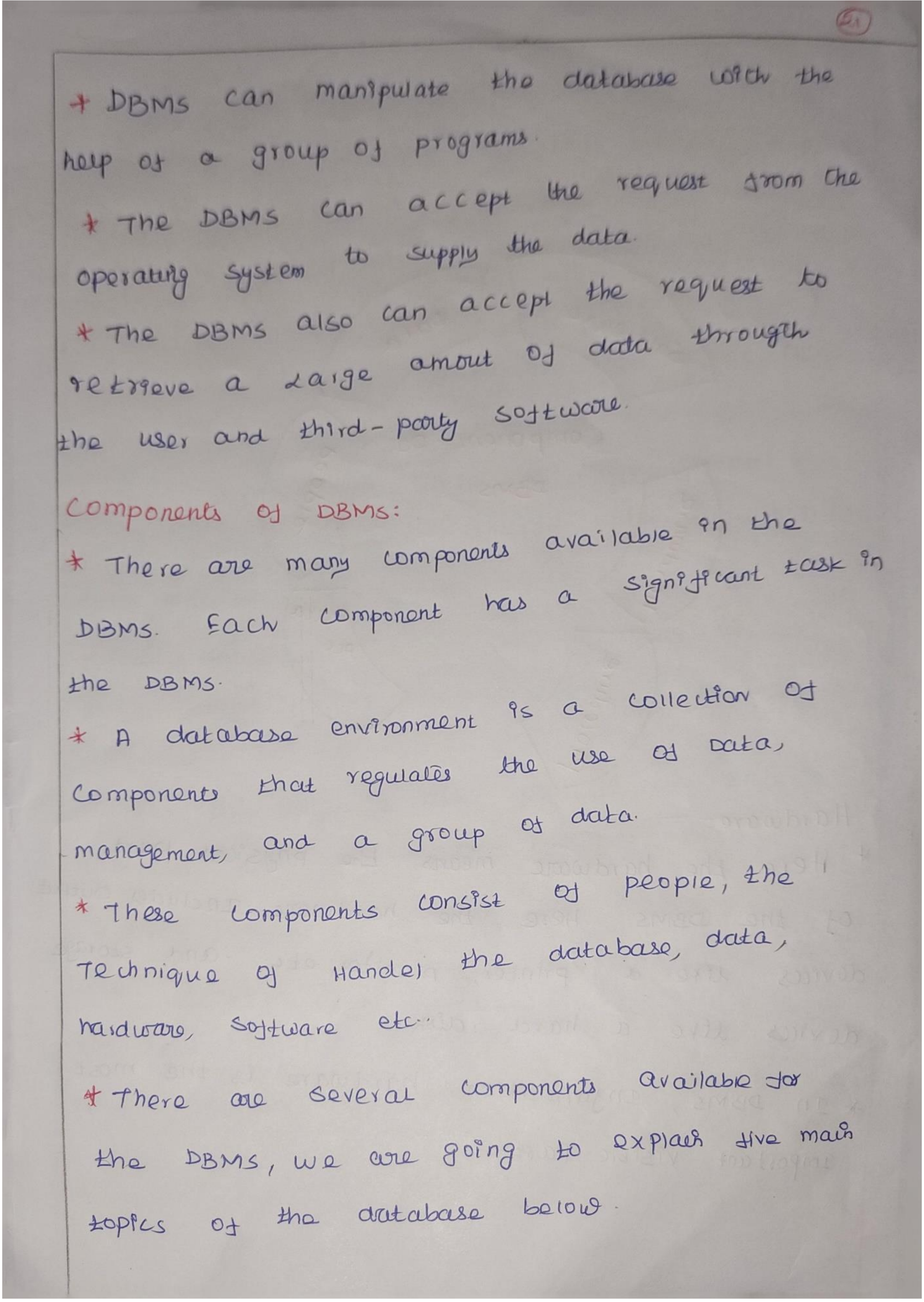
* The external/ conceptual mapping lies between the external level and the conceptual level.

* Its Role is to define the correspondence between a particular external and the conceptual view.

Components of DBMS:

* DBMS stands for Database management System.

* DBMS is a Type of Software by which we can save and retrieve the user's data with the security Process.



* DBMS can manipulate the database with the help of a group of programs.

* The DBMS can accept the request from the operating system to supply the data.

* The DBMS also can accept the request to retrieve a Laige amout of data through the user and third-party software.

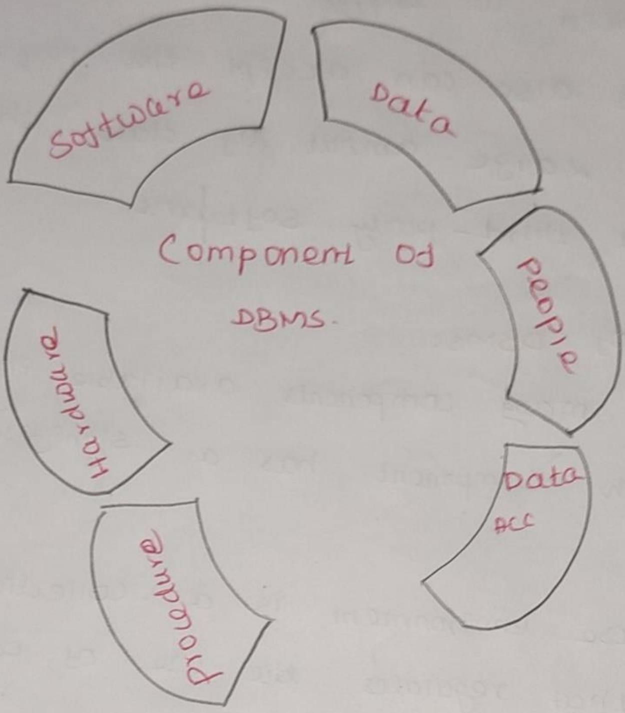
Components of DBMS:

* There are many components available in the DBMS. Each component has a significant task in the DBMS.

* A database environment is a collection of components that regulates the use of Data, management, and a group of data.

* These components consist of people, the Technique of Handel the database, data, hardware, Software etc..

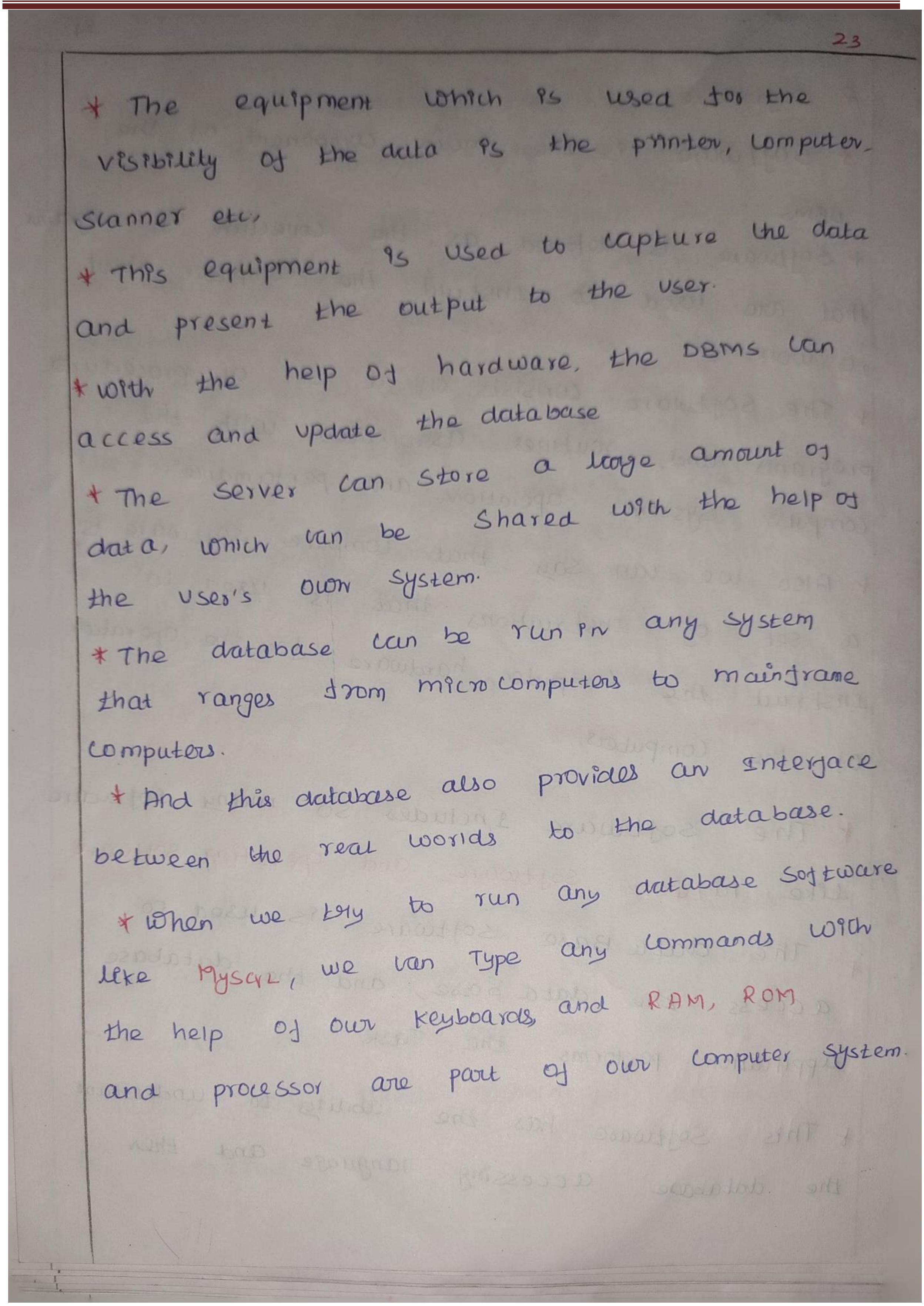
* There are several components available for the DBMS, we are going to explain five main topics of the database below.



Hardware;

* Here the hardware means the physical part of the DBMS. Here the hardware Include output devices like a printer, monitor, etc., and storage devices like a hard disk.

* In DBMS, Information hardware is the most Important visible Part.



* The equipment which is used for the visibility of the data is the printer, computer, scanner etc,

* This equipment is used to capture the data and present the output to the user.

