20/02

DATE

PAGE NO

* Database lang → SQL (technology)
structural query Lang.
operation perform – (CRUD) → Delete
Create Read Update

SQL is technology
of Database testing, ETL- BI testing → Implimentation.

* Database testing
→ In Database testing, we are going to check impact of Backend due to frontend operations of users, and operations are - CRUD.
purpose → validation of data

+ Domain →
1. Transactional appln
– Run fundamental process

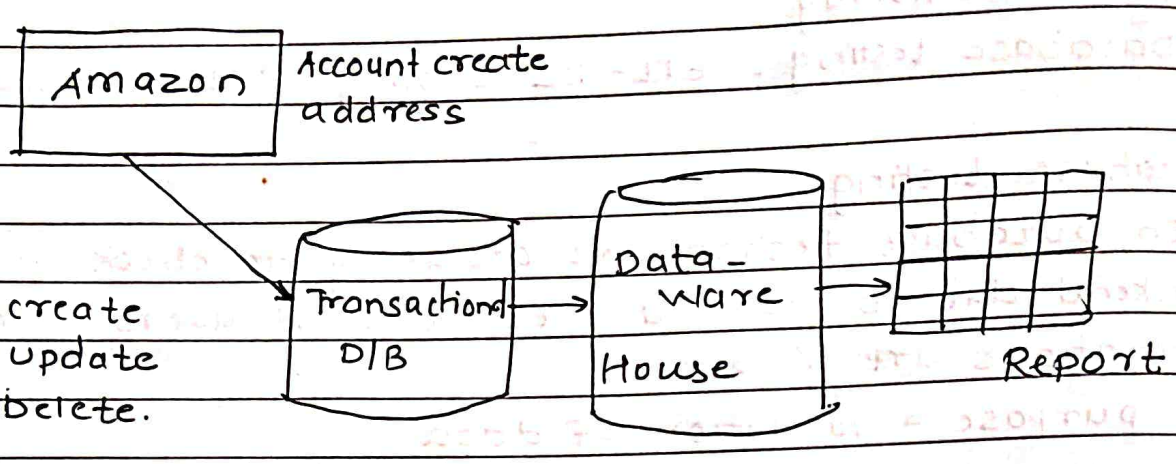
e.g Amazon - steps a) Amazon open
b) login
c) search product
d) Add to cart
e) Address
f) payment
g) Product deliver.

Gpay
- money transfer
- Recharge, bill.
} middle man.

what's app - data transfer

2) Analytical applications
→ Analysis
understand business, improve -

23/02



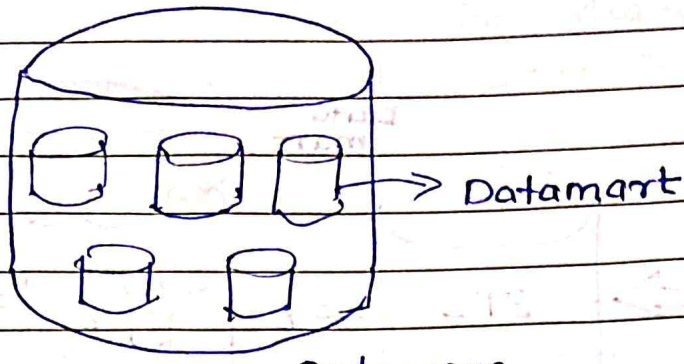
- Drowback of transactional system.
  1. Degrade performance
  2. time consumming because of multiple location
  3. Not fit for analysis purpose

24/02

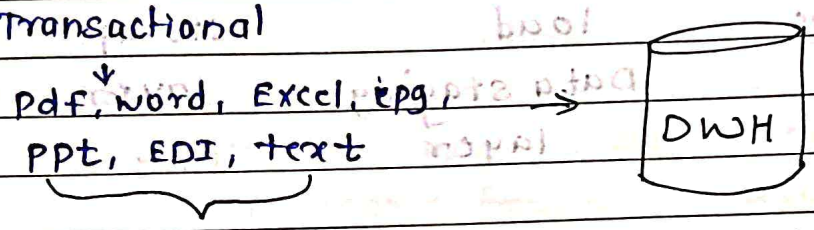
* Dataware house
  - It is a database that is designed for quering and analysis rather than transactional processing
  - It separates analysis workload from T.P
  - This help in
    i. Maintaining historical records
    ii. Analyzing the data to gain a better understand of the business and to improve the business.

