WINTER – 2019 EXAMINATION
MODEL ANSWER

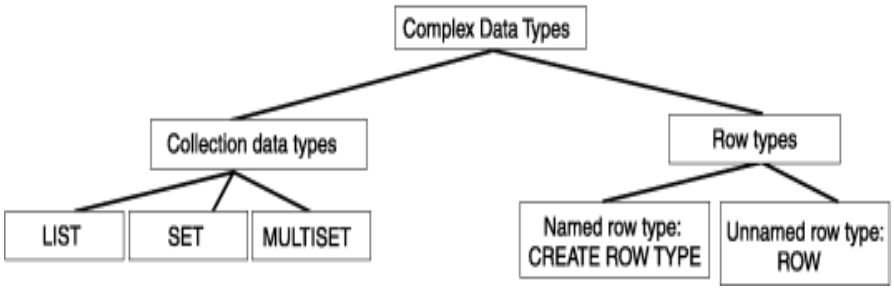
Subject: Advanced Database Management Systems (Elective I) **Subject Code: 22521**

Important Instructions to examiners:

1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills).
4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q. No.	Sub Q.N.	Answer	Marking Scheme
1.	**(a) Ans.**	**Attempt any FIVE of the following:** **Explain the use of two phase locking protocol.** • It offers Concurrency Control • Two Phase Locking offers serializability. • It checks for local and global deadlocks and solve them by resuming transactions to their initial states.	**10** **2M** ***Any 2 Use 1M each***
	(b) Ans.	**Write any two benefits of distributed database system.** There are following advantages of DDBMs: • Reflects organizational structure • Improved share ability and local autonomy • Improved availability • Improved reliability • Improved Performance • Economics • Modular growth	**2M** ***Any 2 Benefits 1M each***
	(c) Ans.	**Define complex data types.** A complex data type is usually a composite of other existing data types. For example, you might create a complex data type whose components	**2M**

include built-in types, opaque types, distinct types, or other complex types. An important advantage that complex data types have over user-defined types is that users can access and manipulate the individual components of a complex data type.



Definition 2M

(d) Ans.

Enlist types of functions in SQL.

There are two types of SQL functions, aggregate functions, and scalar (non-aggregate) functions.

- SQL Aggregate Function
- SQL Arithmetic Function/SQL Mathematical Function
- SQL Character Function

2M

Any 2 Functions 1M each

(e) Ans.

State the difference between structured versus unstructured data types.

Parameters	Structured Data type	Unstructured Data Type
Technology	Relational Database Table(s)	Character and binary data
Transaction Management	Matured Transaction also various concurrency techniques	No transaction management, No concurrency
Version Management	Version over tuples, rows tables etc.	Versioned as whole
Flexibility	Schema dependent rigorous schema.	Very flexible due to absence of schema
Scalability	Scaling database is difficult	Very scalable
Robustness	Very robust.	Does not support robustness
Query Performance	Structured Query allows complex joins	Only textual queries are possible.

2M

Any 2 points of different iation 1M each

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Advanced Database Management Systems (Elective I) **Subject Code: 22521**

	(f) Ans.	**Describe the benefits of No-SQL.** **Benefits of No-SQL:** • Less need for Extract, Transform and Load (ETL) • Support for unstructured text • Ability to handle change over time • No reliance on SQL magic • Easy to Scale. • Breadth of functionality • Support for multiple data structures • Vendor choice • No legacy code • Executing code next to the data • Schema less • Adhoc queries support • Indexing • Replication : Provides high availability with replica set. • Grid file storage system	**2M** ***Any 4 benefits ½M each***
	(g) Ans.	**Describe the application of data mining.** **(Any relevant applications can be considered)** **Healthcare** Data mining holds great potential to improve health systems. It uses data and analytics to identify best practices that improve care and reduce costs. **Market Basket Analysis** Market basket analysis is a modelling technique based upon a theory that if you buy a certain group of items you are more likely to buy another group of items. **Education** There is a new emerging field, called Educational Data Mining, concerns with developing methods that discover knowledge from data originating from educational Environments. **Manufacturing Engineering** Knowledge is the best asset a manufacturing enterprise would possess. Data mining tools can be very useful to discover patterns in complex manufacturing process. **CRM** Customer Relationship Management is all about acquiring and retaining customers, also improving customers' loyalty and implementing customer focused strategies. To maintain a proper relationship with a	**2M** ***Any 2 applications 1M each***