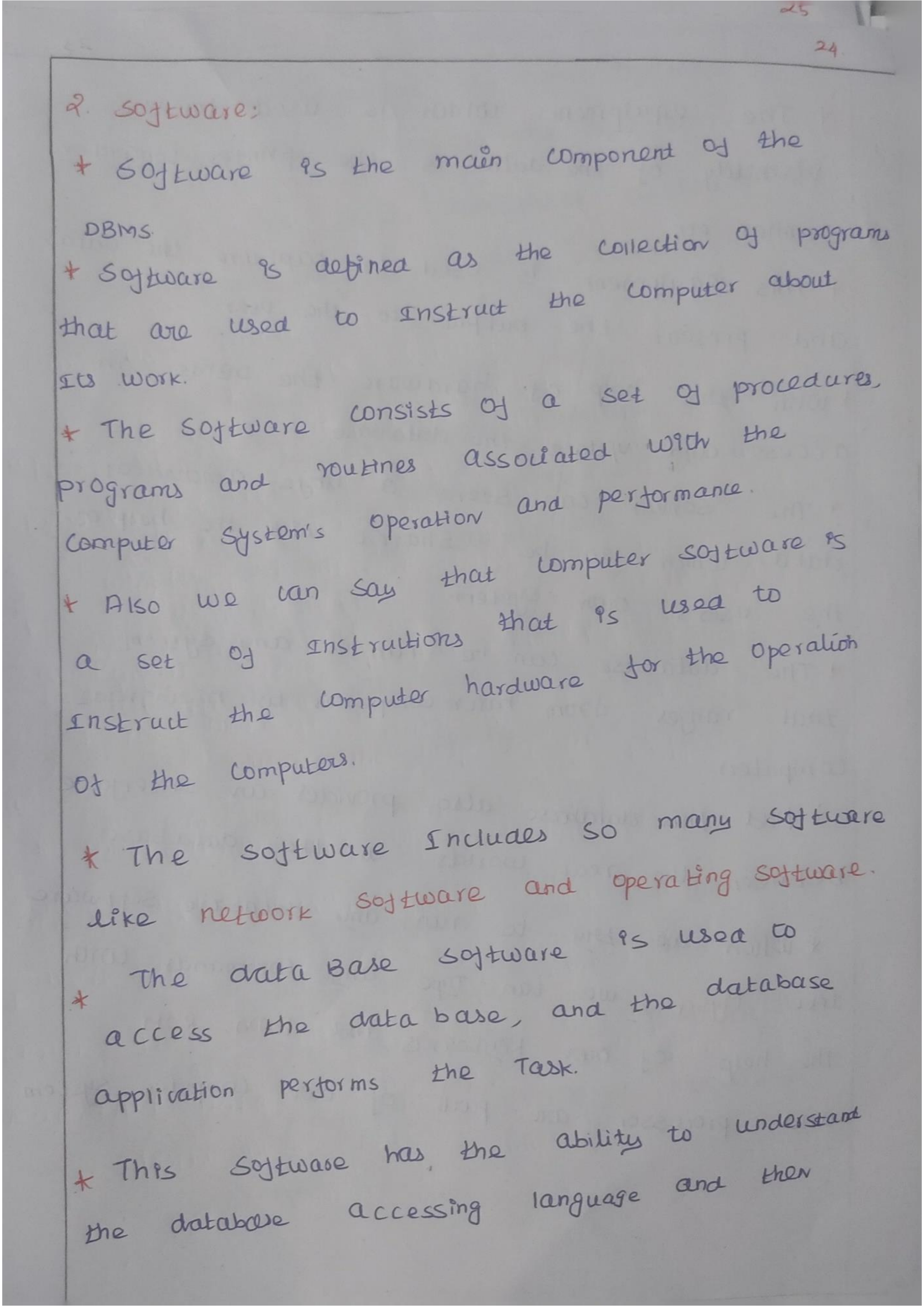
* with the help of hardware, the DBMS can access and update the database

* The server can store a large amount of data, which can be Shared with the help of the user's own system.

* The database can be run in any system that ranges from micro computers to mainframe computers.

* And this database also provides an interface between the real worlds to the database.

* When we try to run any database software like MySQL, we can Type any commands with the help of our keyboards, and RAM, ROM and processor are part of our computer system.

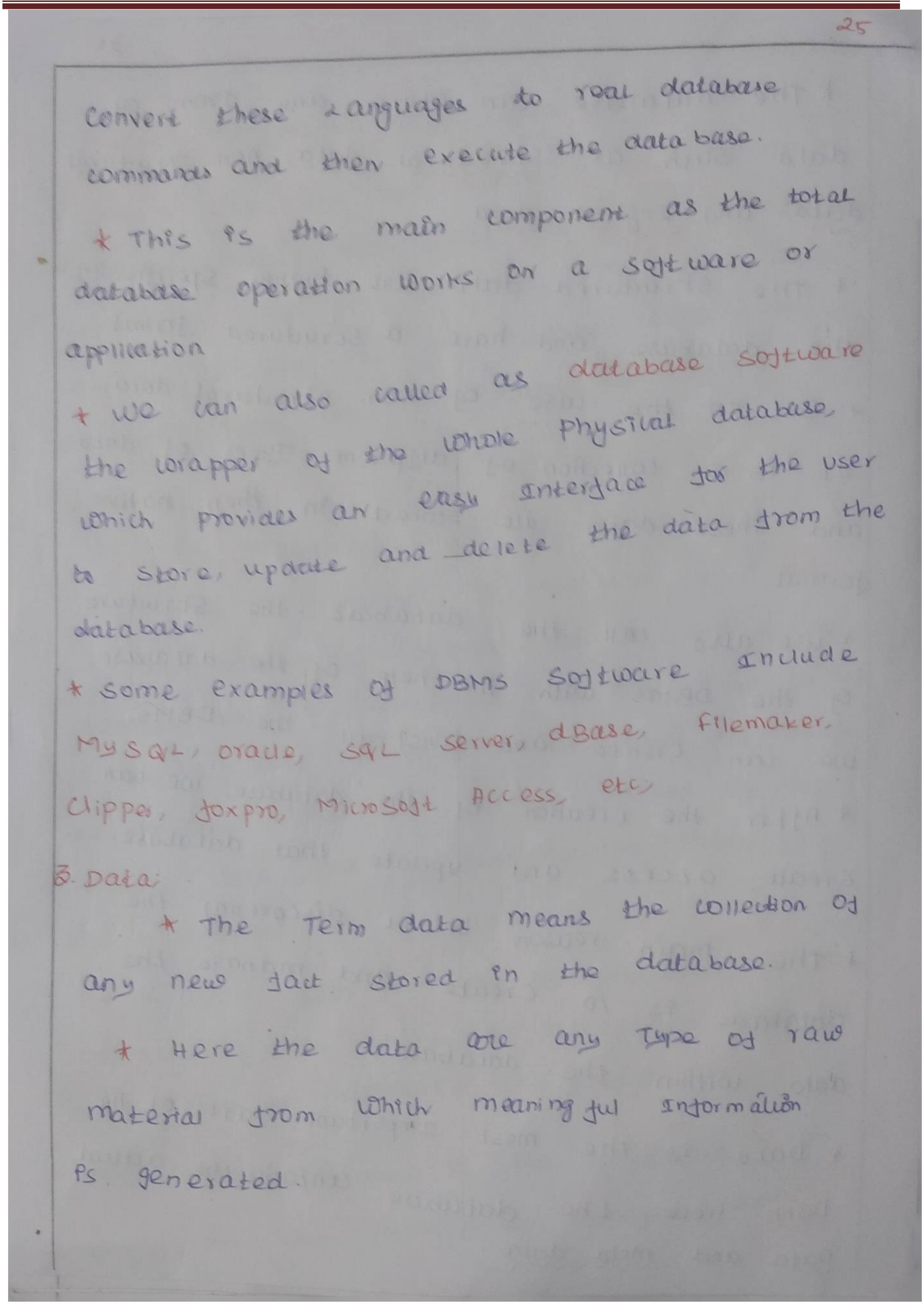


25

24

2. Software:

* Software is the main component of the DBMS.
* Software is defined as the collection of programs that are used to instruct the computer about its work.
* The Software consists of a set of procedures, programs and routines associated with the computer system's operation and performance.
* Also we can say that computer software is a set of instructions that is used to instruct the computer hardware for the operation of the computers.
* The software Includes so many software like network software and operating software.
* The data Base software is used to access the data base, and the database application performs the Task.
* This Software has the ability to understand the database accessing language and then



Convert these languages to real database commands and then execute the data base.

* This is the main component as the total database operation works on a software or application

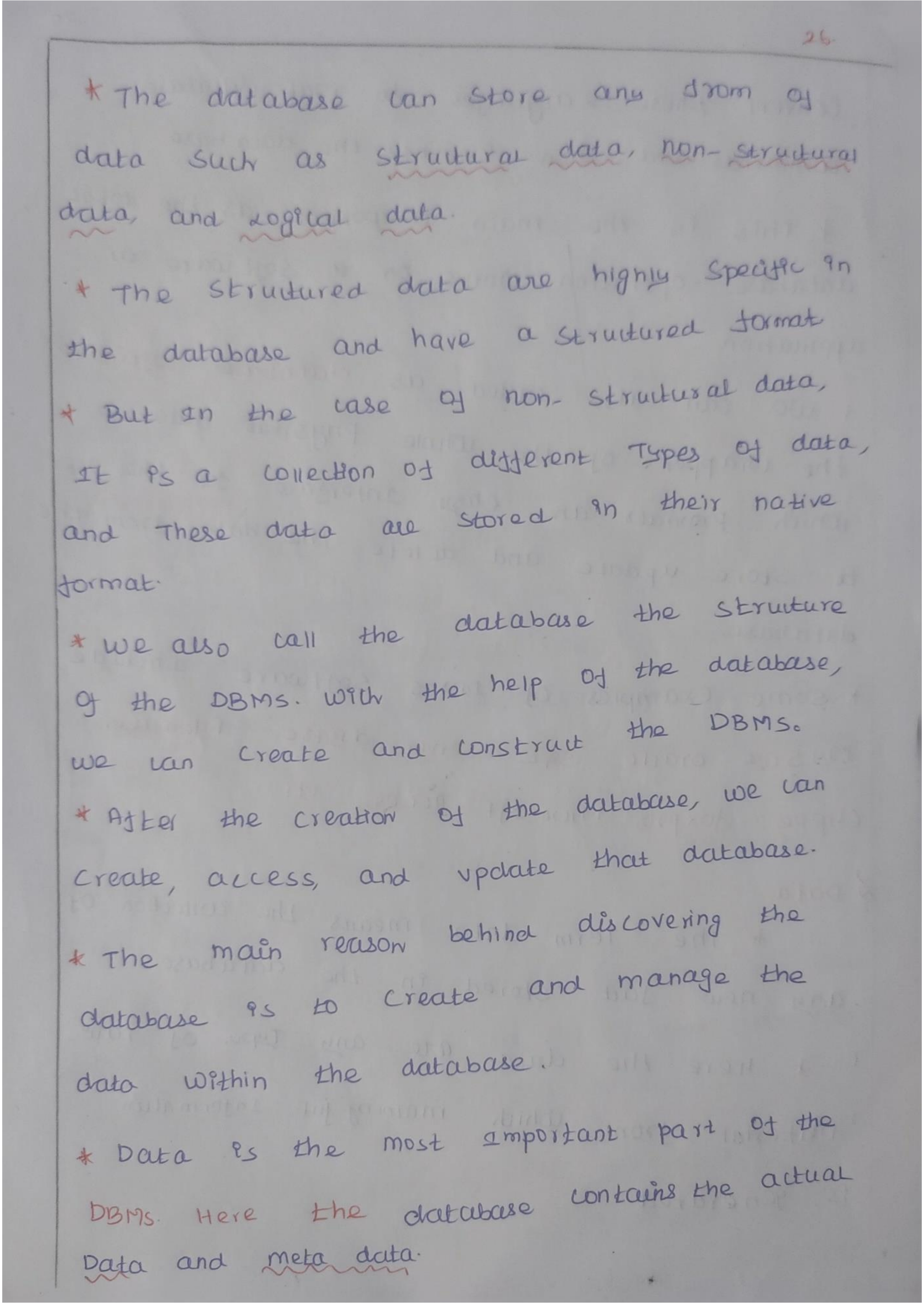
* We can also called as database software the wrapper of the whole physical database, which provides an easy interface for the user to store, update and delete the data from the database.

* Some examples of DBMS software include MySQL, oracle, SQL server, dBase, Filemaker, Clipper, foxpro, Microsoft Access, etc,

3. Data:

* The Term data means the collection of any new fact stored in the database.

* Here the data are any Type of raw material from which meaningful information is generated.