Imp → Dataware house is a Subject-oriented, integrated, time-varying, non-volatile collection of data in support of the management's decision making process

- Subject oriented
  - subject wise data store



- Integrated
  - save in one format, from diff sources.



e.g. Qoader.

- Time-varying
  - Historical data is maintain.

e.g. Employee info., year wise

101	pooja	2010	Tralnee	T.P
101	Pooja	2010	Trainee	A.P.
101	Pooja	2011	Tester	
101	Pooja	2012	Manager	

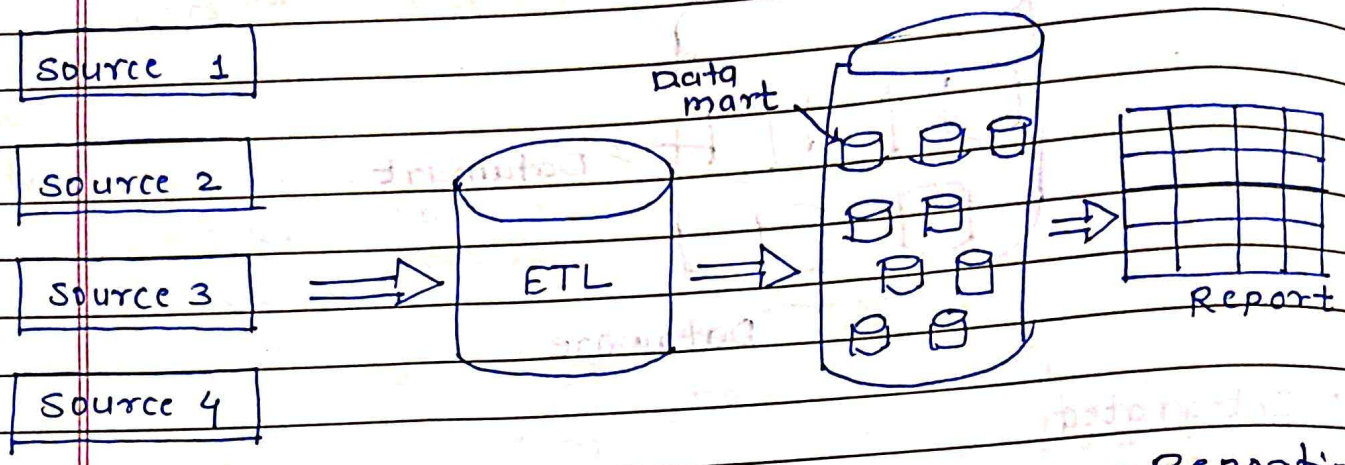
History

current

- Non-volatile → once datawlare is placed in dwH, it cant be changed. (~~No~~-Read, NO-Gorcate, NOupdate NO-delete)

25/02

* Architecture of DWH



Data source layer (transactional sys DB) OLTP

Extract transform load
Data staging layer

DWH.
Data storage layer
OLAP

Reporting layer

1. Data source layer
   → It is nothing but transactional system Database
   → stores various data, business data like sale, customer, product, purchase, delivery (in different format)

2. Data staging layer (Landing zone)
   - ETL
   Extract → Reading data from source database
   Transform → converting extracted data in Req^r form
   Load → Writing the data into target database

- It is process which defines how data is loaded from source system to target system. (Respective data mart)
- Transactional data → Analytical Data

→ minimise data loss, temp data processing during ETL

3. Data storage layer
- Data warehouse
- the place where the successfully cleaned, integrated, transformed & ordered data is stored in a multi-dimensional environment
- Now, data is available for quering & analysing.

Data mart - Subset of DWH
- subject wise data stored, single functional area

26/02

Meta data - it is data about data.
Meta data repository is used to stored meta data of data which is actually present in DWH.

