



Virtual University of Pakistan



😊😐😞 **MUHAMMAD FAISAL** 😊😐😞

MIT 4th Semester

Al-Barq Campus (VGJW01) Gujranwala

faisalgrw123@gmail.com

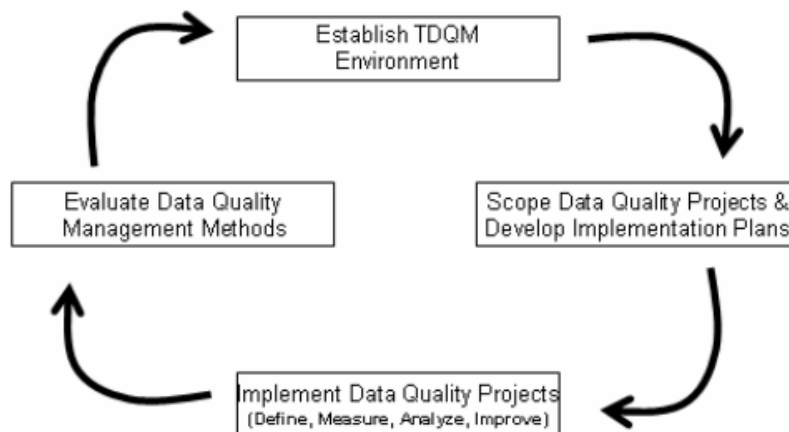
Reference Short Questions for Final TERM EXAMS

CS614 – Data Warehousing

Data Quality Management Process: (Page#180)

- 1) Establish Data Quality Management Environment
- 2) Scope Data Quality Projects & Develop Implementation Plan
- 3) Implement Data Quality Projects (Define, Measure, Analyze, Improve)
- 4) Evaluate Data Quality Management Methods

Data Quality Management Process



How to improve Data Quality? (Page#183)

The **four categories** of **Data Quality Improvement**

- 1) Process
- 2) System
- 3) Policy & Procedure
- 4) Data Design

Quality Management Maturity Grid: (Page#184)

There are **five stages** and these are:

Stage 1: Uncertainty

Stage 2: Awakening

Stage 3: Enlightenment

Stage 4: Wisdom

Stage 5: Certainty

Parallelize: (Page#188)

Parallel execution improves processing for:

- Large table scans and joins
- Creation of large indexes
- Partitioned index scans
- Bulk inserts, updates, and deletes
- Aggregations and copying

Speed -Up & Amdahl's Law: (Page#192)

Reveals maximum expected speedup from parallel algorithms given the proportion of task that must be computed sequentially. It gives the speedup S as

$$S \leq \frac{1}{f + (1 - f)/N}$$

Parallelization OLTP Vs. DSS (Page#193)

There is a big difference.

DSS:

- Parallelization of a SINGLE query

OLTP:

- Parallelization of MULTIPLE queries
- Or Batch updates in parallel

Parallel Processing (Page#194)

Parallel Hardware Architectures:

Parallel Hardware Architectures consists of:

- ◆ Symmetric Multi-Processing (SMP)
- ◆ Distributed Memory or Massively Parallel Processing (MPP)
- ◆ Non-uniform Memory Access (NUMA)

Parallel Software Architectures:

Parallel Software Architectures consists of:

- ◆ Shared Memory
- ◆ Shard Disk
- ◆ Shared Nothing

Types of parallelism: (Page#194)

- Data Parallelism
- Spatial Parallelism

NUMA (Non-Uniform Memory Access): (Page#194)

In **NUMA** systems, some memory can be accessed more quickly than other parts, and thus called as **Non-Uniform Memory Access**. This term is generally used to describe a shared-memory computer containing a hierarchy of memories, with different access times for each level in the hierarchy.

SMP (Symmetric Multiprocessing): (Page#194)

SMP (Symmetric Multiprocessing) is a computer architecture that provides fast performance by making multiple CPUs available to complete individual processes simultaneously (multiprocessing).

Shared disk RDBMS Architecture: (Page#196)

Advantages:

- A benefit of the shared disk approach is it provides a high level of fault tolerance with all data remaining accessible even if there is only one surviving node.

Disadvantages:

- Maintaining locking consistency over all nodes can become a problem in large clusters.