* The database can store any from of data such as structural data, non-structural data, and logical data.

* The structured data are highly specific in the database and have a structured format.

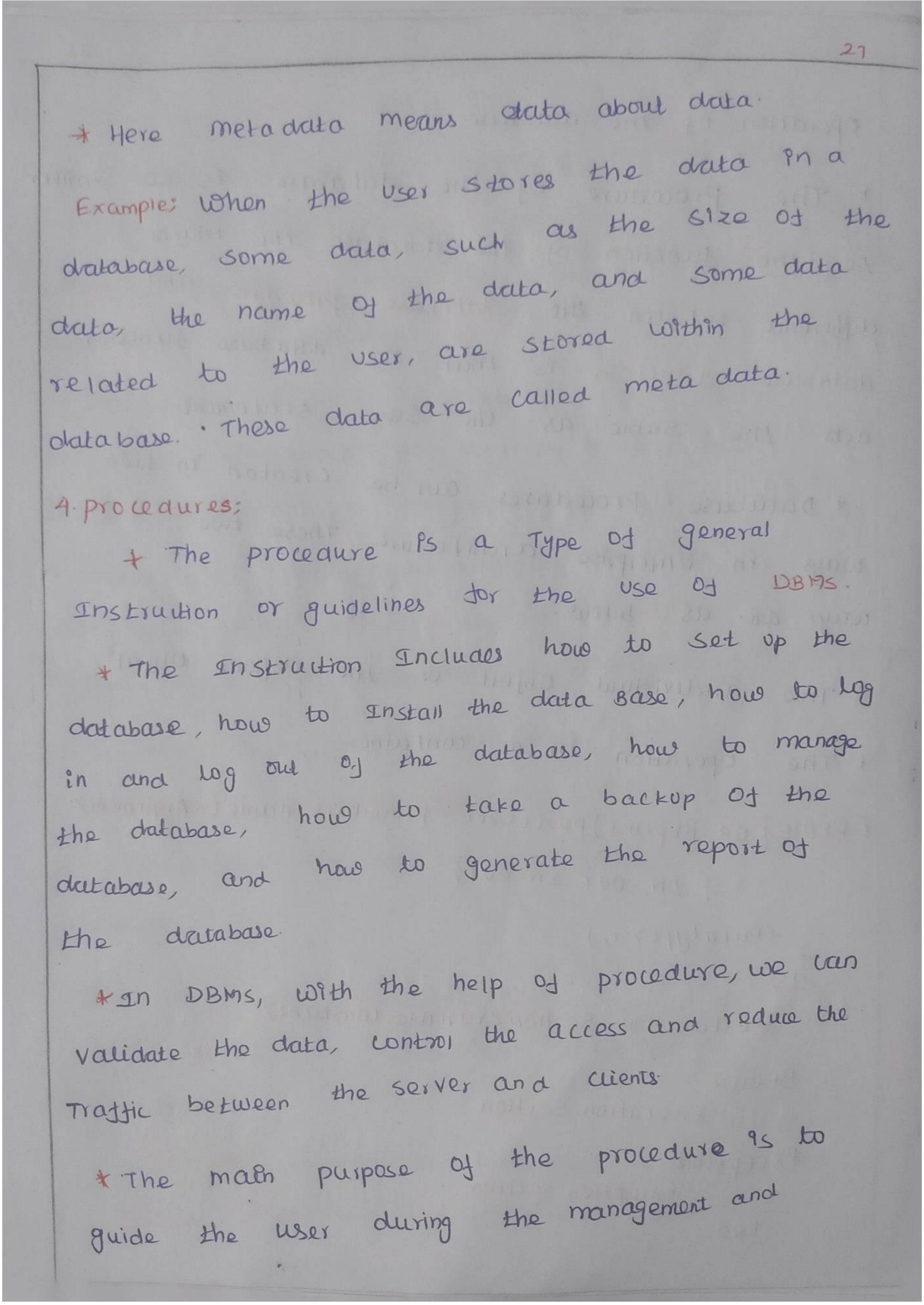
* But in the case of non-structural data, It is a collection of different Types of data, and These data are stored in their native format.

* we also call the database the structure of the DBMS. with the help of the database, we can create and construct the DBMS.

* After the creation of the database, we can create, access, and update that database.

* The main reason behind discovering the database is to create and manage the data within the database.

* Data is the most important part of the DBMS. Here the database contains the actual Data and meta data.



* Here meta data means data about data.

Example; When the user stores the data in a database, some data, such as the size of the data, the name of the data, and some data related to the user, are stored within the database. These data are called meta data.

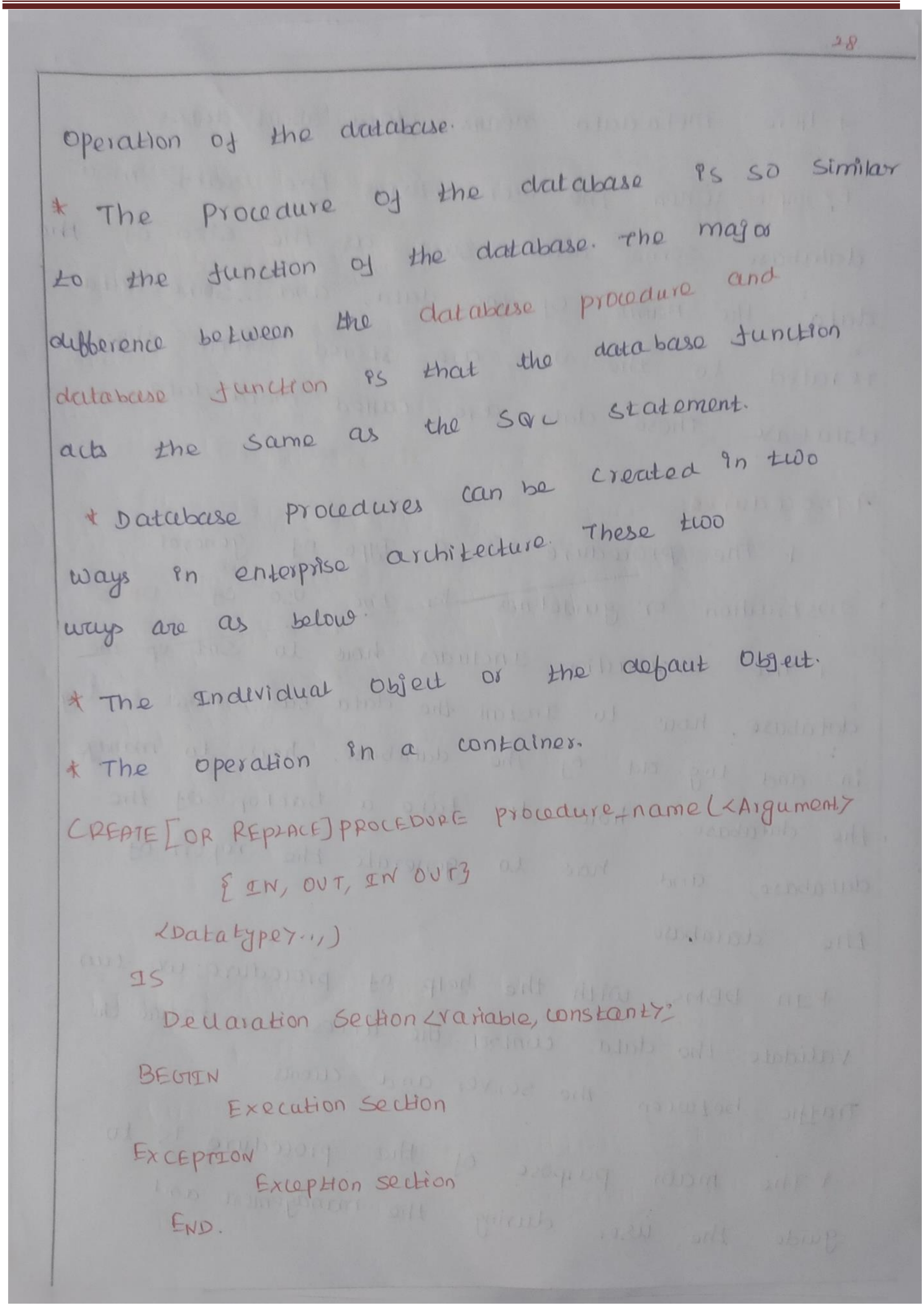
4. procedures;

+ The procedure is a Type of general Instruction or guidelines for the use of DBMS.

* The Instruction Includes how to set up the database, how to Install the data base, how to log in and log out of the database, how to manage the database, how to take a backup of the database, and how to generate the report of the database.

* In DBMS, with the help of procedure, we can validate the data, control the access and reduce the Traffic between the server and clients.

* The main purpose of the procedure is to guide the user during the management and



operation of the database.

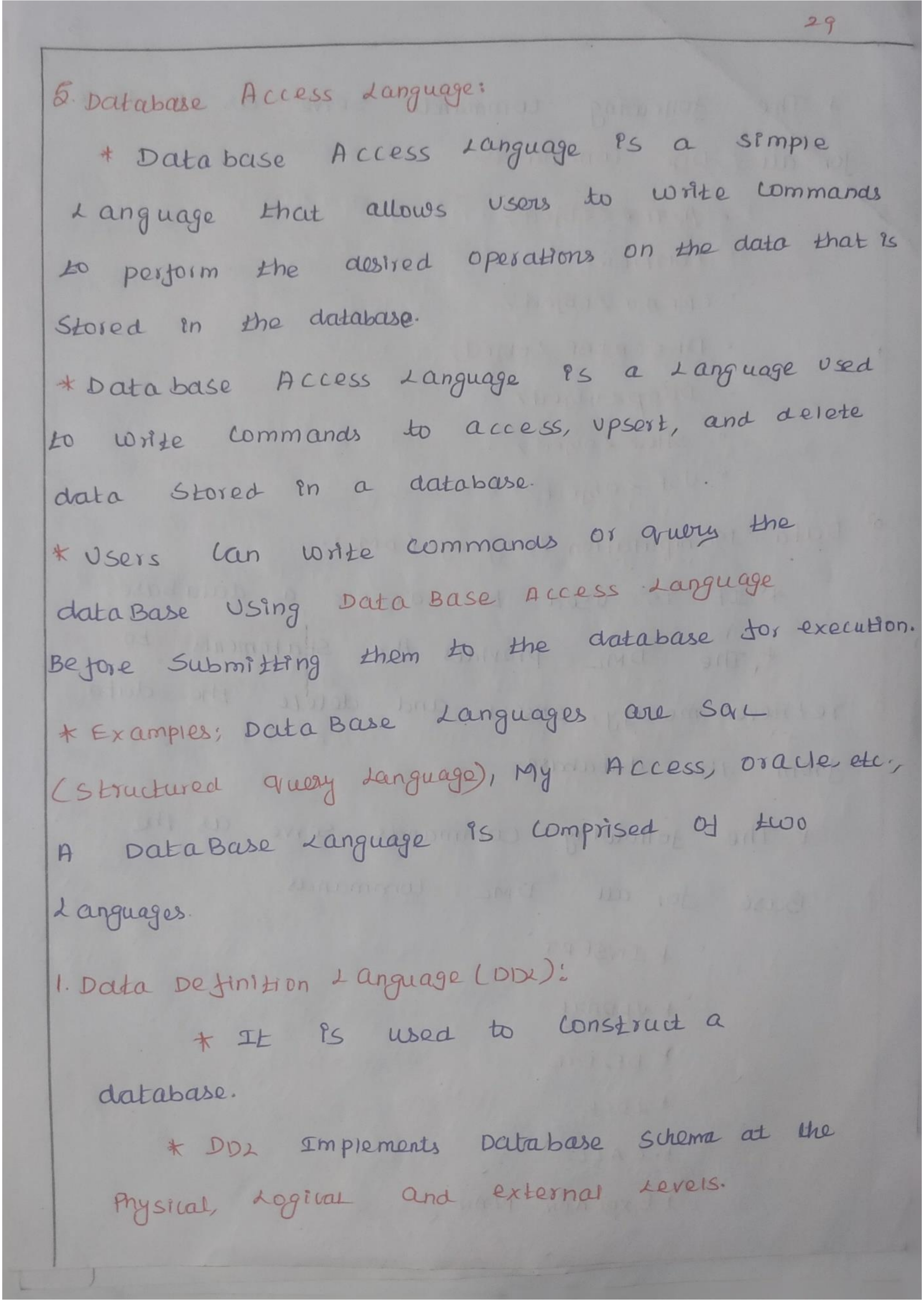
* The Procedure of the database is so similar to the function of the database. The major difference between the database procedure and database function is that the database function acts the same as the SQL statement.