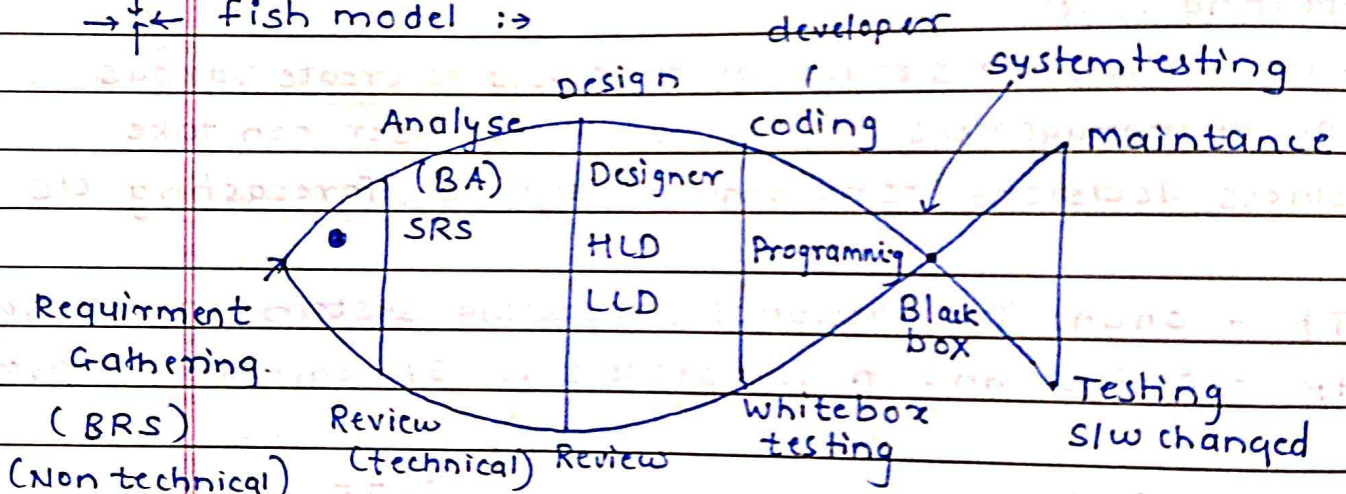
4. Reporting layer
- Data, in data storage layer is used to create various type of management reports from where user can take business decisions for planning, designing, forecasting etc

* OLTP - online Transactional processing system - source lay
OLAP - online Analytical processing system - storage lay

	OLTP	OLAP
• use	used for transaction processing	query processing
• Data	Holds current data stores all data small database. volitile data (CRUD)	current / History only relevent data Large database Non volitile (only Read)

• source	original source of Data.	Data comes from various OLTP sys.
• purpose	To control & run fundamental business task	To help with planning, Problem solving, decision support
• Queries	std & simple sql queries	complex sql queries.
• DB design	Highly normalized with many tables (3NF)	De-normalized with fewer table
• Users.	Many users	few users

→ fish model :→



27/02 Normalization

carid	Model	color
101	Bmw 1	Red
102	Bmw 2	Blue
103	Bmw 3	Green
104	Bmw 4	Blue
105	5	Red.
106	6	

← Mode 1

model 2 ↗ Primary key

Card_id	model	color_id
101	1	1
102	2	2
103	3	3
104	4	1
105	5	1
106	6	2

↘ foreign key

color_id	color_Name
1	Red
2	Blue
3	Green
4	

	model	model 2
size	large	small
Data facing	simple	complex
DB Design	simple	complex
performance	low (Denormalization)	Better (Normalized)
	↓ OLAP - Easy data facing	↓ OLTP - data frequently add/ update / delete

* Normalization is the process of efficiently organizing the data in the dB.
- used to minimize redundancy.
- used to eliminate the undesirable characteristics like insertion, updation, Deletion
- Divides larger table into smaller table & links them using relationship.

Normal form:- 1NF, 2NF, 3NF, 4NF, BCNF, 5NF, 6NF, 7NF.

full → first Name last name
Partial → Name Qualification

e.g.