		customer a business need to collect data and analyse the information. **Fraud Detection** Billions of dollars have been lost to the action of frauds. Traditional methods of fraud detection are time consuming and complex. Data mining aids in providing meaningful patterns and turning data into information. **Intrusion Detection** Any action that will compromise the integrity and confidentiality of a resource is an intrusion. **Financial Banking** With computerised banking everywhere huge amount of data is supposed to be generated with new transactions. **Research Analysis** History shows that we have witnessed revolutionary changes in research. Data mining is helpful in data cleaning, data pre-processing and integration of databases.	
2.	**(a) Ans.**	**Attempt any THREE of the following:** **Explain Concurrency Control Techniques.** There are different concurrency control techniques such as: • Lock based protocols • Two phase Locking protocols • Time stamp based protocols • **Lock based protocol :** To ensure serializability it requires that th data items be accessed in a mutually exclusive manner. i.e. While one transaction is accessing a data item, no other transaction can modify that data. Method used to implement this requirement is to allow transaction to access a data item only if it is currently holding a lock on that item. **Locks:** Lock is a data variable which is associated with a data item. Locks help synchronize access to the database items by concurrent transactions. All lock requests are made to the concurrency-control manager. Transactions proceed only once the lock request is granted. There are different types of locks: **Binary loc:** A binary lock on a data item can either have locked or unlocked states.	**12** **4M** ***List 1M*** ***Explanation 3M***

Shared Lock: A shared lock is also called as Read only lock.
With the shared locks data items can be shared between transactions.
Because with shared locks you will never have permission to update data on the data item.
Shared lock is denoted by S.

Exclusive Lock: With the exclusive lock a data item can be read as well as written.
This lock can't be held concurrently on the same data item.
It is denoted by X.
Exclusive lock is requested using lock-X instruction.

- **Two phase Locking protocol:** which is also known as 2PL. Two phase locking protocol requires that each transaction issues lock and unlock requests in two phases:

Growing phase: A transaction may obtain locks but may not release any lock.
Shrinking phase: A transaction may release locks, but may not obtain any new locks.
If the conversion is allowed, then upgrading of locks from S(A) to X(A) happens in growing phase and the downgrade of locks from X(A) to S(A) happens in shrinking phase.
It is true that 2PL protocol offers serializability. However it does not ensure that dead locks not happen.

- **Time stamp based protocols:** The timestamp-based algorithm uses a timestamp to serialize the execution of concurrent transactions.

This protocol ensures that every read and write operations are executed in timestamp order.
These protocol uses the System Time or logical count as a timestamp.
The older transaction is always given priority in this method.
This is the most commonly used concurrency protocol.
E.g:
Suppose there are transactions T_1, T_2 and T_3
T_1 has entered the system at time 0010
T_2 has entered the system at 0020
T_3 has entered the system at 0030
Thus the priority will be given to transaction T_1, then transaction T_2 and then lastly to Transaction T_3.

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Advanced Database Management Systems (Elective I) **Subject Code: 22521**

	(b) Ans.	**Explain multiset types in SQL.** Multiset is a collection type. Multiset is much like an array, but unordered and it has more operators. Unlike arrays, multisets never have a declared maximum cardinality thus it does not have ordinal position or index. ***Consider example:*** Create table book (book_id char (5), book_title varchar (50), book_author varchar (30), book_edition integer array [10], book_rating integer multiset); In this example book_rating attribute is implemented as multiset because it is assumed that a book can have same rating several times.	**4M** ***Explanation of multiset types in SQL 3M*** ***Example 1M***
	(c) Ans.	**Compare SQL and No-SQL database system.**	**4M**

Sr. No.	SQL	No-SQL
1	SQL databases are primarily called as Relational Databases (RDBMS)	NoSQL database are primarily called as non-relational or distributed database.
2	SQL databases are table based databases	NoSQL databases are document based, key-value pairs, graph databases or wide-column stores
3	SQL databases have predefined schema	NoSQL databases have dynamic schema for unstructured data
4	SQL databases are vertically scalable	NoSQL databases are horizontally scalable.
5	SQL databases uses SQL (structured query language) for defining and manipulating the data, which is very powerful	NoSQL database, queries are focused on collection of documents. Sometimes it is also called as UnQL (Unstructured Query Language)
6	SQL database examples:	NoSQL database examples:

Any four comparison 1M each

			MySql, Oracle, Sqlite, Postgres and MS-SQL	MongoDB, BigTable, Redis, RavenDb, Cassandra, Hbase, Neo4j and CouchDb
		7	SQL databases are good fit for the complex query intensive environment	NoSQL databases are not good fit for complex queries.
		8	SQL databases are not best fit for hierarchical data storage.	NoSQL database fits better for the hierarchical data storage as it follows the key-value pair way of storing data similar to JSON data.
			SQL databases emphasizes on ACID properties	NoSQL database follows the Brewers CAP theorem
		9	SQL databases as either open-source or close-sourced from commercial vendors.	NoSQL databases can be classified on the basis of way of storing data as graph databases, key-value store databases, document store databases, column store database and XML databases.
	(d) Ans.	**Explain XML document schema.** Databases have schemas, which are used to constrain what information can be stored in the database and to constrain the data types of the stored information. e the first schema-definition language included as part of the XML standard, the Document Type Definition, as well as its more recently defined replacement, XML Schema. Another XML schema-definition language called Relax NG is also in use. XML Schema defines a number of built-in types such as string, integer, decimal date, and boolean. In addition, it allows user-defined types; these may be simple types with added restrictions, or complex types constructed using constructors such as complex Type and sequence The first thing to note is that schema definitions in XML Schema are themselves specified in XML syntax, using a variety of tags defined by XML Schema. To avoid conflicts with user-defined tags, we prefix the XML Schema tag with the namespace prefix "xs:"; this prefix is associated with the XML Schema namespace by the xmlns:xs specification in the root element:	**4M** ***Description of XML Document schema 4M***

		<xs:schemaxmlns:xs="http://www.w3.org/2001/XMLSchema"> Note that any namespace prefix could be used in place of xs; thus we could replace all occurrences of "xs:" in the schema definition with "xsd:" without changing the meaning of the schema definition. All types defined by XML Schema must be prefixed by this namespace prefix. ***Example:*** <xs:schemaxmlns:xs="http://www.w3.org/2001/XMLSchema"> <xs:element name="university" type="universityType" /> <xs:element name="department"> <xs:complexType> <xs:sequence> <xs:element name="dept name" type="xs:string"/> <xs:element name="building" type="xs:string"/> <xs:element name="budget" type="xs:decimal"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="course"> <xs:element name="course id" type="xs:string"/> <xs:element name="title" type="xs:string"/> <xs:element name="dept name" type="xs:string"/> <xs:element name="credits" type="xs:decimal"/> </xs:element> <xs:element name="instructor"> <xs:complexType> <xs:sequence> <xs:element name="IID" type="xs:string"/> <xs:element name="name" type="xs:string"/> <xs:element name="dept name" type="xs:string"/> <xs:element name="salary" type="xs:decimal"/> </xs:sequence> </xs:complexType> </xs:element>	
3.	**(a) Ans.**	**Attempt any THREE of the following:** **Explain any four operation with Mongo DB with example.** The basic operations of Mongo DB are CRUD operations. Create, Read, Update & delete documents. • **Create operation**: The create or insert operations are used to add	**12** **4M**

		new document into the collection.
		If the collection is not exist, then the insert operation will create the collection.
		The different methods to insert document into a collection:
		- db.collection.insertone ( )
		- db.collection.insertmany ( )
		E.g:
		Db.student.insertOne
		(
		{
		Name:"Kunal"
		Age: "17"
		Status: "file pending"
		grade: "A"
		}
		)
		• **Read operations:** It is used to retrieve the documents from the collection.
		The find ( ) command is used to queries a collection for documents or simply to retrieve the documents from the collection.
		Syntax:
		Db.collection.find ( )
		E.g:
		db.student.find ( ) – for all documents
		or you can retrieve specific document with the help of attributes of that document.
		E.g:
		db.student.find(
		{
		name:"Kunal"
		}
		)
		• **Update operation:** This operation is used to modify the existing documents in a collection.
		Different methods are used for updation as
		-db.collection.updateone ( )
		-db.collection.updateMany ( )
		-db.collection.replaceOne ( )
		The Mongo DB uses the update operation for a single collection.
			Each operation with example 1M

		One can update the all documents without specifying any criteria.
		One you can update specific document by providing specific criteria.
		E.g: db.collaction.updateMany ( )
		E.g:
		db.student.updateOne (
		{
		Name:{$ SN: "Kunal"} ←update filter
		},
		{ $ set: { status: "completed"} ← update action.
		}
		)
		• **Delete operations:** It is used to remove documents from a collection.
		Different methods to delete documents are
		-db.collection.deleteOne ( )
		-db.collection.deleteMany ( )
		The delete operation is performed on a single collection.
		To delete the specific document you have to provide the specified criteria as per requirements.
		E.g:
		db.student.deleteMany ( ) or
		
		db.student.deleteOne(
		{
		Name:{$SN: "Kunal"}
		}
		)	