* Database procedures can be created in two ways in enterprise architecture. These two ways are as below.

* The Individual object or the default object.

* The operation in a container.

```
CREATE [OR REPLACE] PROCEDURE procedure_name(<Argument>
           {IN, OUT, IN OUT}
     <Datatype>..,)
   IS
     Declaration Section<variable, constant>;
   BEGIN
          Execution Section
   EXCEPTION
            Exception section
     END.
```



29

5. Database Access Language:

* Database Access Language is a simple Language that allows users to write commands to perform the desired operations on the data that is stored in the database.

* Database Access Language is a Language used to write commands to access, upsert, and delete data stored in a database.

* Users can write commands or query the dataBase using Data Base Access Language Before Submitting them to the database for execution.

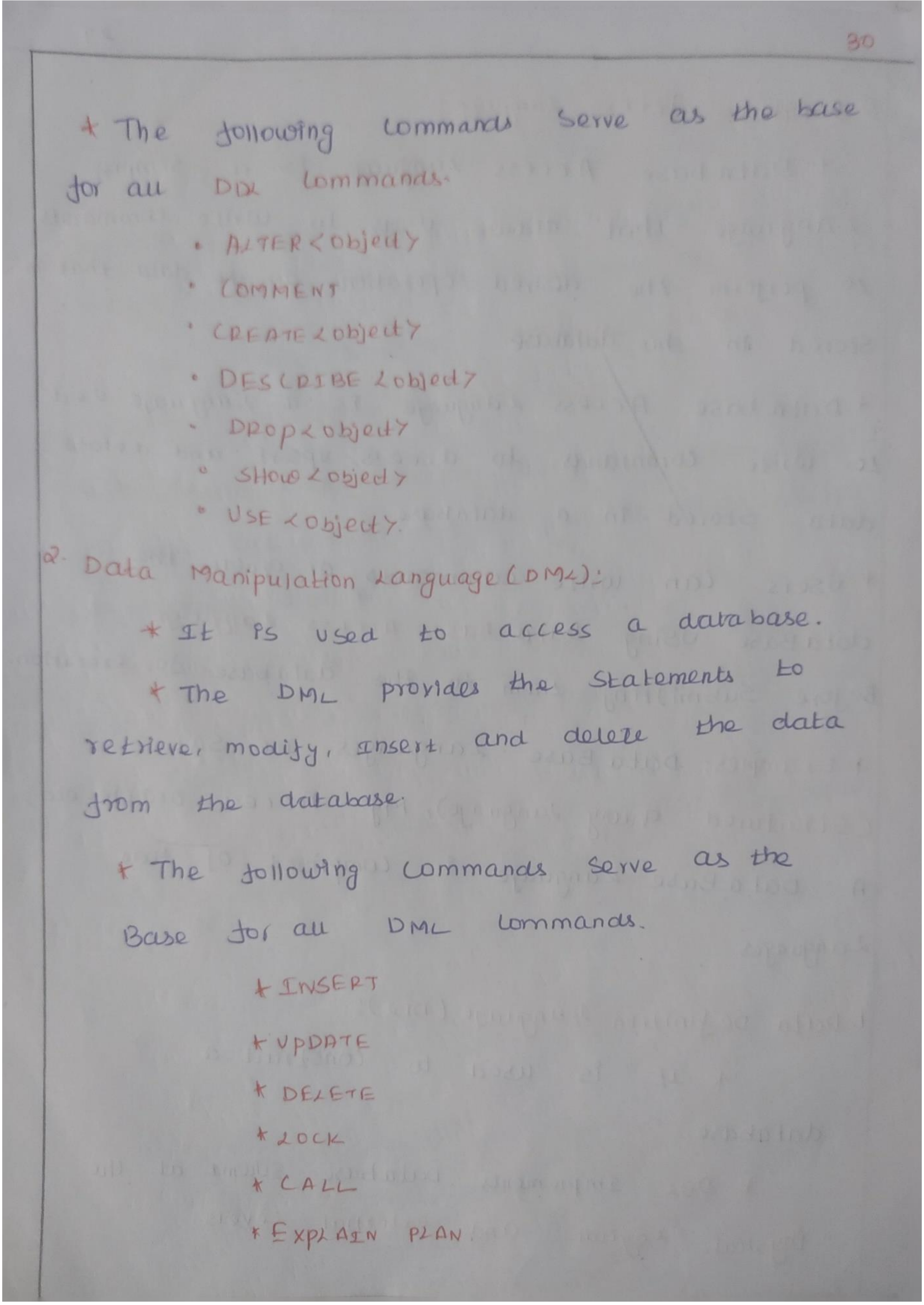
* Examples; Data Base Languages are SQL (Structured query Language), My Access, oracle, etc.,

A DataBase Language is comprised of two Languages.

1. Data Definition Language (DDL):

* It is used to construct a database.

* DDL Implements Database schema at the Physical, Logical and external Levels.



* The following commands serve as the base for all DDL commands.

- ALTER<object>
- COMMENT
- CREATE<object>
- DESCRIBE <object>
- DROP<object>
- SHOW <object>
- USE <object>.

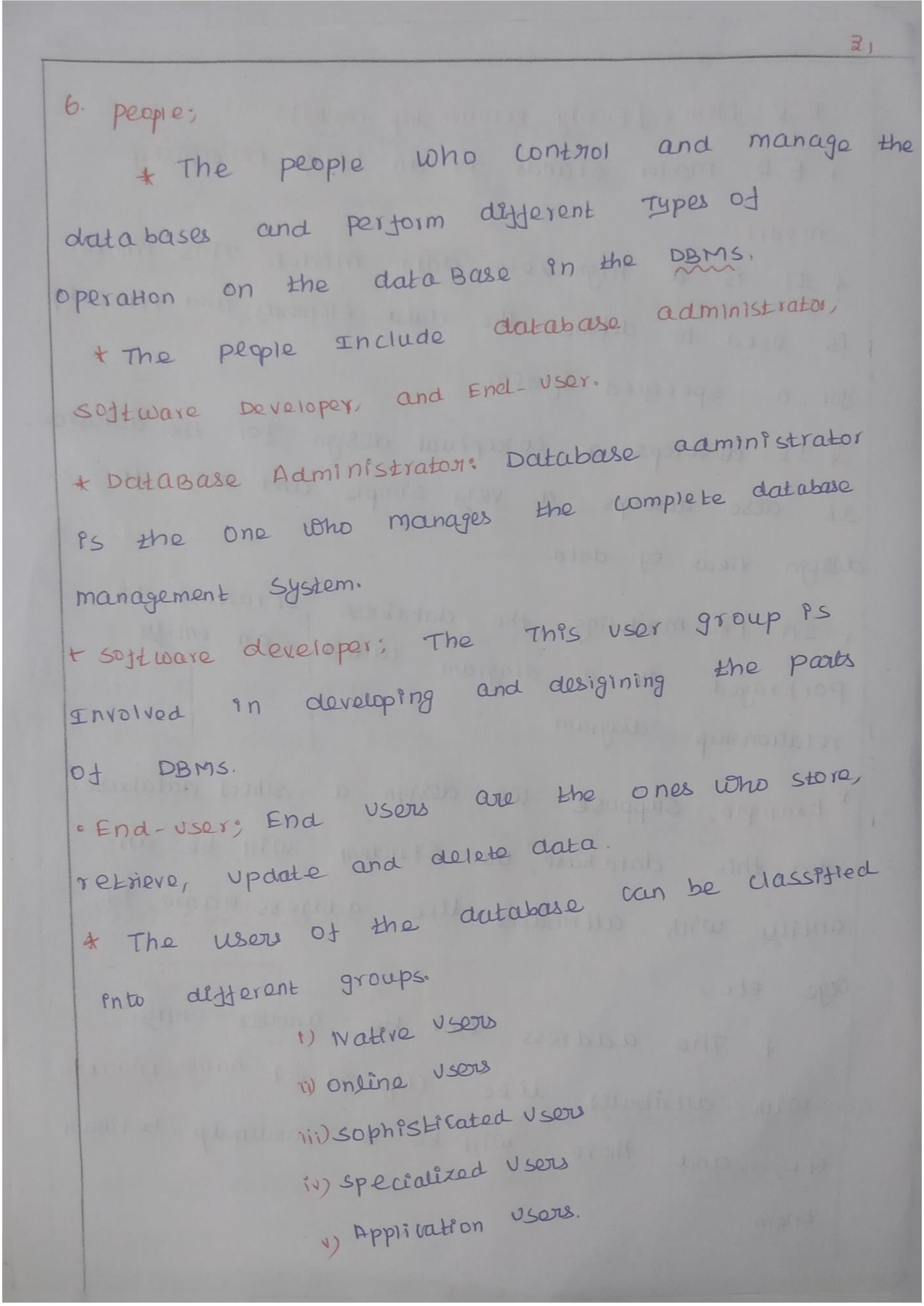
2. Data Manipulation Language (DML):

* It is used to access a database.

* The DML provides the statements to retrieve, modify, insert and delete the data from the database.

* The following commands serve as the Base for all DML commands.

* INSERT
* UPDATE
* DELETE
* LOCK
* CALL
* EXPLAIN PLAN.



31

6. People;

* The people who control and manage the databases and perform different Types of operation on the data Base in the DBMS.

* The people Include database administrator, software Developer, and End-user.

* DataBase Administrator: Database administrator is the One who manages the complete database management System.

* software developer; The This user group is the parts Involved in developing and desigining of DBMS.

* End-user; End users are the ones who store, retrieve, update and delete data.

* The users of the database can be classified into different groups.