Shared Nothing RDBMS Architecture: (Page#197)

Advantages:

- This works fine in environments where the data ownership by nodes changes relatively infrequently.
- There is no overhead of maintaining data locking across the cluster

Disadvantages:

- The data availability depends on the status of the nodes. Should all but one system fail, then only a small subset of the data is available.

How do you perform the partitioning? (Page#199)

- ♥ Hash partitioning
- ♥ Key range partitioning.
- ♥ List partitioning.
- ♥ Round-Robin
- ♥ Combinations (Range-Hash & Range-List)

Pipelining: Speed-Up Calculation: (Page#203)

Pipelining: Speed-Up Calculation

Time for sequential execution of 1 task = T

Time for sequential execution of N tasks = N * T

(Ideal) time for pipelined execution of one task using an M stage pipeline = T

(Ideal) time for pipelined execution of N tasks using an M stage pipeline = T + ((N-1) × (T/M))

$$\text{Speed-up (S)} = \frac{NT}{T + (N-1) \times \frac{T}{M}}$$

Parallel Sorting: (Page#205)

We do parallel sorting as,

- Scan in parallel, and range partition on the go.
- As partitioned data becomes available, perform “local” sorting.
- Resulting data is sorted and again range partitioned.

Skew in Partitioning: (Page#206)

Skew in Partitioning done through:

- Attribute-value skew
- Partition skew

Handling Skew in Range-Partitioning through:

- Sort
- Construct the partition vector
- Duplicate entries or imbalances

Barriers to Linear Speedup & Scale-up: (Page#206)

- ❖ Amdahl’ Law
- ❖ Startup
- ❖ Interference
- ❖ Skew

Conventional indexes: (Page#206)

Basic Types:

- ✦ Sparse
- ✦ Dense
- ✦ Multi-level (or B-Tree)

Dense Index (Advantages & Disadvantages): (Page#211)

Pro:

- A dense index, if fits in the memory, costs only one disk I/O access to locate a record given a key.

Con:

- A dense index, if too big and doesn't fit into the memory, will be expensive when used to find a record given its key.

Sparse Index (Advantages & Disadvantages) (Page#211)

- Store first value in each block in the sequential file and a pointer to the block.
- Uses even less space than dense index, but the block has to be searched, even for unsuccessful searches.
- Time (I/Os) logarithmic in the number of blocks used by the index.

B-tree Indexing: (Page#213)

Limitations of B-tree Indexing:

- ⊗ If a table is large and there are fewer unique values.
- ⊗ Capitalization is not programmatically enforced.
- ⊗ Outcome varies with inter-character spaces.
- ⊗ A noun spelled differently will result in different results.
- ⊗ Insertion can be very expensive.

Special Index Structures: (Page#219)

Special index structures are:

- ➔ Inverted index
- ➔ Bit map index
- ➔ Cluster index
- ➔ Join indexes

Bitmap Index: (Page#222)

The **advantages** of **Bitmap Index** are:

- ◆ Very low storage space
- ◆ Reduction in I/O, just using index
- ◆ Counts & Joins
- ◆ Low level bit operations

The **disadvantages** of **Bitmap Index** are:

- ◆ Locking of many rows
- ◆ Low cardinality
- ◆ Keyword parsing
- ◆ Difficult to maintain - need reorganization when relation sizes change (new bitmaps)

Cluster Index: (Page#225)

Here are the **issues** of **Cluster Index**:

- Works well when a single index can be used for the majority of table accesses.
- Selectivity requirements for making use of a cluster index are much less stringent than for a non-clustered index.
- Typically by an order of magnitude, depending on row width.
- High maintenance cost to keep sorted order or frequent reorganizations to recover clustering factor.
- Optimizer must keep track of clustering factor (which can degrade over time) to determine optimal execution plan

Join algorithms/techniques: (Page#226)

Here are the **join algorithms/techniques**:

- ☞ Nested loop join
- ☞ Sort Merge Join
- ☞ Hash based join

Nested-Loop Join: Cost Formula (Page#229)

Join cost = Cost of accessing Table_A +
of qualifying rows in Table_A × Blocks of Table_B to be scanned for each qualifying row

OR

Join cost = Blocks accessed for Table_A +
Blocks accessed for Table_A × Blocks accessed for Table_B

Nested-Loop Join: Variants (Page#230)

- ☆ Naive nested-loop join
- ☆ Index nested-loop join
- ☆ Temporary index nested-loop join

Sort-Merge Join: (Page#232)

The **features** of **Sort-Merge Join** are:

- ☞ Very fast
- ☞ Sorting can be expensive
- ☞ Presorted data can be obtained from existing B- tree

The optimizer uses a hash join to join two tables if they are joined using an equi-join and if either of the following conditions are true: (Page#233)

- A large amount of data needs to be joined.
- A large portion of the table needs to be joined.

