

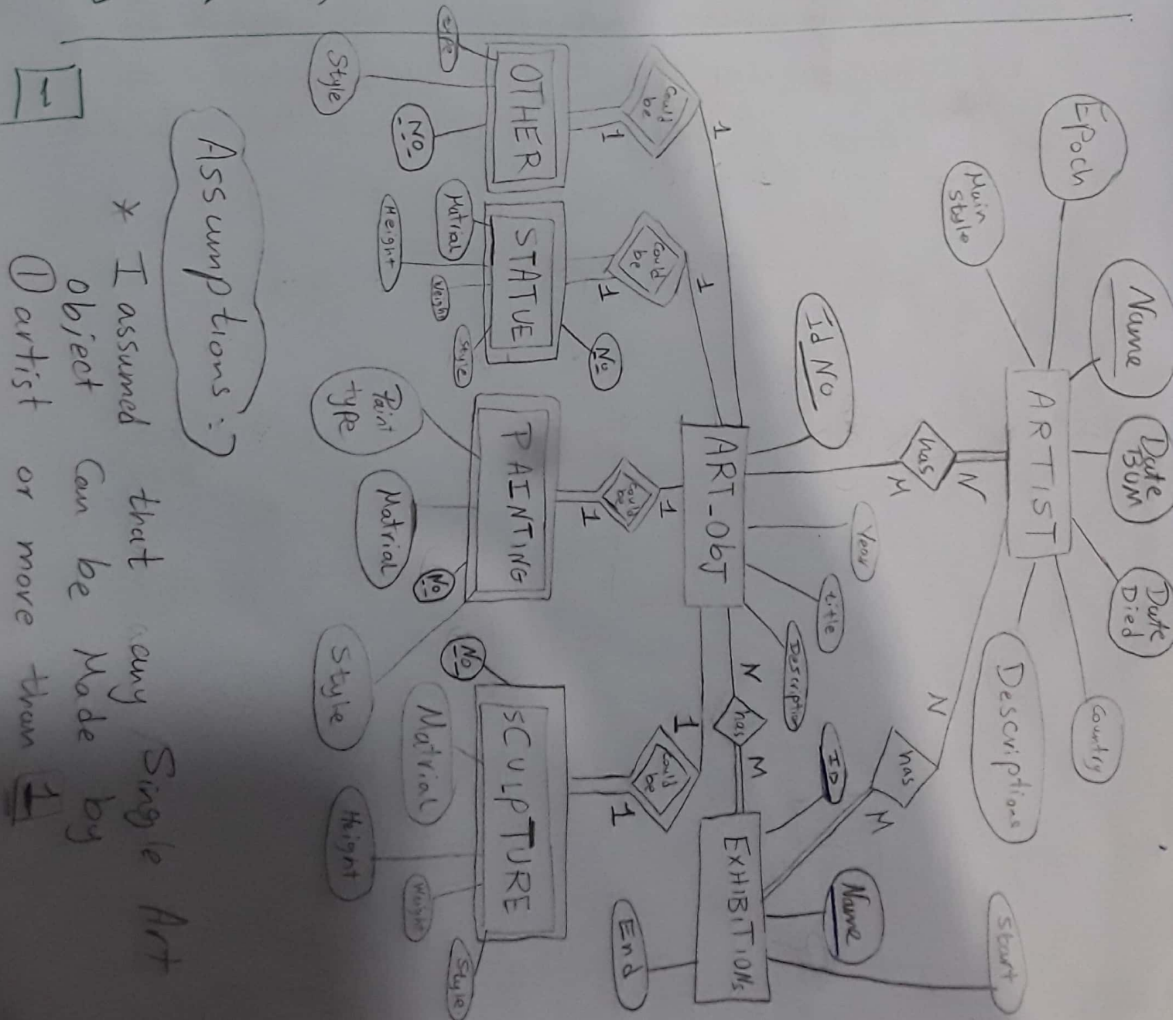
# Data Base Management System

(1)

Name: Asmaa Gamal Abdel-Halem Mabrouk Nagy  
Department: Communications & Electronics  
Course type: Audience Course

## Sheet (2)

### ER Model, Relational Model, Relational Design





② User requirements:

1) Bank: Code, Name, Add

2) each Bank has Branches:  
add, No.