	(b) Ans.	**Explain and draw data warehouse life cycle.** The Data warehouse life cycle contains Project Planning Requirement gathering, Business Requirements, Design, ETL development, Project Management & Deployment. Data warehouse life-cycle is used to indicate the phases & their relationship through which the data warehouse system goes.	**4M**

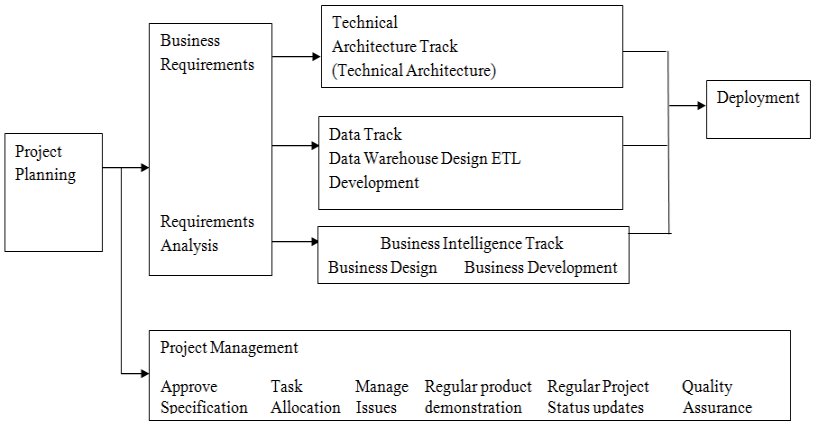


Diagram 1M

- Project Planning: Contains the requirement gathering & project management.
- Requirement gathering: It is done by business analyst, onsite technical lead & client.

The business Analyst prepares Business requirements specification (BRS) document.
80% of requirement collection takes place at client side.
The business requirement document can be prepared from the gathered requirement.

- Requirement Analysis: After collecting the requirements the requirement analysis. This is the very tough task as it affects every decision.

The user requirement analysis can following into 4 categories:

- Data driven
- User Driven
- Goal Driven
- Mixed Driven

- Technical Architecture Track: After requirement gathering & requirement analysis the technical architecture or the project design takes place.

This process involves preparing business requirement document into high level design that includes various modules in the data warehouse project.
This high level design is prepared by the architects.

Explanation 3M

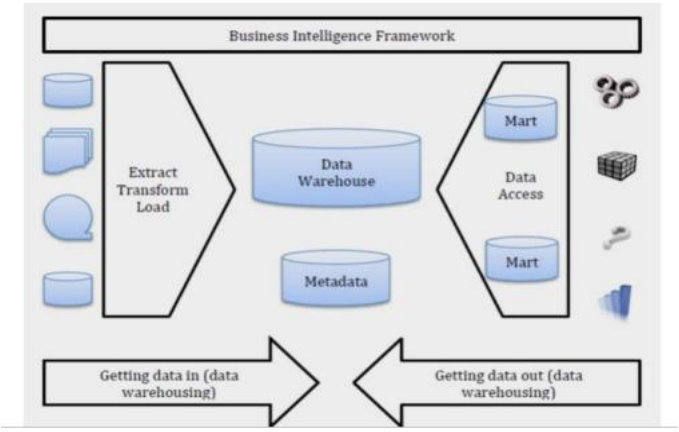
		• Data Track: The data track contains the data warehouse design & ETL development. Data ware design – is a process of designing the data base by fulfilling user requirements. A data modeler is responsible for creating Data Warehouse or Data Marts with different schemas as 1) Star schema: Simplest warehouse schema diagram resembles star. 2) Snowflake schema: Extention of star schema, adds additional dimensions, diagram resembles snowflake • ETL development: Designing ETL applications to fulfill the specifications of documents which are prepared in the analysis phase. The ETL development contains the ETL code review, Peer review and ETL testing. • Business Intelligence track: It contains BI design C BI development. The business logic is developed by the developers as per the requirement. • Deployment: It is the next phase after construction. The deployment phase concerns with training support and the maintenance of the product. This phase is also known as pilot phase or stabilization phase. • Project Management: The overall process of data warehouse life Cycle is managed by the project management It contains different phases as: Approve specification, Task allocation, Manage issues, Regular product demonstration, Regular product status updates and quality assurance. • Data Warehousing Development: Data warehouse is also known as enterprise data warehouse. It is a system used for reporting and data analysis. It is considered as the core component of business Intelligence.	
	(c) Ans.	**Describe BI components framework.** The Major Components of Business Intelligence (BI) The five primary components of BI include: **OLAP (Online Analytical Processing):** This component of BI allows executives to sort and select aggregates of data for strategic monitoring. With the help of specific software products, a certification in business intelligence helps business owners can use data to make adjustments to overall business processes.	**4M** ***Description 4M***