ID	Name	Qualification
101	Sumedha	BE
102	Vipin	BE, ME
103.	Viha	BE, M.E, PhD.

28/02

1NF

ID	Name	Qu.
101	Sumedha	BE
102	Vipin	BE
102	Vipin	ME
103	Viha	BE
103	Viha	ME
103	Viha	PhD

- Attributes of table can't hold multiple values

2NF
- should be in 1NF
- Should not have partial dependancy

Dependancy - full, Partial, transitive

ID	Name	Qu_id	Qu
101	Sumedha	1	BE
102	Vipin	2	ME
103	Viha	3	PhD

Name → partitially dependent on qualification

ID_qu_id	ID	Qu_id
1	101	B 1
2	102	1
3	102	2
4	103	1
5	103	2
6	103	3

3NF → should be in 2NF
Doesn't have transitive Dependancy

ID	Name	city	state	country
101	S	Newyork	US	US
102	P	London	UK	UK
103	Y	Pune	MH	MH IN.

ID	Name	city_id
101	S	1
102	P	2
103	V	3

city_id	city	state	contry
1	New	US	US
2	London	UK	UK
3	Pune	MH	IN
4	mum	MH	IN
5	Ng	MH	IN

2NF

State id	State	contry
1	US	US
2	UK	UK
3	MH	IN.

country

country_id	contry
1	US
2	UK
3	IN

03/03. Data Models.

→ Data models tells how the logical structure of a DB is designed.

→ it is defined as how data is connected to each other & how it will be proceed and stored inside the system

→ Types of data model

1. conceptual
2. logical
3. physical

1. conceptual design / model

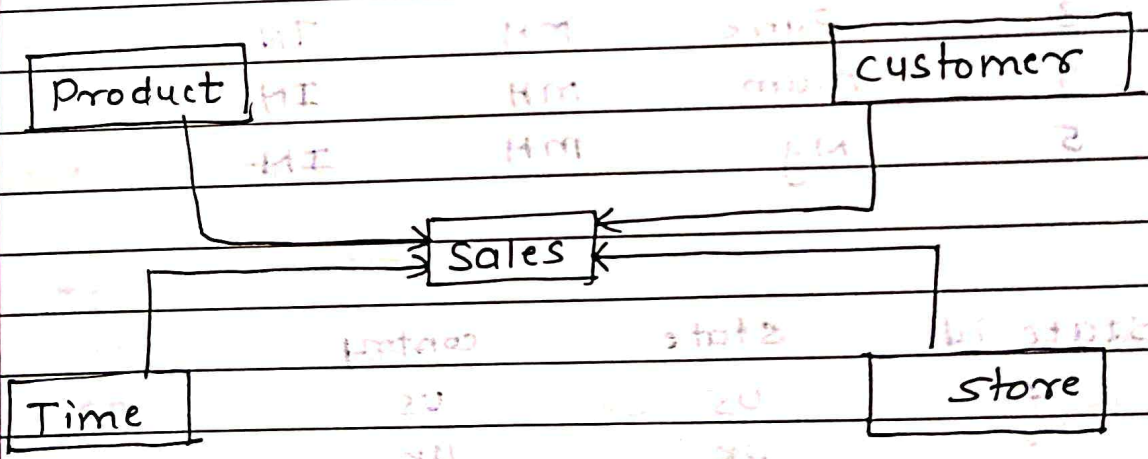
— it is high level design of DB

— features :- Display imp entities & relationship among them

:- No attributes is specifies

:- No primary key is specifies

:- level 1



2. logical data model.

- It is define. the data as much as possible to show they can be physically implemented in dB
- Detail info
- table name, col name, datatype, Pk.fk

→

* logical data model

→ design by Architect.

colname ↓ Datatype ↓ Null ↓ constraints ↓ sales manmt ↓

1. include all entities & relationship among them
2. All attributes for each entity are specified.
3. primary key, foreign key specified. also - constraints.

→ Physical model

- Actual implementation of logical model into database is called physical model.

→ Diff b/w Relational Datamode & Dimensional Datam...