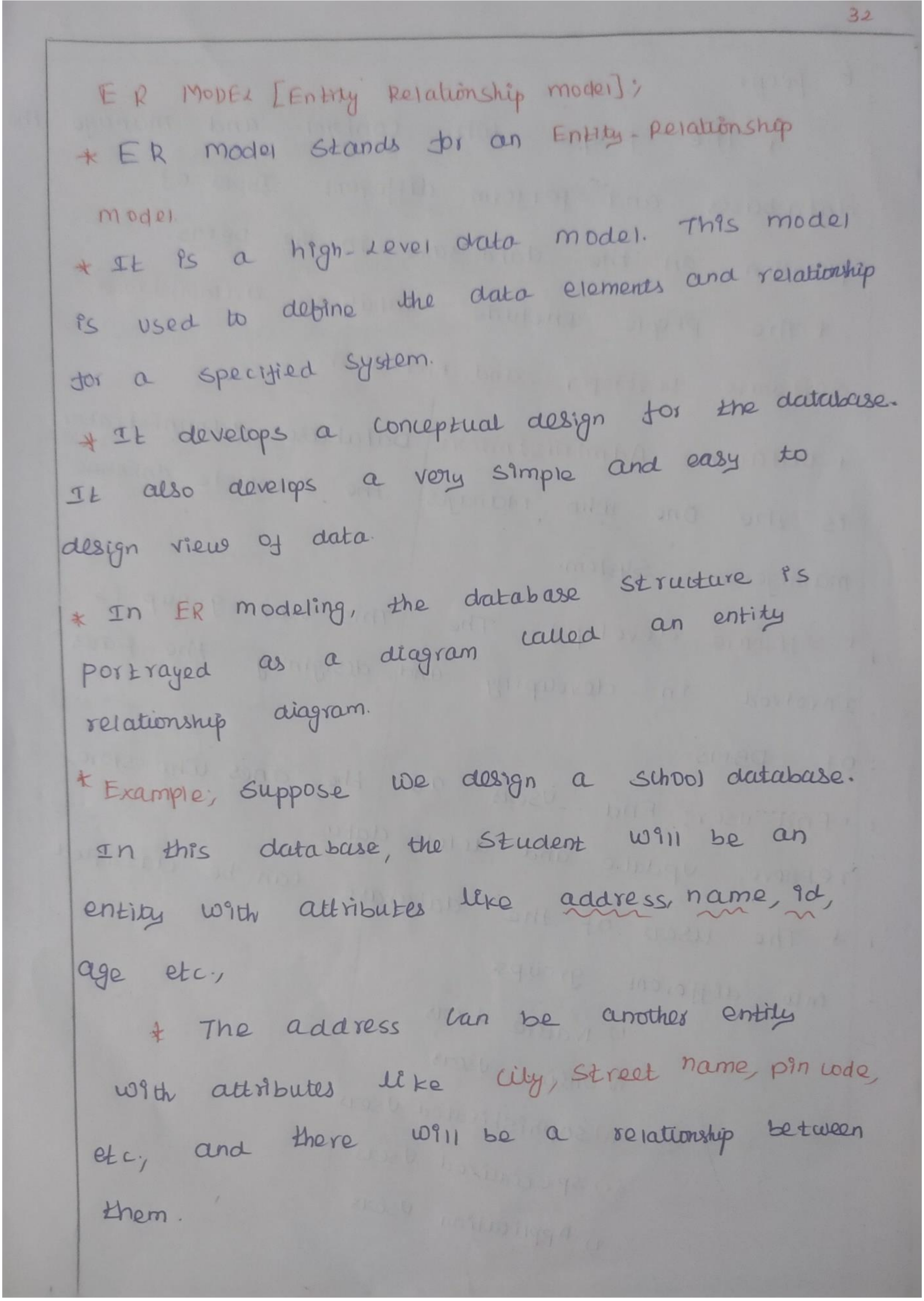
i) Native users

ii) online users

iii) sophisticated users

iv) specialized users

v) Application users.



E R MODEL [Entity Relationship model];

* E R model Stands for an Entity - Relationship model

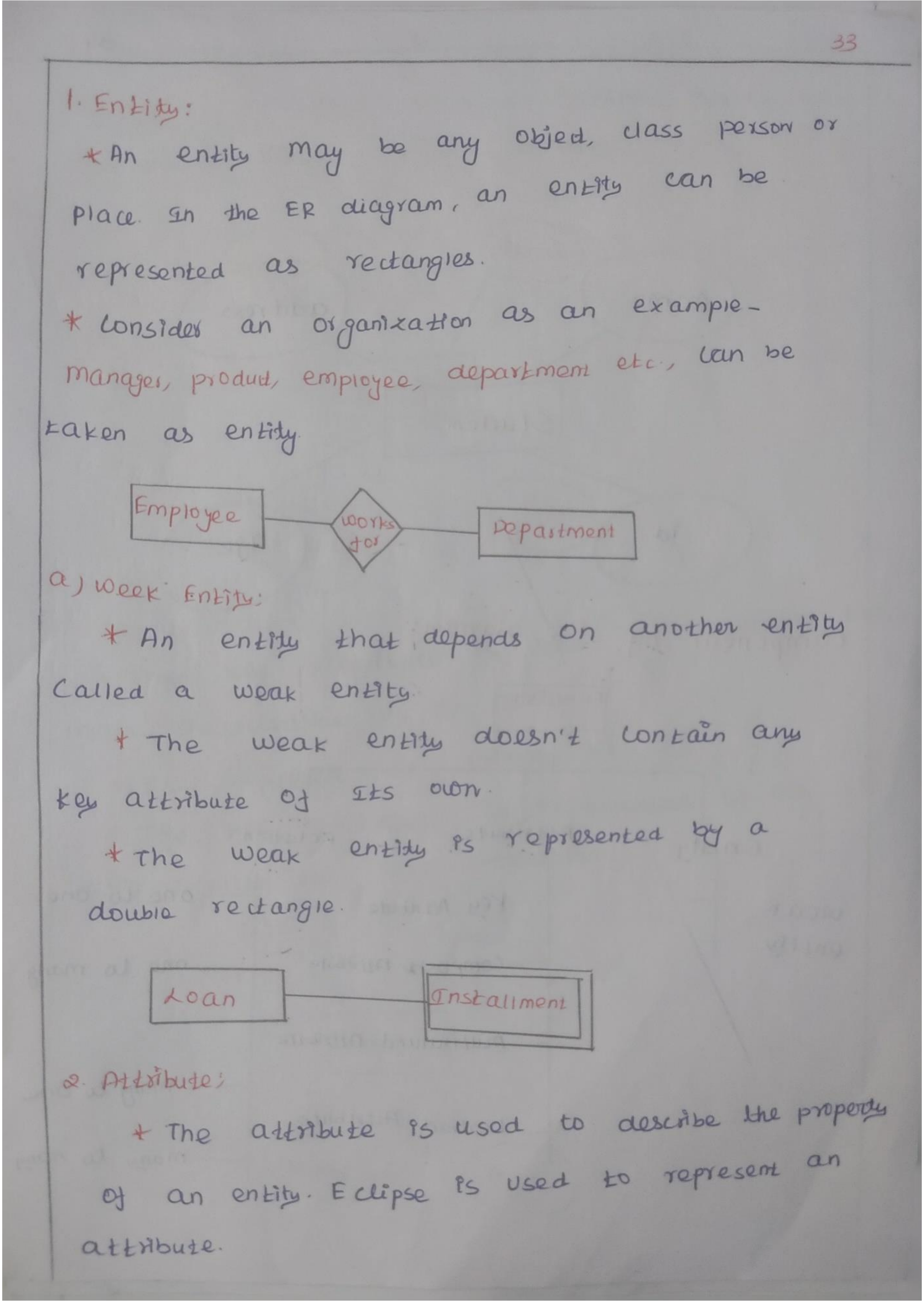
* It is a high-level data model. This model is used to define the data elements and relationship for a specified system.

* It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.

* In ER modeling, the database structure is portrayed as a diagram called an entity relationship diagram.

* Example; Suppose we design a school database. In this database, the Student will be an entity with attributes like address, name, id, age etc.,

* The address can be another entity with attributes like city, street name, pin code, etc., and there will be a relationship between them.



33

1. Entity:

* An entity may be any object, class person or place in the ER diagram, an entity can be represented as rectangles.

* Consider an organization as an example- manager, product, employee, department etc., can be taken as entity.

Employee
works for
Department

a) Week Entity:

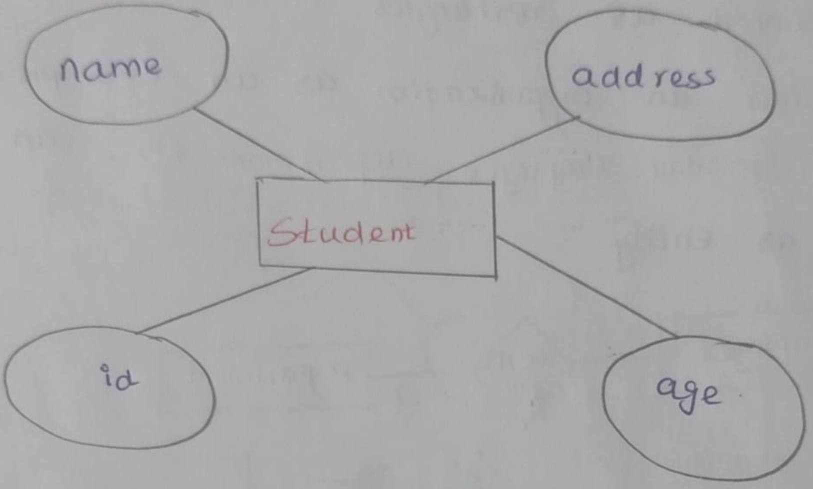
* An entity that depends on another entity called a weak entity.

* The weak entity doesn't contain any key attribute of its own.

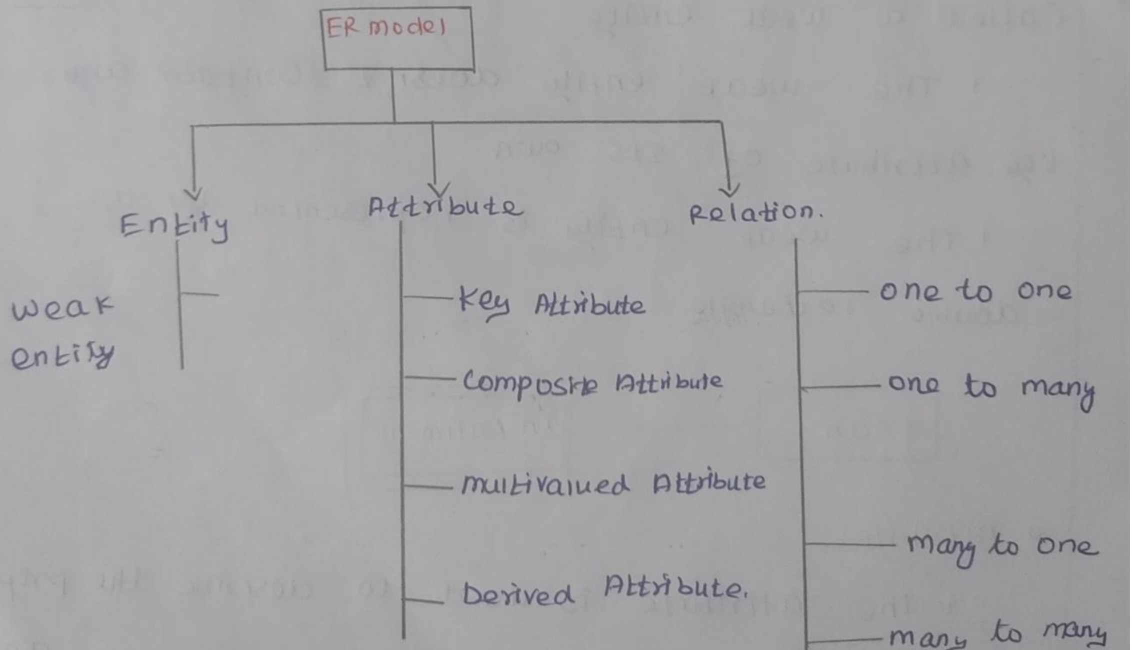
* The weak entity is represented by a double rectangle.