Advanced Analytics or Corporate Performance Management (CPM): This set of tools allows business leaders to look at the statistics of certain products or services. For instance, a fast food chain may analyze the sale of certain items and make local, regional and national modifications on menu board offerings as a result. The data could also be used to predict in which markets a new product may have the best success.

Real-time BI: Using software applications, a business can respond to real-time trends in email, messaging systems or even digital displays. Because it's all in real-time, an entrepreneur can announce special offers that take advantage of what's going on in the immediate.

Data Warehousing: Data warehousing lets business leaders sift through subsets of data and examine interrelated components that can help drive business. Looking at sales data over several years can help improve product development or tailor seasonal offerings.

Data Sources: This component of BI involves various forms of stored data. It's about taking the raw data and using software applications to create meaningful data sources that each division can use to positively impact business.



A Business Intelligence Framework is a framework that seamlessly connects the various elements of a business: organizational roles, KPIs (Key Performance Indicators), authorization, and visualization. This helps you implement Business Intelligence plans both easier and faster.

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Advanced Database Management Systems (Elective I) **Subject Code: 22521**

(d) Ans. **Differentiate between parallel & distributed databases.** — **4M**

Sr. No.	Parallel databases	distributed databases
1	It is a tightly coupled system because they are using shared memory.	It is loosly coupled system because it is using distributed memory.
2	Global clock control	No global clock control
3	The processor interconnection is in the order of terabyte (TB per second)	The processor interconnection is in the order of GB per seconds.
4	Main focus is on performance of system and scientific computing.	Main focus is on scalability, reliability and resource sharing.
5	In parallel database system machines are connects with dedicated high-speed LANs and switches.	In distributed database system machines can be connected using public-purpose network i.e, internet
6	Communication cost is very small	Communication cost is comparatively high
7	It has shared-memory, shared-disk, or shared-nothing architecture	It usually have shared- nothing architecture.

Any four differences 1M each

4. **Attempt any THREE of the following:** — **12**

(a) Ans. **Explain basic datatypes and arrays in MongoDB.** — **4M**

Basic Datatypes:

Mongo DB stores document on disk in the BSON serialization format. BSON is the binary representation of JSON documents. Hence BSON format provides more data types than JSON.

Data types supported by Mongo DB are:

- String: String in Mongo DB must be UTF-8 valid.
- Integer: Integer can be 32 bit or 64 bit depending upon server.
- Boolean: This type is used to store a Boolean values as True or false.
- Double: Used to store floating point values.
- Min/Max keys: This type is used to compare a value against the lowest and highest BSON elements.

Any 2 Basic data types with explanation 2M

		• Arrays: Used to store multiple values into one key.
		• Timestamp: Used for storing the time when a document has been modified or added.
		• Object: This datatype is used for embedded documents
		• Null: Used to store a null value
		• Symbol: Used to identically convert symbol into string. This datatype is not supported by the shell. If the shell gets a symbol from the database, it will convert it into a string.
		• Date: Used to store the current date or time in UNIX time format.
		• Object ID: This datatype is used to store the documents ID.
		• Binary data: Used to store binary data.
		• Code: This is used to store Javascript code into the document.
		• Regular expression: Used to store regular expression
		
		Arrays: Arrays are values which can be interchangeably referred for both ordered operating as lists, stack or queues or for unordered operations as sets.
		Arrays in Mongo DB are able to store different data types values.
		E.g:
		{
		"things": ["pi", 3.14]
		}
		Mongo DB enables atomic updates which helps to modify the contents of arrays.
			Array with exaplanation2M