Relational.	Dimensional
- Suitable for OLTP application.	- suggested for DWH applications.
- many number of table with chain relation among them	- fixed str, fact table, Dimension table
- CRUD activity	- only Read.
- Normalised.	- De-Normalised.

04/03

Fact Dimension

Batch.	Number of student
T_9	31
T_{10}	29
T_{11}	30
T_{12}	40
T_{13}	41
Total	171

order NO.	Date	Product	city	Qt	Total price
981	16/02	Laptop	Pune	9	11400 $
982	17/02	PD	Mum	2	1300 $
				- 11	- 12700 $

Dimension — fact.

fact table
Primary key present in Dimension table & their foreign key present in fact table

fact
It is counted or measured event

Dimension
It contain refferential info about fact.

fact table
- consist of measurements of facts of a business process
- it is central table in dimension model surrounded by dimension table

- A fact table typically has two types of col.
  i. Those that contain facts
  ii. Those that are a foreign key to dimension table.

→ Dimension table
used to described dimensions.

06/03

- Types of Dimension model
  i. Star (schema) → logical arrangement)
  ii Snowflack
  iii. Galaxy

1. Star
   - it is simplest form
   - In this, central table is called fact table & radically connect other table are called dimension table
   - Entity relationship diagram is look like star
   - De-normalized form
   - It is good for data mart with simple relationship.

2. Snowflack
   - The process of normalizing dimension table is called snowflaking
   - Dimension table is in normalized form
   - It is extension of star schema
   - ER diagram is look like snowflake

3. Galaxy | fact constellation
   collection of stars
   - contain two or more fact table that share same dimension.
   - looks like collection of star

• Diff betn star & snowflake

	star	snowflake
1.	contain less no. of rows	i. Big in size
2.	Good for datamart	2. Good for DWH
3.	Simple DB design	3. complex
4.	less easy to maintain	4. Easier to maintain
5	contain only single dimension	5 more than one
6.	fact & Dimension both are in de-normalized form	6. Dimension - Normalized fact - DeNor.

* Types of fact :-

1. Additive - Additive facts are facts can be summed up through all of the dimensions in the fact table

ii. Non-add - that cannot be summed up for any of the dimensions present in the fact table

iii semi-add - that can be summed up for some of the dimensions in the fact table, but not with all.

07/03

* Types of Dimensions :-

(i.) Slowly changing Dimension

- Dimensions that changes slowly over a period of time, rather than changing on regular schedule
- is a dimension that stored & manages both current & Historical data over time of in a DWH
- It is considered & implimented as one of the most critical ETL task in tracking the history of dimension record

types SCD

SCD - 0 - Passive Method
SCD - 1 - overwriting the old value
SCD - 2 - Creating New additional record
SCD - 3 - Adding New col
SCD - 4 - using Historical table
SCD - 5 - combine 1,2,3

* SCD 0

- NO special Action is performed upon dimensional changes.
- Dimension data that is remain some as it was first time inserted

Before

	101	Rohit	paris	2011

After

	101	Rohit	Newyork	2008

* SCD - 1

- old value is simply replaced
- only new value maintained
- No history
- easy to maintain, e.g- correction of spellings, Special characters.

Before

	101	Rohit	paris	2011

After

	101	Rohit	paris	2011

* SCD-2

- old & New value is present in same table
- New row is inserted for new value.
- All history maintain

Before

101	Rohit	Newyork	2008

After

101	Rohit	Newyork	2008
101	Rohit	Paris	2011

* SCD-3

- old and new value is kept in same table, same row
- New value located into 'new' col and old one into 'previous' col.
- History is limited
- least commonly needed technique

Before

101	Rohit	Newyork	2008

After

101	Rohit	Paris	2011	Newyork	2008

* SCD-4

- Separate table are there for old value & new value
- Separate history table is used to track all History
- Main table keeps only new data.