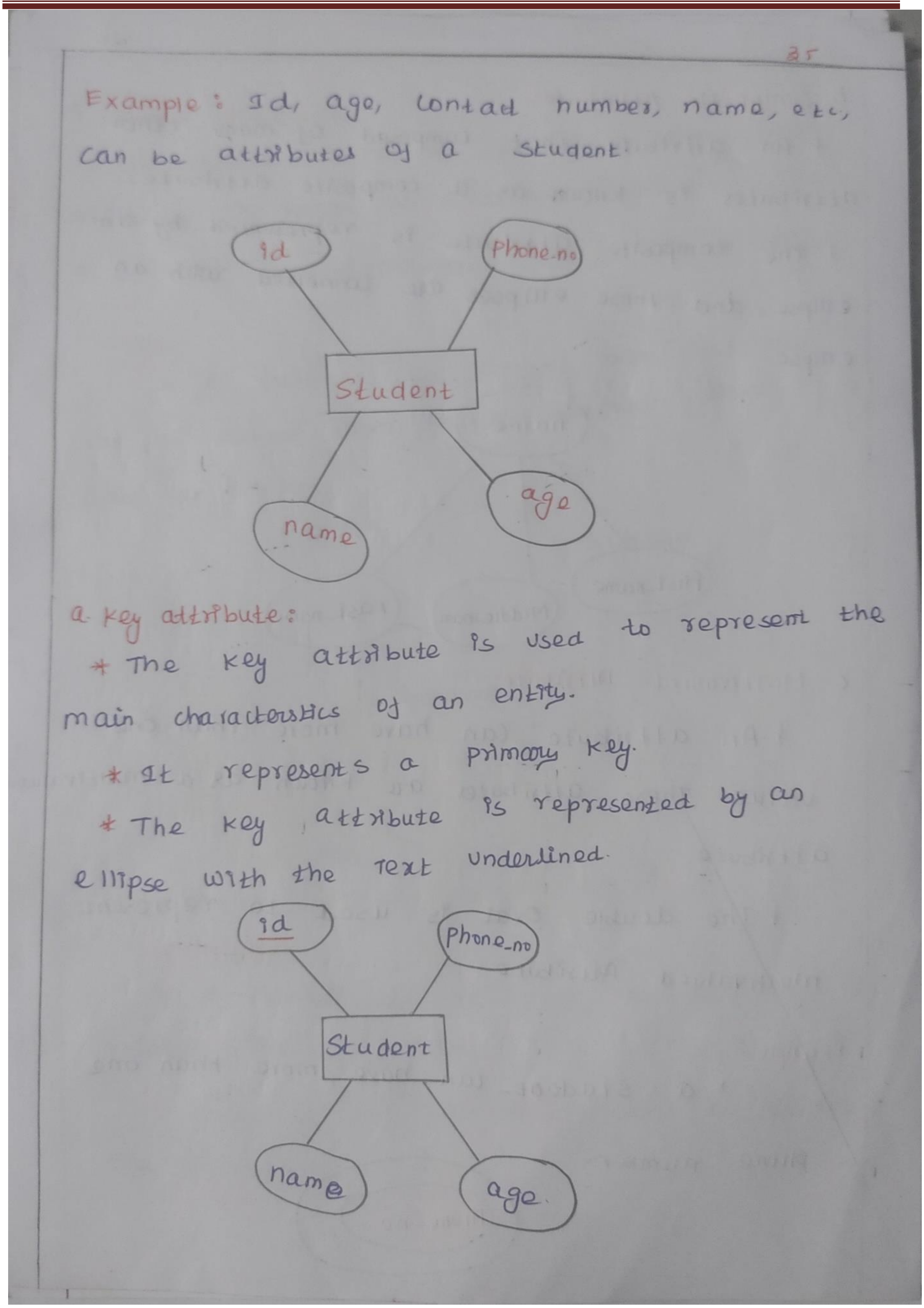
2. Attribute:

* The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.



Component of ER Diagram:





Example: Id, age, contact number, name, etc, can be attributes of a student.

id
Phone.no
Student
name
age

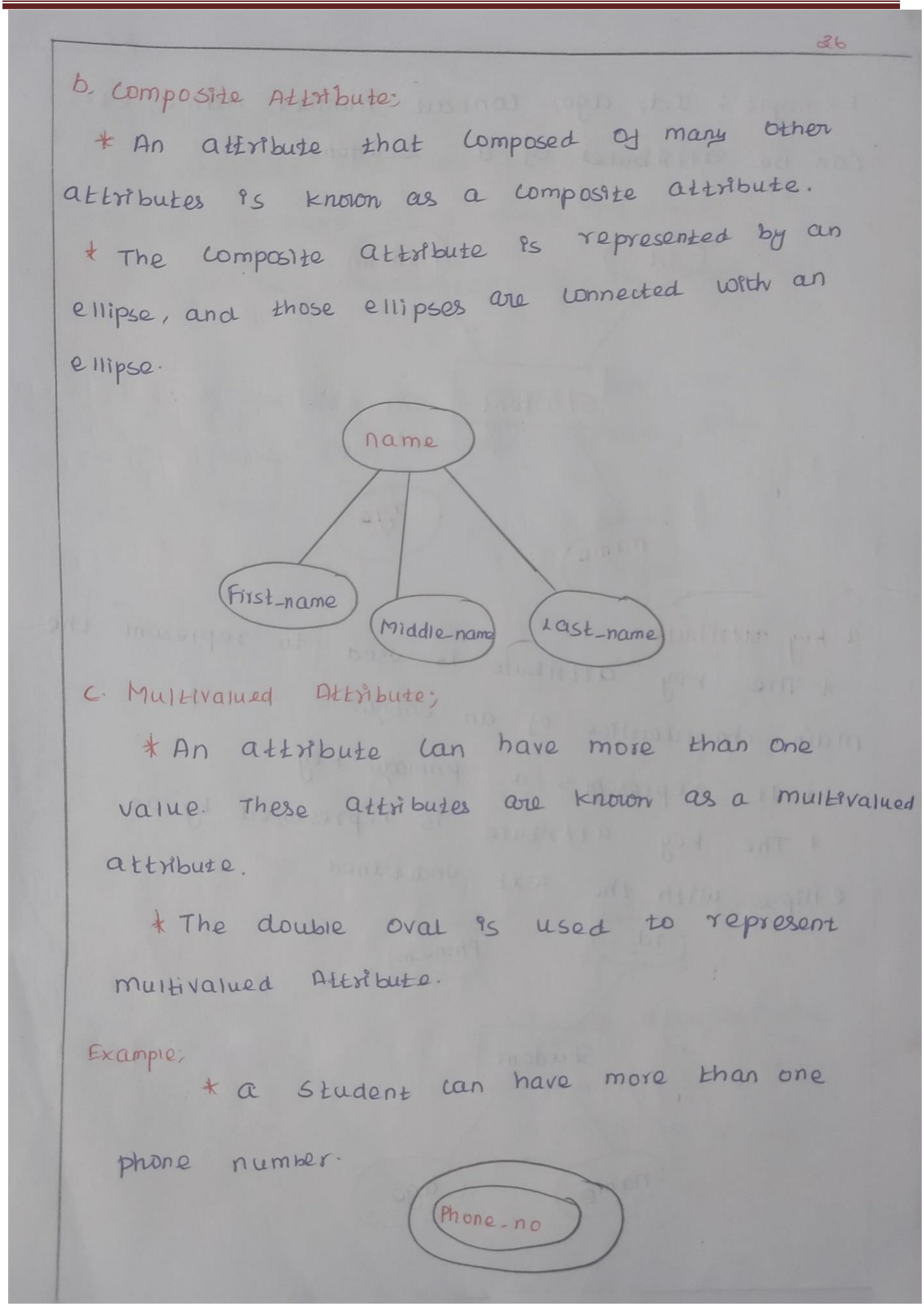
a. key attribute:

* The key attribute is used to represent the main characterstics of an entity.

* It represents a primary key.

* The key attribute is represented by an ellipse with the text underlined.

id
Phone_no
Student
name
age



b. Composite Attribute;

* An attribute that composed of many other attributes is known as a composite attribute.

* The composite attribute is represented by an ellipse, and those ellipses are connected with an ellipse.

name
First_name
Middle_name
Last_name

c. Multivalued Attribute;

* An attribute can have more than one value. These attributes are known as a multivalued attribute.

* The double oval is used to represent multivalued Attribute.

Example;

* a Student can have more than one phone number.

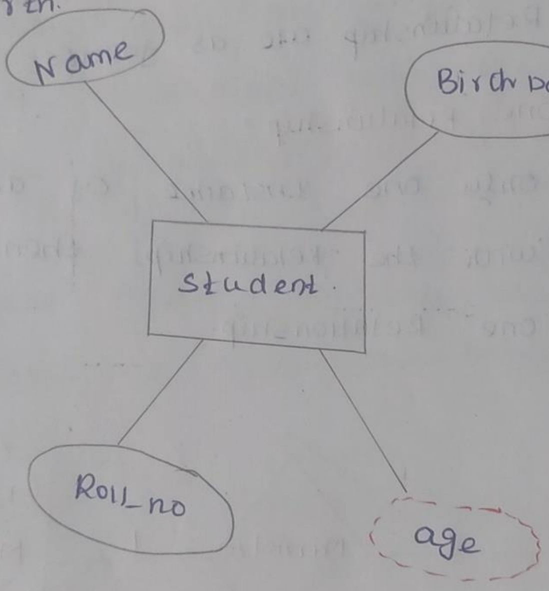
d. derived Attribute:

* An attribute that can be derived from other attribute is known as a derived attribute.

* It can be Represented by a dashed ellipse.

Example:

* A person's age changes over time and can be derived from another attribute like Date of birth.

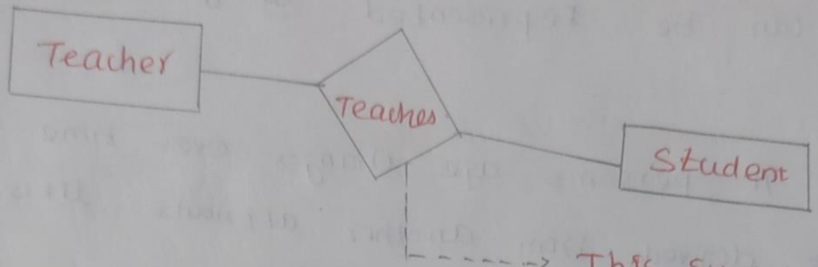


3. Relationship:

* A Relationship is used to describe the relation between entities.

* Diamond or rhombus is used to represent the Relationship.

* A Relationship is used to describe the relation between entities.



