

Advanced DB – Unit -1

Introduction to databases and types of NoSQL databases:

1. What is a database schema?
 - A) A graphical representation of database tables.
 - B) A collection of database tables.
 - C) A formal description of the structure of a database.
 - D) A database query language.

2. Which of the following is a characteristic of a NoSQL database?
 - A) Strict schema requirements.
 - B) ACID transactions.
 - C) Horizontal scaling.
 - D) SQL-based querying.

3. Which type of NoSQL database is optimized for storing and retrieving key-value pairs?
 - A) Document store.
 - B) Column-family store.
 - C) Key-value store.
 - D) Graph store.

4. What is the main advantage of using a document store NoSQL database?
 - A) It supports complex relational queries.
 - B) It provides high availability and fault tolerance.
 - C) It allows storing unstructured data.
 - D) It stores data in a structured, tabular format.

5. Which NoSQL database type is best suited for applications that require processing large amounts of data with low latency?
 - A) Document store.
 - B) Column-family store.
 - C) Key-value store.
 - D) Graph store.

6. Which of the following is an example of a popular key-value store NoSQL database?
 - A) MongoDB.
 - B) Cassandra.
 - C) Redis.
 - D) Neo4j.

7. What does ACID stand for in the context of databases?
 - A) Atomicity, Consistency, Isolation, Durability.
 - B) Asynchronous, Consistent, Isolated, Durable.
 - C) Atomic, Consistent, Integrated, Durable.

- D) All of the above.

8. Which NoSQL database type is typically used for managing relationships and graph data structures?

- A) Document store.
- B) Column-family store.
- C) Key-value store.
- D) Graph store.

Key-Value Pair databases:

1. What is the fundamental data structure used in a Key-Value Pair database?

- A) Table.
- B) Document.
- C) Key-Value pair.
- D) Graph.

2. Which operation retrieves the value associated with a specific key in a Key-Value Pair database?

- A) `GET(key)`
- B) `SELECT(key)`
- C) `FIND(key)`
- D) `FETCH(key)`

3. In a Key-Value Pair database, what happens if you store a value with an existing key?

- A) It overwrites the existing value with the new one.
- B) It creates a new key-value pair.
- C) It throws an error.
- D) It updates the key metadata.

4. Which of the following is an example of a popular Key-Value Pair database?

- A) MongoDB.
- B) Cassandra.
- C) Redis.
- D) PostgreSQL.

5. What is the typical use case for a Key-Value Pair database?

- A) Complex relational queries.
- B) Storing and retrieving session data.
- C) Managing hierarchical data.
- D) Full-text search.

6. How does a Key-Value Pair database handle data retrieval?

- A) Through complex SQL queries.

- B) Using schema-based joins.
- C) By directly accessing values using keys.
- D) Through multi-document transactions.

7. What is a potential limitation of Key-Value Pair databases compared to relational databases?

- A) Scalability.
- B) ACID transactions.
- C) Support for complex queries.
- D) Schema flexibility.

8. Which characteristic makes Key-Value Pair databases suitable for high-performance applications?

- A) Strict schema enforcement.
- B) Strong consistency guarantees.
- C) Horizontal scalability.
- D) SQL-based querying.

Graph Databases:

1. What type of data structure do Graph Databases primarily use?

- A) Tables.
- B) Documents.
- C) Key-Value pairs.
- D) Graphs.

2. In a Graph Database, what are nodes?

- A) Tables that store data.
- B) Documents containing fields and values.
- C) Vertices that represent entities.
- D) Key-Value pairs.

3. What do edges represent in a Graph Database?

- A) Foreign keys.
- B) Indexes.
- C) Relationships between nodes.
- D) Documents.

4. Which operation is typically used to traverse a graph and retrieve related nodes and edges?

- A) `FIND()`
- B) `MATCH()`
- C) `SELECT()`
- D) `JOIN()`

5. Which type of database is optimized for queries involving complex relationships and graph structures?

- A) Document store.
- B) Column-family store.
- C) Key-Value store.
- D) Graph store.

6. Which language is commonly used for querying Graph Databases?

- A) SQL (Structured Query Language).
- B) MQL (MongoDB Query Language).
- C) CQL (Cassandra Query Language).
- D) Cypher.

7. What is a traversal in the context of Graph Databases?

- A) Searching for nodes based on a key.
- B) Navigating through relationships to discover related nodes.
- C) Modifying the structure of a graph.
- D) Exporting graph data to a file.

8. Which of the following is an example of a popular Graph Database?

- A) MongoDB.
- B) Cassandra.
- C) Redis.
- D) Neo4j.

Time-series Databases:

1. What is the primary characteristic of a Time-series Database?

- A) It stores data in tables.
- B) It uses graph structures.
- C) It is optimized for storing and querying time-stamped data.
- D) It supports complex relational queries.