Before

101	Rohit	Newyork	2008

After

current - 101 Rohit Paris 2011

History - 101 Rohit Newyork 2008

* SCD - 6

- combine approaches of 1, 2, 3

:- current_address, current-year - for keeping current va

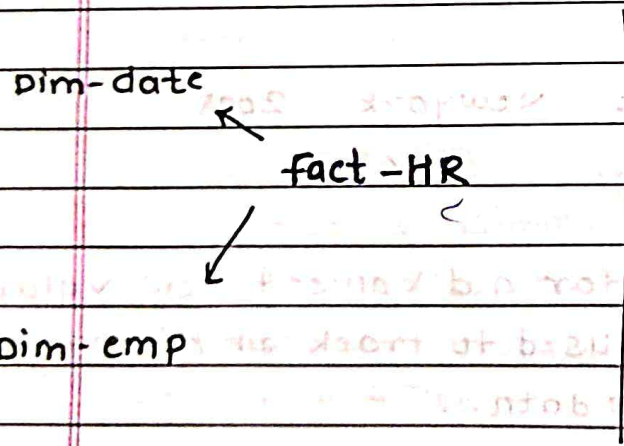
:- previous_address, previous-year - History value

:- current_flag - for keeping information about the most recent record.

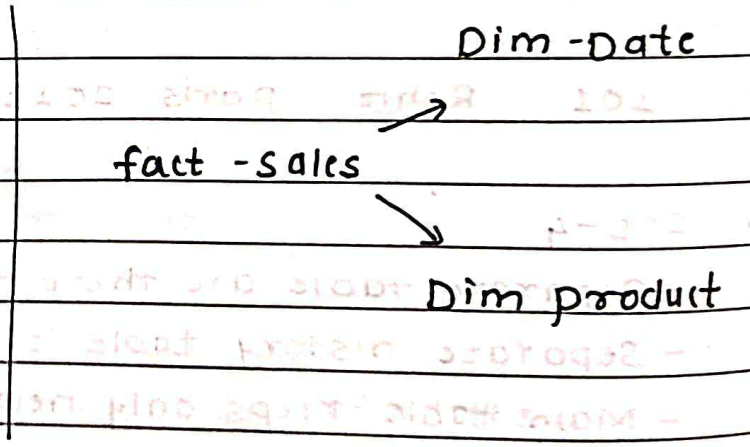
ID	Name	add	year	prev.add	year	flag.
101	Rohit	Mum	2020	Newyork	2008	N
101	Rohit	Mum	2020	Paris	2011	N
101	Rohit	Mum	2020	Pune	2015	Y

(ii). conformed Dimension.

- conformed dimension are those dimensions which have been designing such way that the dimensions can be used accross many fact-tables (Data mart) in different subject areas of DWH.



- calender year
Jan.
Monday - friday
English



- year - Budget year
April
Sun - Sat
french

- instead of creating two dimensional table, create only one

(iii) Degenerated Dimension
- which are directly present in fact table not in separate dimension table.

(iv) Junk Dimension.
- when group of independent dimensions store in one dimension table that dimension table is called as Junk dimension.

03/03

Surrogate key
- primary key madeup of real data in table, that called as Natural key.
- Sometimes in Database we cannot make primary key from real data
- we add one artificial col in table which is unique & not null
- This primary key which is generated from artificial col is called surrogate key.

Mapping Documents
- It define relationship bet^n source data field to their related target data field which are involved in ETL pr
- Map bet^n source data & target data in ETL process.
- ETL tester needs mapping document because, while testing data on target table we have to refer data in source tables & mapping document having detail info about each and every table that is part of ETL process from source to target along with transforma logic.

ETL testing.

ETL - Extract - Transform - load
↳ writing the data in target
Reading data from dB
converting the extracted data to required from

- It is a process which defines how data is loaded from the source to the target system (DWH)

- All the data migration and data movement is done by ETL process.

- We are going to validate data that has been loaded from source system to the data warehouse, is uniform in term of quality, quantity and format.

purpose
→ To identify and mitigate general data errors during ETL process.
→ we need to ensure data accuracy at target system because if the data is not accurate, then the business decision will be wrong.

ETL Testing vs Database Testing.

	ETL	Database.
1.	We are validating data which is loaded from Transactional System to analytical system.	1. We are validating data in dB which is created due to front end operations of application user i.e - We are going to check impact of backend due to front end operations of user.

2.	ETL testing is normally performed on data in a DWH system.	2. DB testing is commonly performed on transactional system.