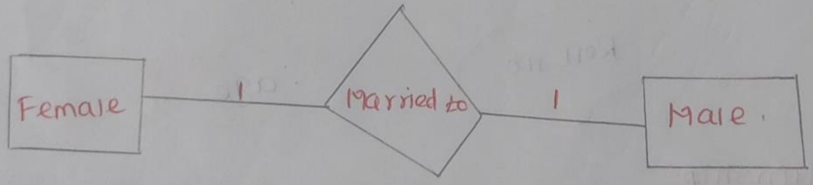
This Symbol Represent Relationship

Types of Relationship are as follows;

a) one-to-one Relationship:

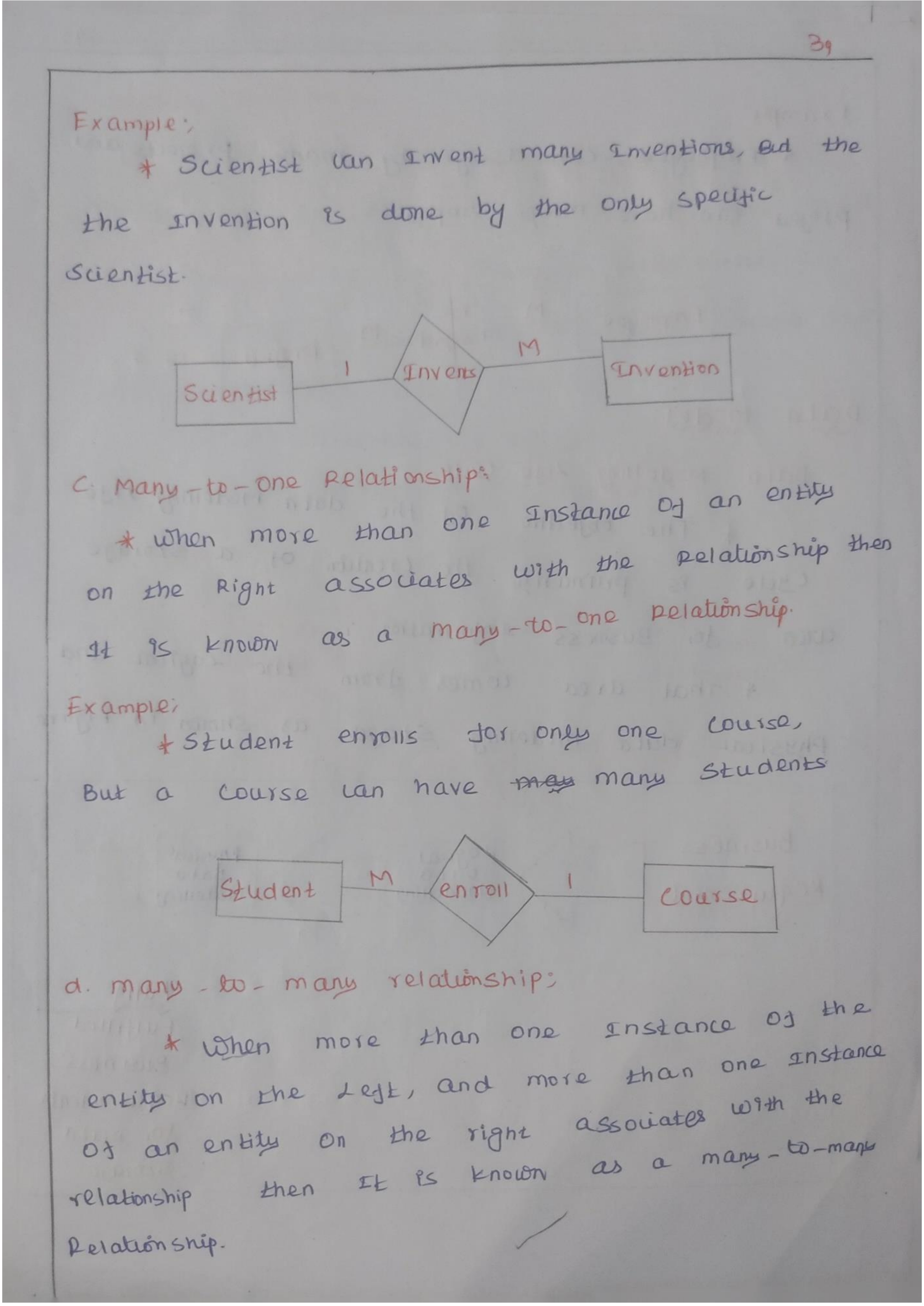
* When only one Instance of an entity is associated with the Relationship, then It is known as one to one Relationship.

Example;



b. one-to-many relationship:-

* When only one Instance of the entity on the left, and more than one Instance of an entity on the Right associates with the Relationship then this is known as a One-to many Relationship.



Example;

* Scientist can Invent many Inventions, But the the invention is done by the only specific Scientist.

Scientist
1
Invents
M
Invention

c. Many-to-one Relationship:

* When more than one Instance of an entity on the Right associates with the Relationship then It is known as a many-to-one Relationship.

Example;

* Student enrolls for only one course, But a course can have ~~may~~ many students

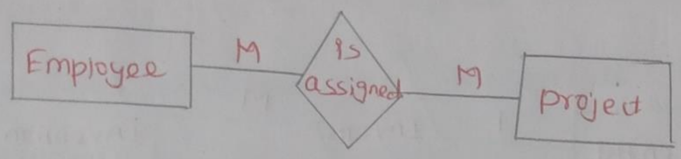
Student
M
enroll
1
Course

d. many-to-many relationship;

* When more than one Instance of the entity on the Left, and more than one Instance of an entity on the right associates with the relationship then It is known as a many-to-many Relationship.

Example;

* Employee can assign by many Projects and Project can have many employees.

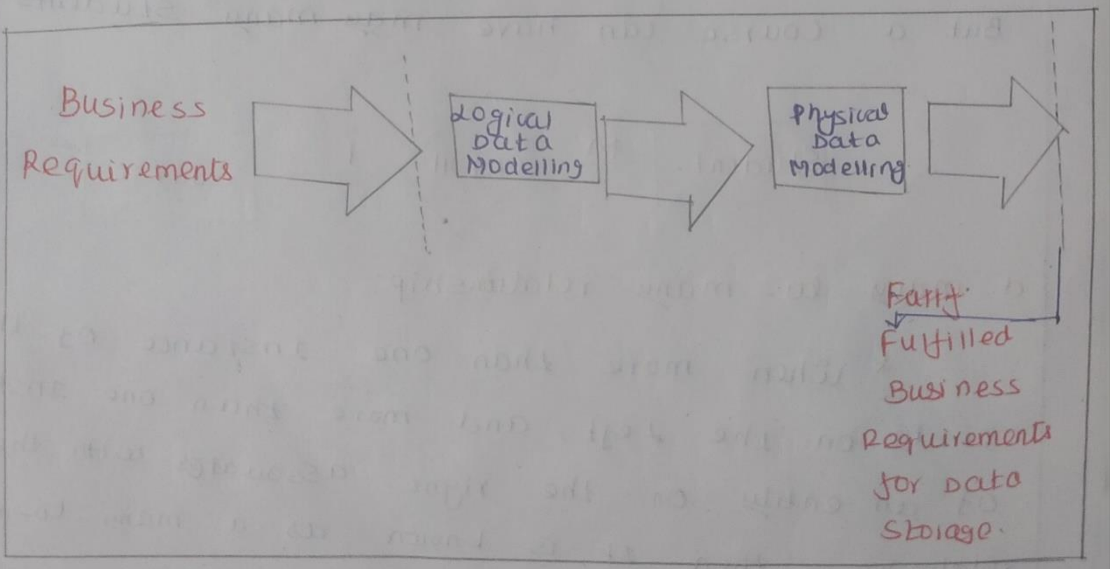


Data Model;

Data Modelling Life Cycle;

* The objective of the data modelling life cycle is primarily the creation of a storage area for Business Information.

* That area comes from the logical and Physical data modelling stages, as shown in figure.



Conceptual Data Model;

* A conceptual data model recognizes the highest level Relationships between the different entities.

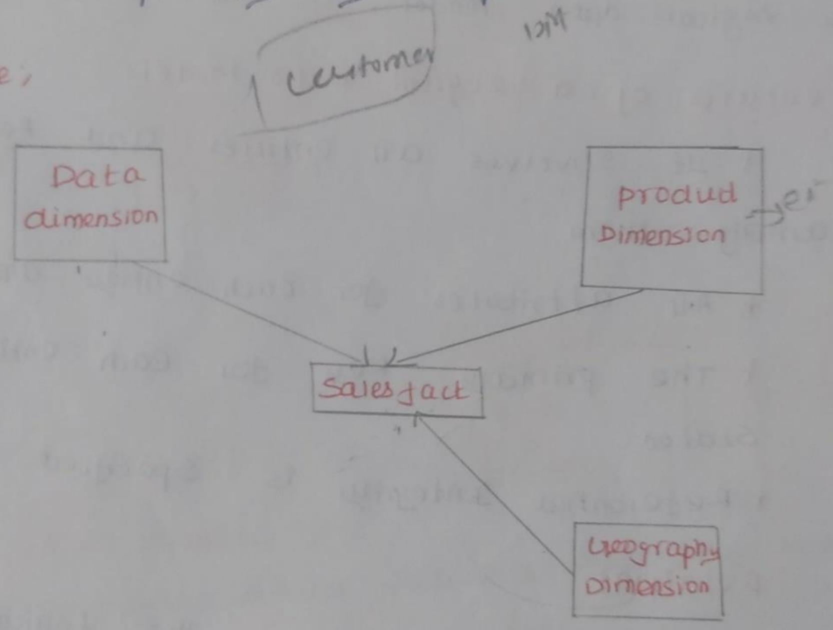
* characteristics of the conceptual data model.

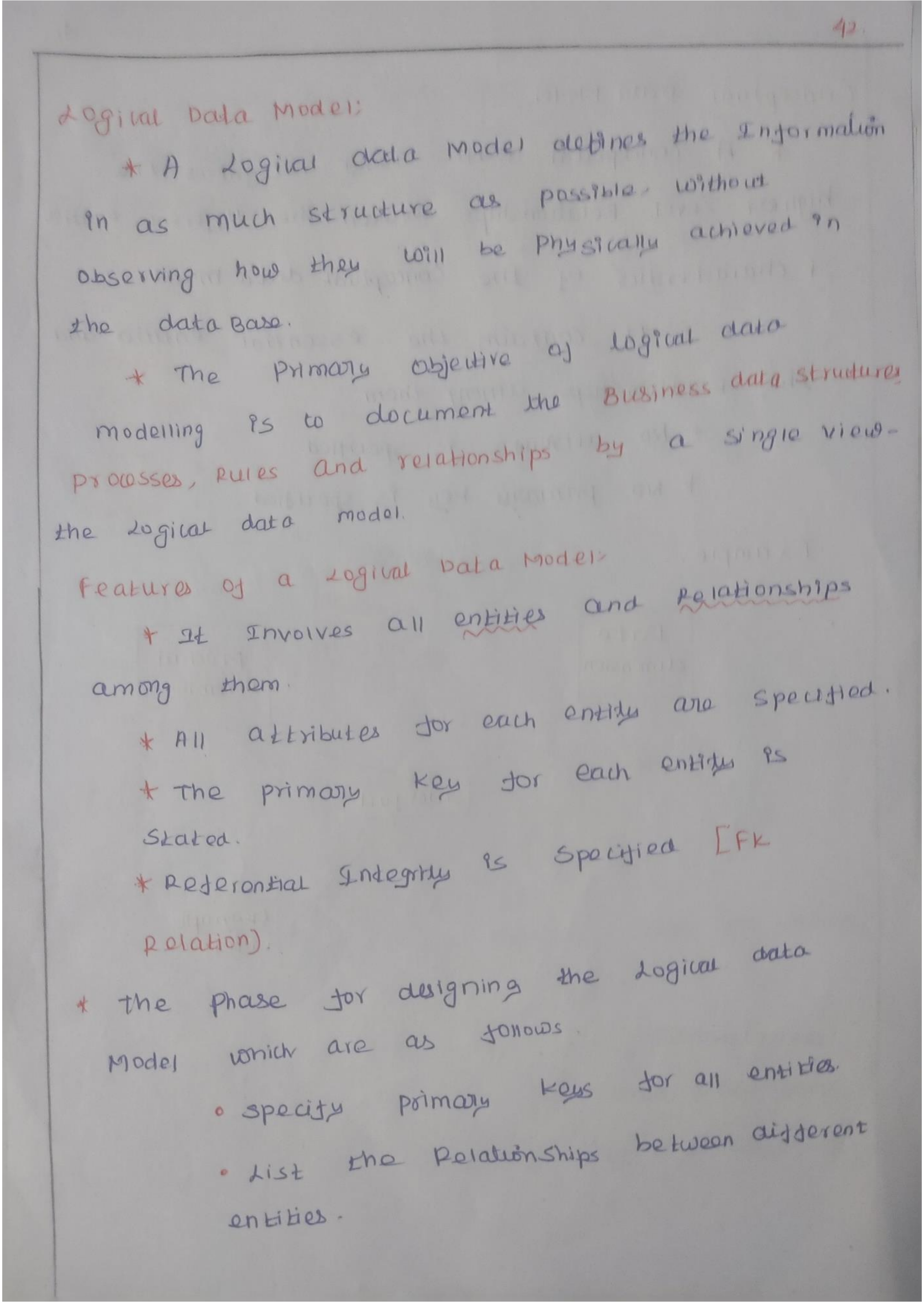
  • It contains the essential entities and the Relationships among them.