	(b) Ans.	**Describe mobile databases and digital database.** **Mobile Database:** Mobile Database is a database that is transportable, portable and physically separate or detached from the corporate database server but has the capability to communicate with those servers from remote sites allowing the sharing of various kinds of data. With mobile databases, users have access to corporate data on their laptop, PDA, or other Internet access device that is required for applications at remote sites. The components of a mobile database environment include: • Corporate database server and DBMS that deals with and stores the corporate data and provides corporate applications • Remote database and DBMS usually manages and stores the mobile data and provides mobile applications • mobile database platform that includes a laptop, PDA, or other	**4M** ***Mobile database 2M***

		Internet access devices • Two-way communication links between the corporate and mobile DBMS. **Digital Database/Libraries:** Digital database/libraries are organized collections of information resources and associated tools for creating, archiving, sharing, searching, and using information that can be accessed electronically. Digital libraries differ from traditional libraries in that they exist in the "cyber world" of computers and the Internet A digital library is a collection of documents in organized electronic form, available on the Internet or on CD-ROM (compact-disk read-only memory) disks. Depending on the specific library, a user may be able to access magazine articles, books, papers, images, sound files, and videos. The digital library is most recent term being used for the recent term being used for the library without books, libraries having information in electronic format and providing access digital formats.	***Digital database 2M***
	(c) Ans.	**Explain the characteristics of Big data.** **List of characteristics of Big data:** 1. Volume 2. Velocity 3. Variety **1. Volume:** The amount of data matters. Organizations collect data from a variety of sources, including business transactions, social media and information from sensor or machine-to-machine data. In the past, storing large data would have been a problem – but new technologies (such as Hadoop) have eased the burden. With big data, you have to process high volumes of low-density, unstructured data. This can be data of unknown value, such as Twitter data feeds, click streams on a webpage or a mobile app, or sensor-enabled equipment. For some organizations, this might be tens of terabytes of data. For others, it may be hundreds of petabytes. **2. Velocity:** Velocity is the fast rate at which data is received and acted on. Normally, the highest velocity of data streams directly into memory versus being written to disk. Some internet-enabled smart products operate in real time or near real time and will require real-time evaluation and action.	**4M** ***List 1M*** ***Explanation 3M***

		3. Variety: Variety refers to the many types of data that are available. Traditional data types were structured and fit neatly in a relational database. With the rise of big data, data comes in new unstructured data types. Unstructured and semi structured data types, such as text, audio, and video,	
	(d) **Ans.**	**Explain the use of R-programming and also give the various applications where R-programming use.** **Use of R-programming:** R is a programming language and free software environment. It is used for statistical computing and graphics supported by the R foundation for statistical computing. The R language is widely used among statisticians and data minors for developing statistical software and data analysis. **Applications of R-Programming:** 1. Banking 2. Finance 3. E-commerce 4. Social-Media 5. Healthcare Most of the companies are using R: 1. Facebook: Facebook uses R to update facebook status updates and its social network graph. 2. Twitter: Basically, Twitter users R to monitor user experience. 3. New York Times: R is used by New York of advertising campaigns. 4. Google: Google uses R to calculate the ROI of advertising campaigns.	**4M** ***Explanation 2M*** ***Any two Applications 2M***
	(e) **Ans.**	**Explain table inheritance in SQL.** • **Table Inheritance:** Table inheritance is the property that allows a table to inheritance the state and behavior (Constraints, storage options, triggers) from the super table. • A table hierarchy is the relationship that can define among tables in which sub table inherits the behavior of super tables. The table inheritance provides issues as: 1) It encourages modular implementation of data model. 2) It ensures consistent reuse of schema components. 3) Allows to construct queries on some or all of the tables. Table inheritance inherits the properties:	**4M** ***Explanation 4M***

		1) Constraints 2) Storage option 3) All triggers 4) Indexes 5) Access method. Steps to inherit table : Table Hierarchy: The structure type must be define before creating the table of that type. The type hierarchy must be define before defining corresponding table hierarchy. Like a the type hierarchy the table hierarchy also use the under keyword. The CREATE TABLE statement is used to define the table. Consider we are already having the types as person_type, student_type, result_type. then we can create the table of this types. CREATE TABLE person of type person_type; CREATE TABLE student of type student_type under person; CREATE TABLE result of type result_type under student; The student table inherits properties from person table while result table inherits the properties from student table.	
5.	**(a) Ans.**	**Attempt any TWO of the following:** **Explain object identity and reference types in SQL with example.** Object identity provides the ability to refer to objects. An attribute of a type can be a reference to an object of a specified type. ***Eg:*** **Create type department(name varchar(20),head ref(person)scope people);** **Create table departments of department;** In above example reference is restricted to tuples of the table people. The restriction of the scope of reference to tuples of a table is mandatory in sql. **Create table departments of department(head with options scope people);** Reference table must have an attribute that stores the identifier of the tuple. The declare attribute was called self-reference attribute, by adding	**12** **6M** ***Explanation 3M*** ***Example 3M***