2. Which of the following is a typical use case for a Time-series Database?

- A) Social networking profiles.
- B) Storing customer transactions.
- C) Monitoring IoT sensor data.
- D) Managing hierarchical data.

3. How does a Time-series Database optimize storage and retrieval of time-stamped data?

- A) By using tables with rows and columns.
- B) By compressing historical data.
- C) By supporting multi-document transactions.

- D) By enforcing strong schema constraints.
-
4. Which indexing technique is commonly used in Time-series Databases to improve query performance?
 - A) B-tree indexing.
 - B) Hash indexing.
 - C) Time-based indexing.
 - D) Graph-based indexing.

 5. Which of the following is a challenge often faced in managing Time-series data?
 - A) Storing unstructured data.
 - B) Ensuring ACID transactions.
 - C) Handling large volumes of data.
 - D) Supporting complex joins.

 6. Which feature is essential for efficient querying in Time-series Databases?
 - A) Full-text search.
 - B) Index-free adjacency.
 - C) Time-series data compression.
 - D) High availability and fault tolerance.

 7. What is a potential limitation of Time-series Databases compared to relational databases?
 - A) Horizontal scalability.
 - B) ACID transactions.
 - C) Support for complex queries.
 - D) Schema flexibility.

 8. Which of the following is an example of a popular Time-series Database?
 - A) MongoDB.
 - B) Cassandra.
 - C) InfluxDB.
 - D) Redis.

ADB Unit -2

MongoDB:

1. What type of database is MongoDB?
 - A) Relational database.
 - B) NoSQL database.
 - C) In-memory database.
 - D) Columnar database.

2. Which of the following best describes MongoDB's storage model?
 - A) Tables with rows and columns.

- B) Documents with fields and collections.
 - C) Key-value pairs.
 - D) Hierarchical graphs.
3. In MongoDB, what is a document?
- A) A record in a table.
 - B) A row in a relational database.
 - C) A JSON-like data structure composed of field-value pairs.
 - D) A collection of related tables.
4. Which of the following is a primary advantage of using MongoDB?
- A) Strong schema enforcement.
 - B) Vertical scaling.
 - C) ACID transactions.
 - D) Scalability and flexibility.
5. How does MongoDB handle relationships between data?
- A) Using foreign keys.
 - B) Embedding related documents within one another.
 - C) Creating separate collections for related data.
 - D) Storing relationships in separate tables.
6. What is the primary query language used in MongoDB?
- A) SQL (Structured Query Language).
 - B) JSON (JavaScript Object Notation).
 - C) MQL (MongoDB Query Language).
 - D) CQL (Cassandra Query Language).
7. Which of the following is a feature of MongoDB Atlas?
- A) On-premises deployment only.
 - B) Automatic scaling.
 - C) Only supports relational data.
 - D) Uses a fixed schema.
8. What is sharding in the context of MongoDB?
- A) A type of encryption for data security.
 - B) Horizontal partitioning of data across multiple machines.
 - C) Managing indexes for better query performance.
 - D) A backup strategy for disaster recovery.

MCQs on Document Data Model

1. What is a Document Data Model primarily used for?
- A) Structuring relational databases
 - B) Organizing hierarchical data
 - C) Managing unstructured data

- D) Enabling document-based databases
-
2. Which of the following databases is an example of a Document Data Model?
 - A) MySQL
 - B) MongoDB
 - C) PostgreSQL
 - D) Oracle

 3. In a Document Data Model, what format is commonly used to store documents?
 - A) CSV
 - B) XML
 - C) JSON
 - D) SQL

 4. What is a primary benefit of using a Document Data Model?
 - A) Complex transaction support
 - B) Schema flexibility
 - C) Strong ACID compliance
 - D) Enhanced data normalization

 5. Which of the following is a feature of a Document Data Model?
 - A) Fixed schema
 - B) Table-based storage
 - C) Indexing of documents
 - D) Row and column storage

 6. In the context of a Document Data Model, what does 'schema-less' mean?
 - A) No structure to the data
 - B) Fixed structure that can't be changed
 - C) Flexible structure that can change dynamically
 - D) No need for indexes

 7. Which language is commonly used for querying document-based databases?
 - A) SQL
 - B) SPARQL
 - C) NoSQL
 - D) Query Language (QL)

 8. How are relationships typically handled in a Document Data Model?
 - A) Foreign keys
 - B) Embedded documents or references
 - C) Join operations
 - D) Indexes

 9. Which of the following is a disadvantage of the Document Data Model?
 - A) Poor scalability
 - B) Rigid schema
 - C) Difficulty in handling hierarchical data

- D) Potential for data redundancy

10. In a Document Data Model, what is an 'embedded document'?

- A) A document that exists within another document
- B) A separate document with a foreign key reference
- C) A document that cannot be queried
- D) A document stored in a separate collection