  • No attribute is specified.

  * No primary key is specified.

Example;





Logical Data Model:

* A Logical data Model defines the Information in as much structure as possible, without observing how they will be Physically achieved in the data Base.

* The Primary objective of logical data modelling is to document the Business data structures processes, Rules and relationships by a single view - the Logical data model.

Features of a Logical Data Model:

* It Involves all entities and Relationships among them.

* All attributes for each entity are specified.

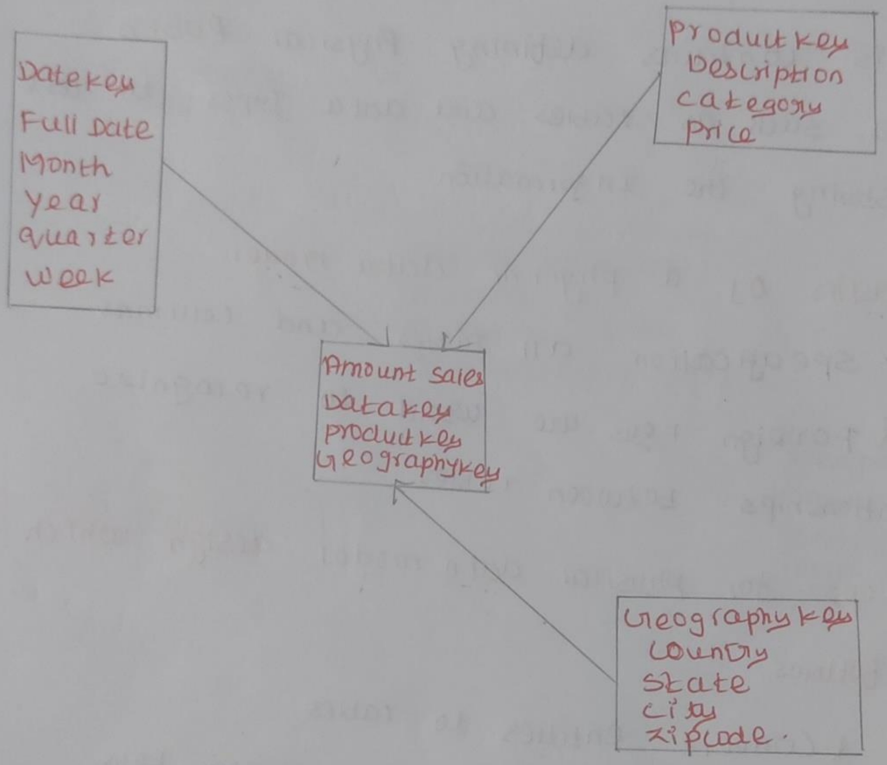
* The primary key for each entity is stated.

* Referential Integrity is specified [FK Relation).

* The Phase for designing the Logical data Model which are as follows.

- specify primary keys for all entities.
- List the Relationships between different entities.

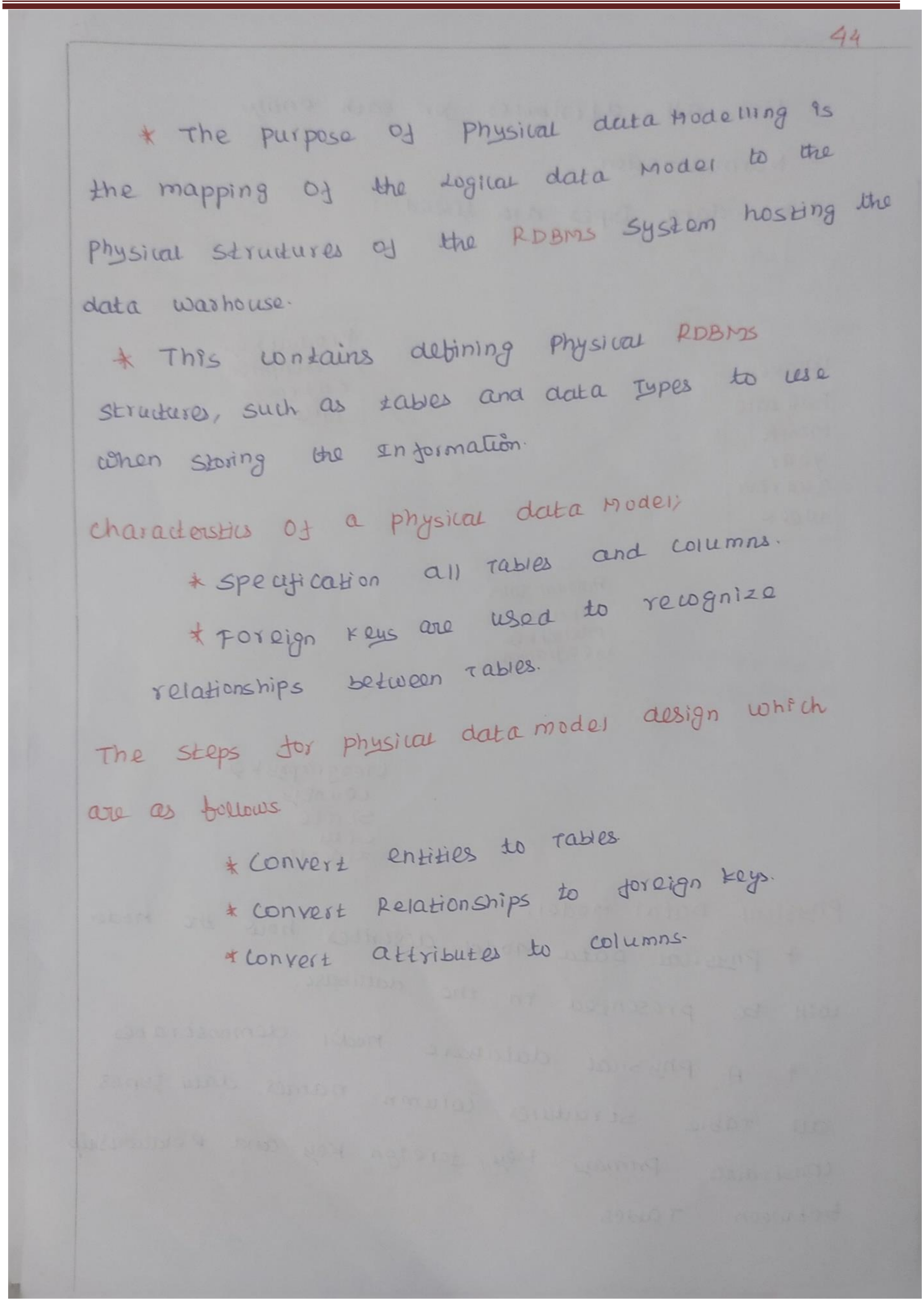
- List all attributes for each entity.
- Normalization
- No data Types are listed.



Physical Datal Model;

* Physical Data Model describes how the model will be presented in the database.

* A Physical database model demonstrates all Table strutures, column names, data Types, constraints, Primary key, foreign key and Relationship between Tables.



* The purpose of Physical data Modelling is the mapping of the logical data Model to the Physical structures of the RDBMS system hosting the data warhouse.

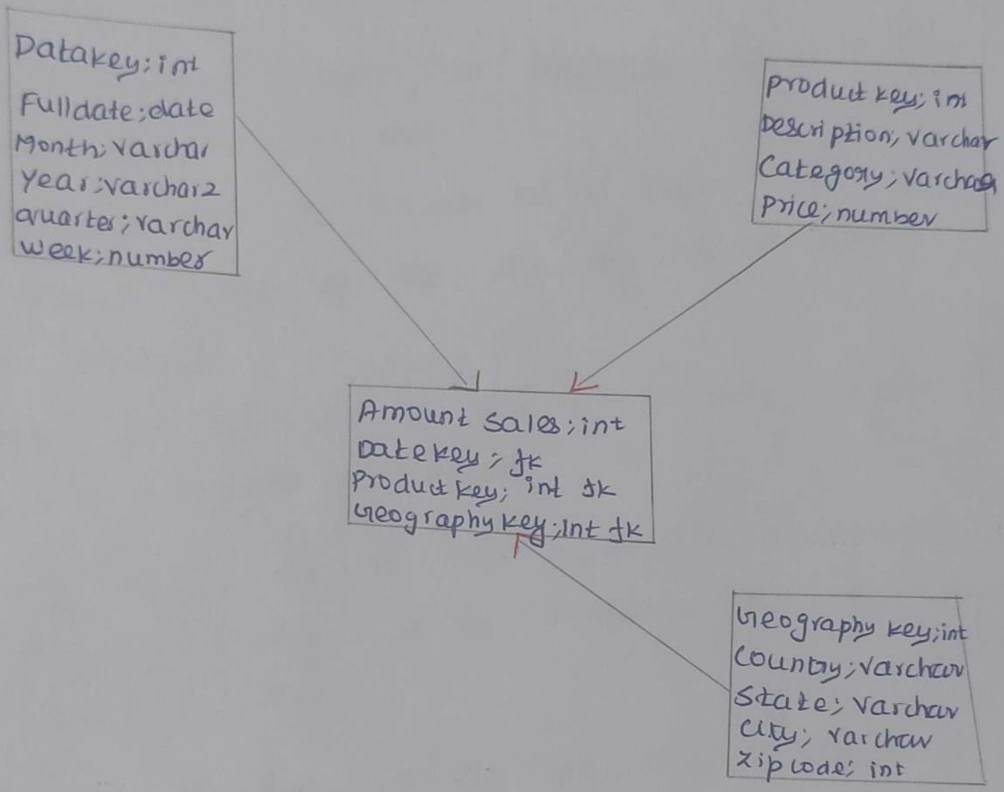
* This contains defining physical RDBMS structures, such as tables and data Types to use when storing the information.

Characterstics of a physical data Model;

* Specification all Tables and columns.
* Foreign Keys are used to recognize relationships between tables.

The steps for physical data model design which are as follows

* Convert entities to Tables.
* Convert Relationships to foreign keys.
* Convert attributes to columns.


Datakey;int
Fulldate;date
Month;varchar
Year;varchar2
quarter;varchar
week;number
product key;int
Description;varchar
Category;varchar
Price;number
Amount sales;int
Datekey;fk
Product key; int fk
Geography key;int fk
Geography key;int
Country;varchar
State;varchar
city;varchar
zip code;int