		a ref is clause to the create table statement. **Create table people of person ref is person_id system generated;** person_id is an attribute name, not a keyword,Create statement specifies that the identifier is generated automatically. System generated identifiers is to allow users to generate identifiers. The type of the self-referential attribute must be stated as part of the type definition of the referenced table and the table definition must specify that the reference is user generated ***Eg*** **Create type person(name varchar(20),address varchar(20)) ref using varchar(20);** **Create table people of person ref is person_id user generated;** References are dereferenced by the → symbol. For above created table departments **Select head →name , head → address from departments;**	
	(b) **Ans.**	**Explain the association rule in data mining. Explain application of association rule mining with example.** • Association rules are if-then statements that help to show the probability of relationships between data items within large data sets in various types of databases. • Association rule mining has a number of applications and is widely used to help discover sales correlations in transactional data or in medical data sets. • Association rule mining, at a basic level, involves the use of machine learning models to analyze data for patterns, or co-occurrence, in a database. • It identifies frequent if-then associations, which are called association rules. • An association rule has two parts: an antecedent (if) and a consequent (then). An antecedent is an item found within the data. A consequent is an item found in combination with the antecedent. **Application of association rule mining:** 1. Medical diagnosis. 2. Protein Sequences. 3. Fraud Detection in Credit Card Transactions.	**6M** ***Explanation 3M*** ***Any one***

		4. Bio-Medical Literature. 5. Customer Relationship Management (CRM). 6. Census Data etc. 7. Market Basket Analysis **1) Market Basket Analysis:** This is the most typical example of association mining. Data is collected using barcode scanners in most supermarkets. This database, known as the "market basket" database, consists of a large number of records on past transactions. A single record lists all the items bought by a customer in one sale. Knowing which groups are inclined towards which set of items gives these shops the freedom to adjust the store layout and the store catalog to place the optimally concerning one another. **2) Medical Diagnosis:** Association rules in medical diagnosis can be useful for assisting physicians for curing patients. Diagnosis is not an easy process and has a scope of errors which may result in unreliable end-results. Using relational association rule mining, we can identify the probability of the occurrence of illness concerning various factors and symptoms. Further, using learning techniques, this interface can be extended by adding new symptoms and defining relationships between the new signs and the corresponding diseases.	***application's explanation 3M***
	(c) Ans.	**Describe Hadoop. Explain architecture of Hadoop.** Hadoop is an open-source software framework for storing data and running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs. It is used to manage data, store data, and process data for various big data application running under clustered systems. Hadoop provides the following: **1) Ability to store and process huge amounts of any kind of data,** quickly. With data volumes and varieties constantly increasing, especially from social media and the Internet of Things (IoT), that's a key consideration. **2) Computing power:** Hadoop's distributed computing model processes big data fast. The more computing nodes you use the more processing power you have. **3) Fault tolerance:** Data and application processing are protected	**6M** ***Explanation 3M***

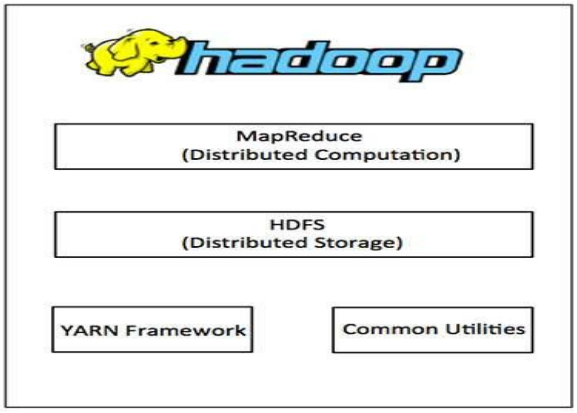
against hardware failure. If a node goes down, jobs are automatically redirected to other nodes to make sure the distributed computing does not fail. Multiple copies of all data are stored automatically.

4) Flexibility: Unlike traditional relational databases, you don't have to preprocess data before storing it. You can store as much data as you want and decide how to use it later. That includes unstructured data like text, images and videos.

5) Low cost: The open-source framework is free and uses commodity hardware to store large quantities of data.

Scalability. You can easily grow your system to handle more data simply by adding nodes. Little administration is required.