Data Mining: (Page#237)

In Data Mining there is:

- ③ Knowledge Discovery in Databases (KDD).
- ③ Data mining digs out valuable non-trivial information from large multidimensional apparently unrelated data bases (sets).
- ③ It's the integration of business knowledge, people, information, algorithms, statistics and computing technology.
- ③ Finding useful hidden patterns and relationships in data.

(Page#242)

Requires solution of fundamentally new problems related **data mining**, grouped as follows:

- ✱ developing algorithms and systems to mine large, massive and high dimensional data sets;
- ✱ developing algorithms and systems to mine new types of data (images, music, videos);
- ✱ developing algorithms, protocols, and other infrastructure to mine distributed data; and
- ✱ improving the ease of use of data mining systems;
- ✱ developing appropriate privacy and security techniques for data mining.

Data Mining is... (Page#246)

- Decision Trees
- Neural Networks
- Rule Induction
- Clustering
- Genetic Algorithms

In **CLUSTERING** there is: **(Page#251)**

- ☆ Task of segmenting a heterogeneous population into a number of more homogenous sub-groups or clusters.
- ☆ Unlike classification, it does NOT depend on predefined classes.
- ☆ It is up to you to determine what meaning, if any, to attached to resulting clusters.
- ☆ It could be the first step to the market segmentation effort.

Here are the **examples of Clustering Applications:** **(Page#251)**

- ✓ Marketing
- ✓ Insurance
- ✓ Land use
- ✓ Seismic studies

Comparing Methods: **(Page#254)**

- Predictive accuracy
- Speed
- Robustness

Main types of DATA MINING: **(Page#257)**

Supervised:

- Bayesian Modeling
- Decision Trees
- Neural Networks etc

Unsupervised:

- One-way Clustering
- Two-way Clustering

A graph can be clustered by: (Page#259)

- ➡ Partitioning the graph
- ➡ Finding cliques in the graph

The K-Means Clustering (Page#267)

Given k , the **k-means algorithm** is implemented in **4 steps**:

- ➔ Partition objects into k nonempty subsets.
- ➔ Compute seed points as the centroids of the clusters of the current partition. The centroid is the center (mean point) of the cluster.
- ➔ Assign each object to the cluster with the nearest seed point.
- ➔ Go back to Step 2, stop when no more new assignment.

The K-Means Clustering Strength & Weakness: (Page#268)

Strength:

The **strength** of **K-Means Clustering**:

- ✓ Relatively efficient.
- ✓ Often terminates at a local optimum. The global optimum may be found using techniques such as: deterministic annealing and genetic algorithms

Weakness:

The **weakness** of **K-Means Clustering**:

- ☞ Applicable only when mean is defined, then what about categorical data?
- ☞ Need to specify k, the number of clusters, in advance.
- ☞ Unable to handle noisy data and outliers.
- ☞ Not suitable to discover clusters with non-convex shapes.

Implementation strategies: (Page#270)

- ☛ Top down approach
- ☛ Bottom Up approach

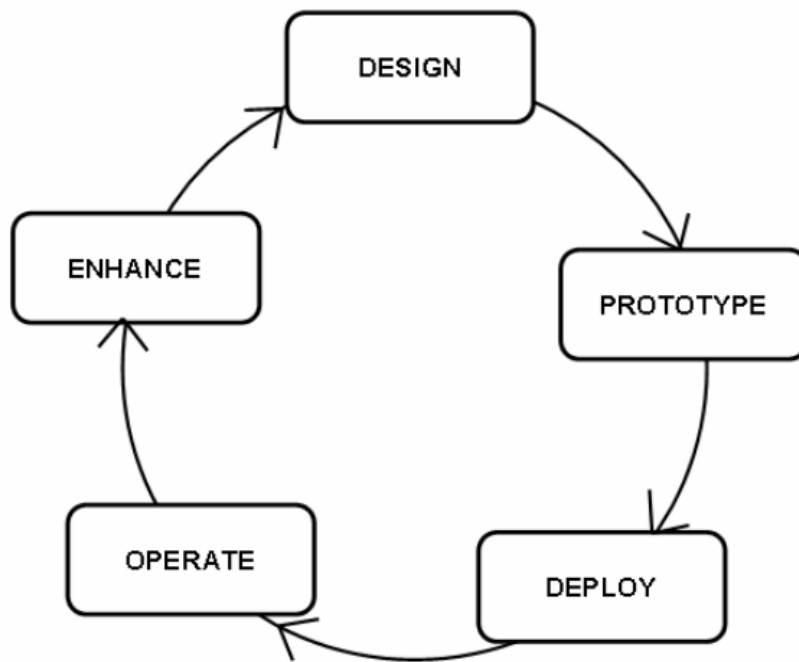
Development methodologies: (Page#270)