3. We have to validate data in both ETL testing and dB testing but approach and steps taken for completion are different.

* Types of ETL Testing

1. Metadata Testing - F - verification phase.
   - We have to validate physical model against its logical model
   - involves verification of
     - Table name
     - col name
     - col datatype
     - col datalength
     - constraints.

2. Data completeness testing - F - (validation phase)

   source - count - level 1 - After build.
   target - count
   also count(null) - check.

   - We are going to ensure that all the expected data is loaded in target (DWH) from the source system.
   - it involvs - checking & validating record counts, null count bet^n source & target.
   - also - aggregate function, filter, against incremental loa & Historical data.

- Initial load - When first time inserting data into target
- Incremental - insert only newly added or modified data from source.
- full load - Truncate all data at target and then insert all source data at target. (History not - maintain)

(iii) Data transformation testing. -f - val.

Source EM101 Patil Sumedha S

target EM101 patilsumedha S.

- format different - we cant compare.
transform source format like target or vice versa.
- We are going to assure source data which is conv as per business rule/ reqr is loaded correctly in target
- Data transformation can be
  - concatenation
  - Joins, splits
  - conditional
  - Aggregation
  - Data conversion.

(iv) Data quality testing -f - val.

- We are going to ensure accuracy of data in target system.
- involves checking f validation of
  i. Duplicate data , ii. Rejected Record
  iii. Data validation rule.
  iv. Data integrity.

- Duplicate data - Duplicate value
- Rejected - present at source, but because of transformati- not present at target.
- Validation rule - constraints e.g. from date < To date
- Data Integrity - check pk, fk.

Test scenarios - QA team
- statement describing the functionality of application to be tested
- Test condition / Test posibility
- what we have to test
- High level test case
- one test scenarios cover one or more test cases.

e.g. login page. - Homepage, id, pasd page.

Test cases.
- set of positive and Negative executable steps of a test scenario which has a set of objective, actions, expected result and actual result
- How to be tested
- To validate test scenario by executing set of steps

- Good test cases
  - should be short
  - easily understable
  - strong objective
  - must have expected result.

- Steps.
1. understanding the req^{r}
2. Test scenario
3. Test cases
4. Test execution will start after getting build (code) from dev. team
   i. Run ETL job/ ETL code
   ii. Validate data at target system

Development → ETL code → job → Tester.

ETL code - SSIS ETL packages run/ some of the ETL codes are in scripts and we have to execute that for loading data from source system to target system.

Example → Test scenario & case
e.g. login page.

- Test scenario
  1. Verify user login
- Test cases.

id	type	objective	Action	Expected result	Actual	status
1.	P	Verify user login	Enterid Paswd	user should login and go to homepage	not Showing Homepage	fail.

Tester roles & responsibility

1. understand logical flow Of the appl^{n} (Data → source → target)
2. understand and review database design (logical model vs physical model)

3. understand and review source to target data mapping document. (what transformation logic used)
4. Analysis on business rules / data transformation / validation rules provided by client.
5. create test plane (created by team leader)
6 Identify test scenarios from mapping document for ETL testing.
7. After that we have to developed test cases from scenario
8. create test data and write sql queries for all test cases
9. Review test cases.
10. After getting build from development team we are going to start executing test cases.
11. Run ETL job to load source data in target system.
12. Document test result and log the defect for failed test cases.
13. Retesting for faild test cases.
14. Regression testing for code changes.
(Defect fixed, Enhancement, Newfeature)
15. Test closure
16. Sign off from testing team to project manager and Development team.

* challenges
1. Data loss during ETL process.
S → T (excelt to dev. team)
(invalid data, time matter,) wrong implementation of code)
2. Large volume of data.
(time max reqr for large data)
soln → divide data.
3. Invalid and duplicate data at target system.
extra time, efforts reqr.
waste of time → stop testing.

4. Many No. of source system.
   - Data at multiple places - difficult.
5. source to target mapping information may not be provided to ETL tester.
   - old document.
6. Does not have permission to execute ETL code/Job.
7. unstable testing environment.
   (Remote testing)