Architecture of Hadoop:



Architecture 3M

Hadoop has two major layers namely −

- Processing/Computation layer (MapReduce), and
- Storage layer (Hadoop Distributed File System).

MapReduce

MapReduce is a parallel programming model for writing distributed applications devised at Google for efficient processing of large amounts of data (multi-terabyte data-sets), on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. The MapReduce program runs on Hadoop which is an Apache open-source framework.

Hadoop Distributed File System

		The Hadoop Distributed File System (HDFS) is based on the Google File System (GFS) and provides a distributed file system that is designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. It is highly fault-tolerant and is designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications having large datasets. Apart from above mentioned two core components Hadoop framework also includes two modules as 1) Hadoop common utilities – These are the java libraries and utilities required by other Hadoop modules. 2) Hadoop YARN – This is a framework for job scheduling and cluster resource management.	
6.	**(a) Ans.**	**Attempt any TWO of the following:** **Explain concurrency control with locked based protocol.** In a multiprogramming environment where multiple transactions can be executed simultaneously, it is highly important to control the concurrency of transactions. We have concurrency control protocols to ensure atomicity, isolation, and serializability of concurrent transactions. Concurrency control protocols can be broadly divided into two categories − • Lock based protocols • Time stamp based protocols **Lock-based Protocols:** Database systems equipped with lock-based protocols use a mechanism by which any transaction cannot read or write data until it acquires an appropriate lock on it. Locks are of two kinds– • **Binary Locks**: A lock on a data item can be in two states; it is either locked or unlocked. • **Shared/exclusive:** This type of locking mechanism differentiates the locks based on their uses. If a lock is acquired on a data item to perform a write operation, it is an exclusive lock. Allowing more than one transaction to write on the same data item would lead the database into an inconsistent state. Read locks are shared because no data value is being changed. ***Example:***	**12** **6M** ***Explanation 4M***

		lock-S(A);
		
		read (A);
		
		unlock(A);
		
		lock-S(B);
		
		read (B);
		
		unlock(B);
		
		display(A+B)
			Example 2M

	(b) Ans.	**Explain flower expressions and nestead queries in Xquery.** The programming language XQuery defines **FLWOR** (pronounced 'flower') as an expression that supports iteration and binding of variables to intermediate results. **FLWOR** is an acronym: FOR, LET, WHERE, ORDER BY, RETURN. • **For** - selects a sequence of nodes • **Let** - binds a sequence to a variable • **Where** - filters the nodes • **Order by** - sorts the nodes • **Return** - what to return (gets evaluated once for every node) ***E:g:*** **for $x in doc("books.xml")/bookstore/book** **where $x/price>30** **order by $x/title** **return $x/title** The **for** clause selects all book elements under the bookstore element into a variable called $x. The **where** clause selects only book elements with a price element with a value greater than 30. The **order by** clause defines the sort-order. Will be sort by the title	**6M** ***Flower expression with Eg: 3M***

		element.
		
		The **return** clause specifies what should be returned. Here it returns the title elements.
		
		Nested xquery can be considered the multiple nested operation from the xquery. In nested queries, a query is written inside a query. The result of inner query is used in execution of outer query
		
		For each book in the bibliography, list the title and authors, grouped inside a result element.
		
		E:g:
		**<results>
		{
		for $b in doc("bib.xml")/bib/book
		return
		<result>
		{ $b/title }
		{
		for $a in $b/author
		return $a
		}
		</result>
		}
		</results>**
			Nested queries with Eg: 3M

	(c)	**Write query to excate find() function on collection: Inventory.** **(i) to display all document in collection** **(ii) to display all document where status equals "A" and qty is less than 40.** **(iii) to display all documents where status equals "A" and qty is less than 40 or item starts with character S.** ***(Note: Any other relevant form of query with correct logic shall be considered).***	**6M**
	Ans.	**(i) to display all document in collection:** Db.inventory.find()	

		(ii) to display all document where status equals "A" and qty is less than 40: Db.inventory.find({$and : [{"status": "A"} ,{"qty" : {$lt: 40}}]}) **(iii) to display all documents where status equals "A" and qty is less than 40 or item starts with character S:**Db.inventory.find({$and : [{"status": "A"},{$or:[{"qty":{$lt :40}},{item: {$regex: "S"}}]}]}) **OR** Db.inventory.find({$and : [{"status": "A"},{$or:[{"qty":{$lt :40}},{item: "/^S/"}}]}]})	***Each correct query 2M***