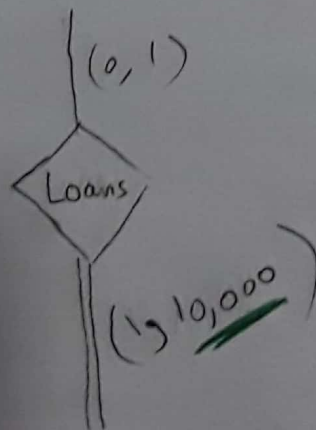
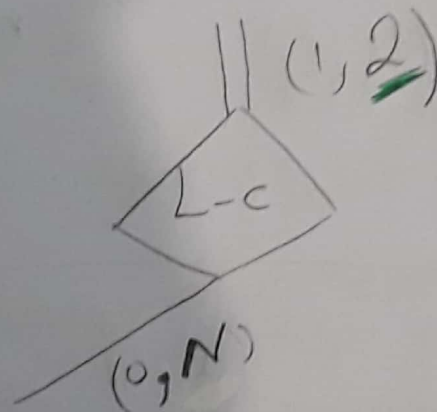
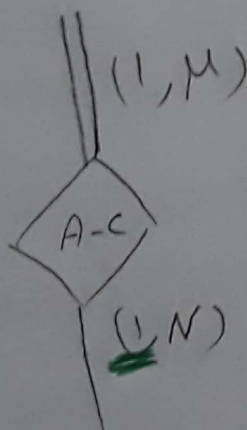
3) each Branch has:

3' Account: type, Balance, No.

3'' LOAN: Amount, type, No.