- ☛ Waterfall model
- ☛ Spiral model
- ☛ RAD Model
- ☛ Structured Methodology
- ☛ Data Driven
- ☛ Goal Driven
- ☛ User Driven

5 keys to a successful rapid prototyping methodology: (Page#271)

- Assemble a small, very bright team of database programmers, hardware technicians, designers, quality assurance technicians, documentation and decision support specialists and a single manager.
- Define and involve a small "focus group" consisting of users and managers.
- Generate a user's manual and user interface first.
- Use tools specifically designed for rapid prototyping.
- Remember a prototype is NOT the final application.

Data Warehouse (DWH) Development Cycle: (Page#274)



DWH Development Cycle

Data Warehouse Development Cycle consists of **5 major steps** described as follows:

- ⇒ Design
- ⇒ Prototype
- ⇒ Deploy
- ⇒ Operation
- ⇒ Enhancement

DWH Lifecycle: Key steps: (Page#274)

1. Project Planning
2. Business Requirements Definition
3. Parallel Tracks
4. Lifecycle Data Track
5. Deployment
6. Maintenance

DWH Lifecycle: Project Planning

In the DWH Lifecycle in step 1 Project Planning includes:

- ❖ Assessing Readiness
- ❖ Factors
- ❖ Scoping
- ❖ Justification
- ❖ Team development
- ❖ Project Plan

In the DWH Lifecycle- Step: 2 Requirements Definition includes:

- ❖ Requirements preplanning
- ❖ Requirements collection
- ❖ Post collection
- ❖ Forum
- ❖ Requirements team
- ❖ Business representatives
- ❖ Post Collection

Prioritization: (Page#285)

In prioritization there is,

- Review & Prioritize findings
- Quadrant Technique

DWH Lifecycle- Step 3.1: Technology Track (Page#287)

It consists of **8-Step Process**:

- 1) Establish an Architecture Task k Force (2-3 people)
- 2) Collect Architecture-Related Requirements
- 3) Document Architecture Requirements
- 4) Develop a high-level Arch. Model
- 5) Design and Specify the Subsystems
- 6) Determine Architectural implementation phases
- 7) Document the Technical Architecture
- 8) Review and finalize the technical architecture

DWH Lifecycle- Step 3.1: Technology Track (Page#291)

The Product selection and Installation includes:

- Understand corporate purchasing process
- Product evaluation matrix
- Market research
- Narrow options, perform detailed evaluations
- Conduct prototype, if necessary
- Keep the competition “hot”
- Select product, install on trial, and negotiate

DW Lifecycle- Step 4: Deployment (Page#295)

- The three tracks converge at deployment.
- Should serve uncooked data
- Not natural, require substantial pre -planning, courage, will-power and honesty.
- Other than data readiness, education and support are critical.
- Educate on complete warehouse deliverable
- For effective education

DW Lifecycle- Step 5: Maintenance and Growth (Page#296)

- Support
- Education
- Technical support
- Program support
- Growth

Possible Pitfalls: (Page#299)

Here are the 11 **Possible Pitfalls**:

- 1) Weak business sponsor
- 2) Not having multiple servers
- 3) Modeling without domain expert
- 4) Not enough time for ETL
- 5) Low priority for OLAP Cube Construction
- 6) Fixation with technology

- 7) Wrong test bench
- 8) QA people NOT DWH literate
- 9) Uneducated user
- 10) Improper documentation
- 11) Doing incremental enhancements