4) Customer: Name, Add, SSN as Unique  
Key, Phone, Accounts, LOANS

③

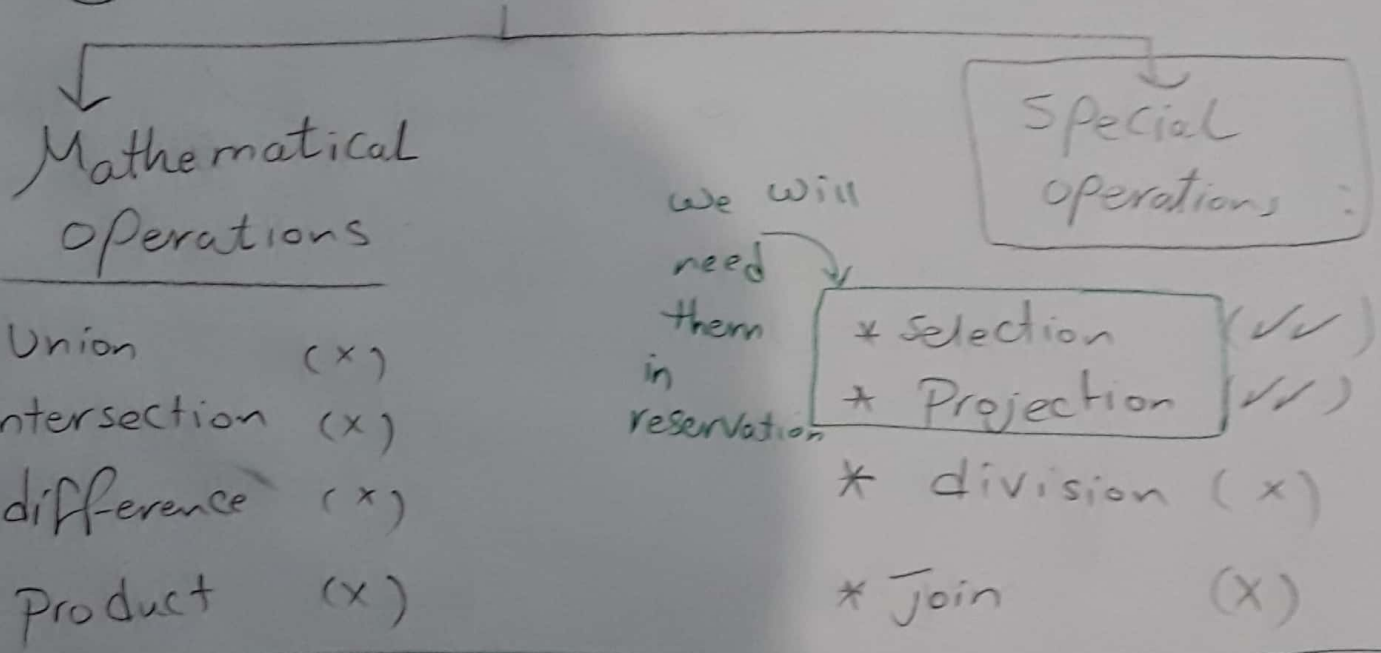




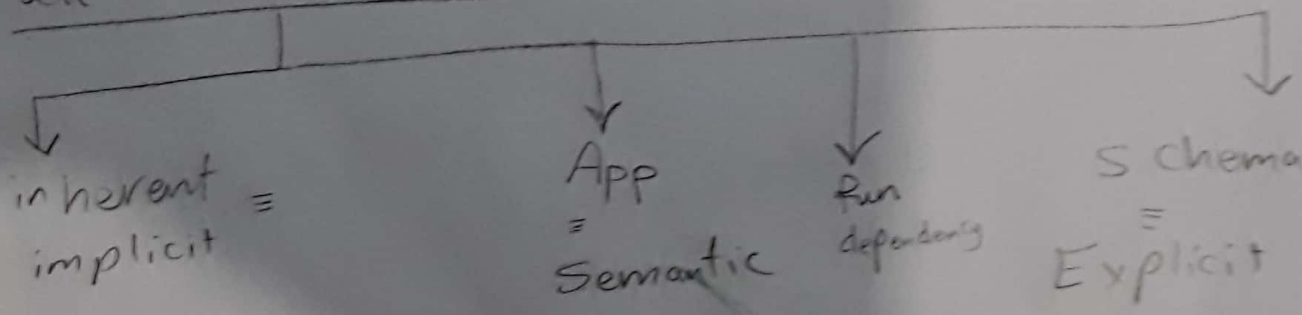
### 3 Foreign Keys

- ① EnROLL (SN, Course #, Quarter)  
Student (SN) →
- ② Book-ADopTION (Course #, Quarter, Book-ISBN)
- ③ TExT (Book-ISBN, ...)

### 4 a \* relational algebra operations



### b in update operation we will check all Constraints :



- Check :
- ① Domain
  - ② Key and Null
  - ③ Entity integrity
  - ④ Referential "key"
  - ⑤ Foreign key

}

Constraints

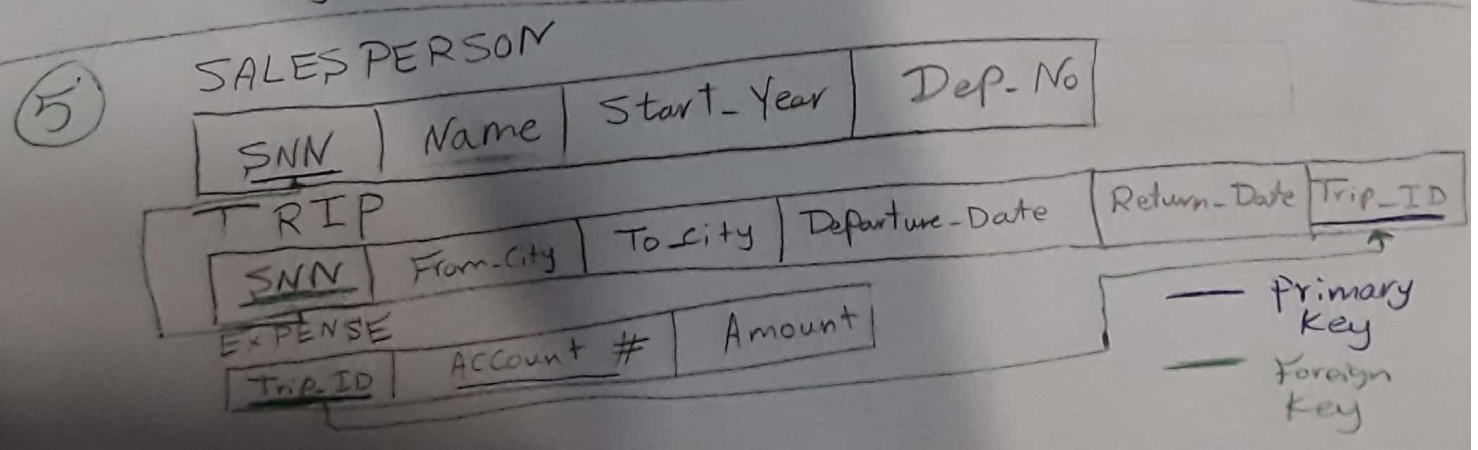
③ \* Entity integrity : No Primary Key Value can be null (in all Database)  
Ex: "Air-Code" in AIRPORT

\* Key: Superkey [ unique ] and removing any attribute (A) from (K) leave a set of attributes (K) that is not a superkey of R = any more. Ex: Flight-num + Fare-Code = Key and if you deleted the Fare-Code this will NOT be a Superkey any more.

\* Referential Integrity:

Ex: "Flight-num" in "FARE" should maintain consistency because it refer to "flight-num" in the "FLIGHT" relation/table.

④ Ex: \* "Leg-num" in "LEG-Instance" is related to "Leg-num" in "FLIGHT-LEG".