Top 10-Common Mistakes to Avoid: (Page#303)

Here are the **Top 10-Common Mistakes to Avoid:**

1. Not interacting directly with the end users.
2. Promising an ambitious data mart as the first deliverable.
3. Never freezing the requirements i.e. being an accommodating person.
4. Working without senior executives in loop, waiting to include them after a significant success.
5. Doing a very comprehensive and detailed first analysis to do the DWH right the very first time.
6. Assuming the business users will develop their own “killer application” on their own.
7. Training users on the detailed features of the tool using dummy data and consider it a success.
8. Isolating the IT support people from the end or business users.
9. After DWH is finished, holding a planning and communications meeting with end users.
10. Shying away from operational source systems people, assuming they are too busy.

Key Steps for a smooth DWH implementation: (Page#305)

Here are the top 7-Key Steps for a smooth DWH implementation:

Step-1: Assigning a full-time project manager, or doing it yourself full- time.

Step-2: Consider handing-off project management.

Step-3: During user interview don't go after answers, let the answers come to you.

Step-4: Assigning responsibilities to oversee and ensure continuity.

Step-5: Accept the “fact” that DWH will require many iterations before it is ready.

Step-6: Assign significant resources for ETL.

Step-7: Be a diplomat NOT a technologist.

The **conclusions** of **DWH** are as under: **(Page#307)**

- DWH is not simple.
- DWH is very expensive.
- DWH is not ONLY about technology.
- DWH designers must be capable of working across the organization.
- DWH team requires a combination of many experiences and expertise.

Large and Typical Early Adopters are: **(Page#310)**

- Financial service/insurance.
- Telecommunications.
- Transportation.
- Government.
- Educational

Example DWH Target Organizations: **(Page#311)**

- Financial service/insurance
- Telecommunications
- Transportation
- Government

Motivation for Transformation: **(Page#319)**

- Trivial queries give wrong results.
- Static and dynamic attributes
- Static attributes recorded repeatedly.

Reasons for web warehousing: (Page#328)

- Searching the web (web mining)
- Analyzing web traffic
- Archiving the web

Web searching: (Page#328)

Three major types of searches, as follows:

1. Keyword-based search
2. Querying deep Web sources
3. Random surfing

Drawbacks of traditional web searches: (Page#329)

1. Limited to keyword based matching.
2. Can not distinguish between the contexts in which a link is used.
3. Coupling of files has to be done manually.

Where does traffic info. come from? (Page#334)

1. Log files.
2. Cookies.
3. Network traffic.
4. Page tagging.
5. ISP (Internet Service Provider)
6. Others

Web log file formats: (Page#336)

Format of web log dependent on many factors, such as:

- Web server
- Application
- Configuration options

Dimensions for W DWH: Page (Page#339)

The **dimensions** for **Web Data Warehouse Page** are:

- Describes the page context for a Web page event
- Definition of page must be flexible
- Assume a well defined function
- The page dimension is small

The **principal sources** of **web traffic** are as follows: (Page#334)

1. Log files.
2. Cookies.
3. Network traffic.
4. Page tagging.
5. ISP

Issues of Clickstream Data: (Page#341)

Clickstream data has many **issues**:

- ❖ Identifying the Visitor Origin
- ❖ Identifying the Session
- ❖ Identifying the Visitor
- ❖ Proxy Servers
- ❖ Browser Caches

Using HTTP's secure sockets layer (SSL): (Page#344)

Offers an opportunity to track a visitor session

Limitations:

- To track the session, the entire information exchange needs to be in high overhead SSL
- Each host server must have its own unique security certificate.
- Visitors are put-off by pop-up certificate boxes.

Using session ID Ping-pong: (Page#344)

- ➔ Maintain visitor state by placing a session ID in a hidden field of each page returned to the visitor.
- ➔ Session ID can be returned to the Web server as a query string appended to a subsequent URL.

Limitations:

- Requires a great deal of control over the Web site's page-generation methods
- Approach breaks down if multiple vendors are supplying content in a single session

Using Persistent Cookies: (Page#345)

Establish a persistent cookie in the visitor's PC

Limitations:

- No absolute guarantee that even a persistent cookie will survive.
- Certain groups of Web sites can agree to store a common ID tag

Proxy servers: (Page#346)

- An HTTP request is not always served from the server specified in a URL.
- Many ISPs make use of proxy servers to reduce Internet traffic.

Forward Proxy: (Page#347)

The type of proxy we are referring to in this discussion is called a forward proxy. It is outside of our control because it belongs to a networking company or an ISP.

Reverse Proxy:

- It can be placed in front of our enterprise's Web servers to help them offload requests for frequently accessed content.
- This kind of proxy is entirely within our control and usually presents no impediment to Web warehouse data collection.

DTS Basics: (Page#356)

- DTS Packages
- DTS Tasks
- DTS Transformations
- DTS Package Workflows
- DTS Tools
- Meta Data

DTS Package: (Page#357)

DTS Package is an organized collection of:

- Connections
- DTS tasks
- DTS transformations
- Workflows

DTS Packages 4 Operations: (Page#365)

Packages can be:

- Edited
- Password protected
- Scheduled for execution
- Retrieved by version

DTS Tasks: (Page#368)

DTS Package contains **one or more tasks**.

Task defines single work item:

- Establishing connections
- Importing and exporting data
- Transforming data
- Copying database objects

Available transformations are: (Page#274)

- Copy column transformation
- ActiveX Script transformations
- Date time string transformations
- Uppercase and lowercase string transformations
- Middle of string transformations
- Read and write file transformations

DTS Connections (Data Source, File Connection, Data Link) (Page#372)

All OLE DB supported databases:

- MS SQL Server
- Oracle
- MS Access 2000

File Connection:

- Text files

Data link connection:

- Intermediate files containing connection strings

Steps towards single source of truth: (Page#389)

- Identify source systems
- Figure out the issues associated with each source system
- Extract data
- Transform data
- Load data
- Quality checks

Seven Steps to Extract Data Using Wizard: (Page#392)

Here are the **Seven Steps to Extract Data Using Wizard:**

- 1) Launch the Wizard
- 2) Choose a Data Source
- 3) Choose a Database
- 4) Specify the Destination
- 5) Choose Destination Database
- 6) Select a table
- 7) Finalizing and Scheduling the package

Execution of a package: (Page#398)

The **execution of a package** is done through:

- Connection with source (Text file) is established
- Connection with destination (MS -SQL Server) is established
- New Database at destination is created
- New table is created
- Data is extracted from source
- Data is loaded to destination

Data Standardization: (Page#418)

Before **combining all tables** we need to **standardize** them:

- Number and types of columns, date formats and storing conventions all of them should be consistent in each table.
- The process of standardization requires transformation of data elements.
- To identify the degree of transformation required we will perform data profiling.

Data profiling: (Page#420)

Data profiling includes:

- Identify erroneous records.
- Copy erroneous records to Exception table and set dirty bit of erroneous records in student table of a campus.

Data profiling is a process which involves gathering of information about column through execution of certain queries with intention to identify erroneous records. In this process we identify the following:

- Total number of values in a column
- Number of distinct values in a column
- Domain of a column
- Values out of domain of a column
- Validation of business rules

Process of Data Profiling: (Page#421)

Data profiling, gathering information about columns fulfils the following **two purposes**:

- Identify the type and extent to which transformation is required
- Gives us a detailed view of data quality

Data profiling is required to be **performed twice**:

- Before applying transformations
- After applying transformations

Handling Dates: Problem: (Page#428)

While profiling we need to run queries to identify:

- Inconsistencies in date formats
- Invalidities like 29th Feb 1975
- Missing values of dates
- Violations in business rules like 10 years student in graduating class

Data Quality: Multiple Inconsistencies (Page#431)

- If any record is selected whose dirty bit is already set we will not copy it again to exception table rather we will modify the comment of Exception table.
- But in this query all selected records have their dirty bits off.
- So copy all record to Exception table and set their dirty bits on in Student table.

How to Correct the Exception Table: (Page#438)

How to correct the exception table, it depends upon the factors like:

- Number of records corrupted
- Type of corruption or error
- Educated guess
- Using golden copy

Transformation Task Properties: (Page#447)

- Press third tab transformation
- Designer tries to map source destination column automatically
- Drop all mappings if Designer asks you through a dialog to delete all mappings

-----Wish U Best of L|U|C|K for EXAMS -----

MUHAMMAD FAISAL

MIT 4TH SEMESTER

VIRTUAL UNIVERSITY OF PAKISTAN