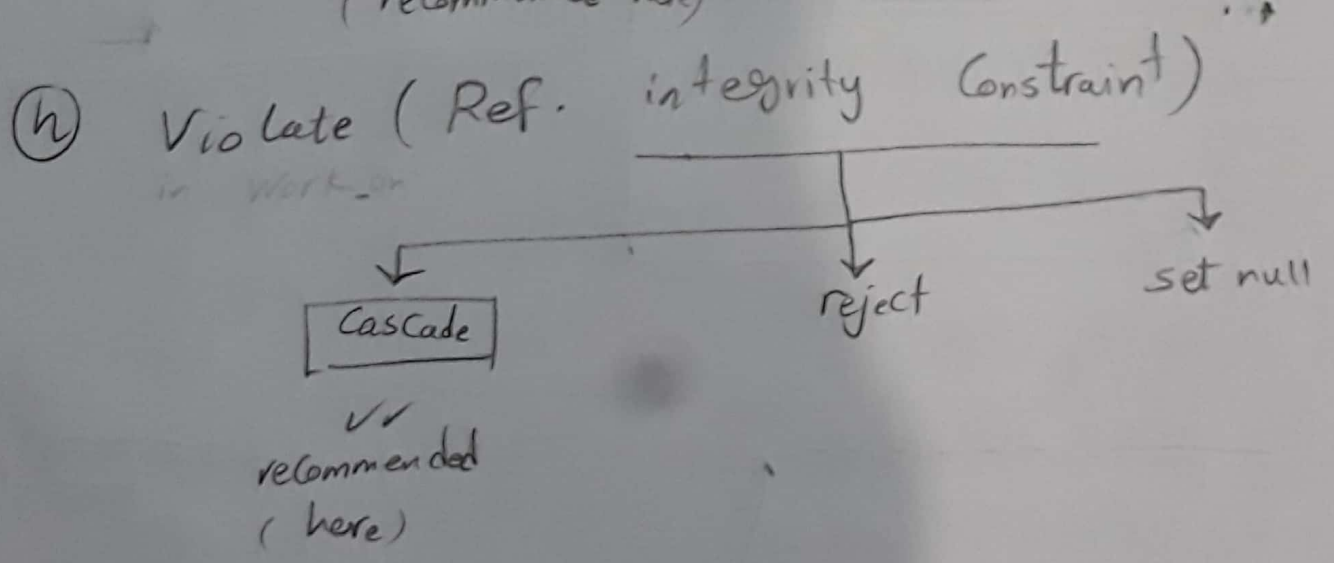
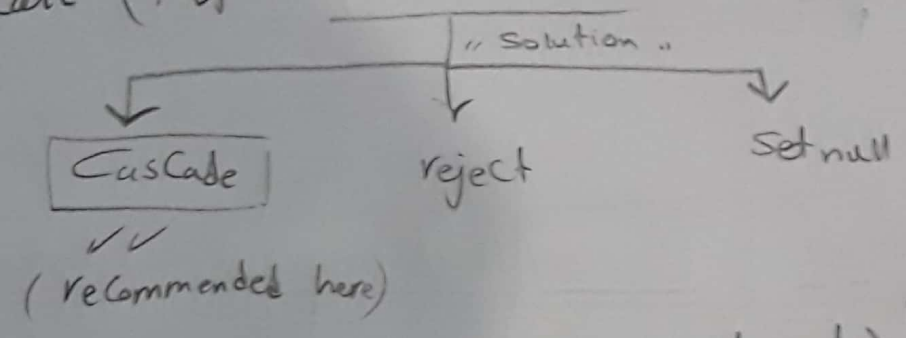


⑥ \*I think all of the deletion of the referenced keys in a referential integrity Constraints should be Cascaded

Except: deleting the "warehouse#" in the "warehouse" relation it should be setted to Null/default value in the "shipment" relation  
Warehouse #

\* I think we should cascade any update of the Primary key and if it violate the Domain Constraint or any other similar Constraints then we should reject it

- Z
- (a) no violation
  - (b) Violate Dnum not exist → reject
  - (c) Violate key constraint → reject
  - (d) Violate, reference Pno → reject  
Can't be null (Ref. integrity constraint)
  - (e) no violation
  - (f) no violation
  - (g) Violate (Ref integrity constraint)



- (i) no violation
- (j) Violate → Reject  
(because this primary key is not exist  
so it can't be used as the new foreign key)
- (k) no violation



Note: \* Port Visit is a weak entity, So, it should have the "Sname" of \* SHIP.

— Primary Key  
— Foreign Key

SHIP MOVEMENT

Latitude	Longitude	Time	Date	<u>No</u>	Sname
----------	-----------	------	------	-----------	-------

SHIP

<u>Sname</u>	Owner	<u>ID</u>	P name	S-Type
--------------	-------	-----------	--------	--------

SHIP-TYPE

<u>Type</u>	Ton nage	Hull
-------------	----------	------

PORT

<u>P name</u>	P_num	State/country Name	Sea/Ocean/Lake Name
---------------	-------	--------------------	---------------------

STATE/COUNTRY

<u>Name</u>	Continent	No
-------------	-----------	----

PORT\_VISIT

<u>Sname</u>	<u>Start-date</u>	End-date	No.	P name
--------------	-------------------	----------	-----	--------

SEA/OCEAN/LAKE

<u>Name</u>
-------------

SHIP-AT-PORT

<u>Sname</u>	Start.date	Pname	State/country name
--------------	------------	-------	--------------------

\*