MCQs on Documents and Collections in MongoDB

1. What is a document in MongoDB?

- A) A row in a table
- B) A set of SQL queries
- C) A record in a collection, represented in JSON-like format
- D) A file stored on the server

2. What is a collection in MongoDB?

- A) A single JSON document
- B) A table of rows and columns
- C) A group of documents
- D) A database schema

3. How does MongoDB store documents?

- A) In tables
- B) In arrays
- C) In collections
- D) In files

4. What format do documents in MongoDB use?

- A) CSV
- B) XML
- C) BSON (Binary JSON)
- D) Plain text

5. Which command is used to insert a document into a collection in MongoDB?

- A) INSERT INTO
- B) ADD DOCUMENT
- C) db.collection.insertOne()
- D) INSERT DOCUMENT

6. How can you retrieve all documents from a collection named "users"?

- A) db.users.findAll()

- B) `db.users.find()`
 - C) `SELECT * FROM users`
 - D) `db.users.getAll()`
7. What is the maximum size of a single document in MongoDB?
- A) 1 MB
 - B) 8 MB
 - C) 16 MB
 - D) 64 MB
8. Which of the following is true about collections in MongoDB?
- A) Collections must have a fixed schema
 - B) Collections can store documents of varying structures
 - C) Collections are stored in tables
 - D) Collections must have indexed documents
9. Which of the following commands deletes a single document from a collection?
- A) `db.collection.delete()`
 - B) `db.collection.removeOne()`
 - C) `db.collection.deleteOne()`
 - D) `db.collection.remove()`
10. How can you update a field in a document within a collection?
- A) `db.collection.modify()`
 - B) `db.collection.update()`
 - C) `db.collection.updateOne()`
 - D) `db.collection.change()`
11. Which of the following commands will create a new collection in MongoDB?
- A) `CREATE COLLECTION`
 - B) `db.createCollection()`
 - C) `db.newCollection()`
 - D) `db.makeCollection()`
12. What happens if you try to insert a document into a collection that doesn't exist?
- A) MongoDB will throw an error
 - B) MongoDB will create the collection and insert the document
 - C) MongoDB will ignore the operation
 - D) MongoDB will prompt for collection creation
13. Which of the following is used to uniquely identify a document in a collection?
- A) Primary Key
 - B) `_id` field
 - C) Document Key
 - D) Collection Key

14. What is the purpose of the ``$set`` operator in MongoDB?
- A) To insert a new document
 - B) To delete a document
 - C) To update specific fields in a document
 - D) To replace an entire document
15. How can you ensure a field is unique across all documents in a collection?
- A) Use a primary key
 - B) Use the ``$unique`` operator
 - C) Create a unique index on the field
 - D) Use the ``db.ensureUnique()`` method

MCQs on MongoDB Shell

1. What is the MongoDB shell used for?
- A) Running web applications
 - B) Interacting with MongoDB databases
 - C) Compiling JavaScript code
 - D) Managing server hardware
2. Which command is used to start the MongoDB shell?
- A) mongo
 - B) mongod
 - C) startmongo
 - D) mongoshell
3. How do you switch to a database named ``mydatabase`` in the MongoDB shell?
- A) use mydatabase
 - B) select mydatabase
 - C) switch mydatabase
 - D) `db.switch('mydatabase')`
4. Which command lists all collections in the current database?
- A) show databases
 - B) show collections
 - C) list collections
 - D) `db.collections()`
5. How do you display all documents in a collection named ``users``?
- A) `db.users.show()`
 - B) `db.users.find()`
 - C) `show db.users`
 - D) `list db.users`
6. What is the purpose of the ``db.stats()`` command in the MongoDB shell?
- A) To display statistics about the current database
 - B) To list all databases

- C) To show server status
- D) To display user roles

7. Which command is used to insert a document into a collection named `employees`?

- A) db.employees.add()
- B) db.employees.insert()
- C) db.employees.addDocument()
- D) db.employees.new()

8. How do you update a document in the `products` collection where the `name` is "Laptop"?

- A) db.products.modify({name: "Laptop"}, {\$set: {price: 1000}})
- B) db.products.updateOne({name: "Laptop"}, {\$set: {price: 1000}})
- C) db.products.change({name: "Laptop"}, {\$set: {price: 1000}})
- D) db.products.edit({name: "Laptop"}, {\$set: {price: 1000}})

9. Which command removes all documents from the `logs` collection?

- A) db.logs.deleteAll()
- B) db.logs.clear()
- C) db.logs.remove({})
- D) db.logs.drop()

10. How do you create an index on the `username` field in the `users` collection?

- A) db.users.createIndex({username: 1})
- B) db.users.addIndex({username: 1})
- C) db.users.index({username: 1})
- D) db.users.newIndex({username: 1})

11. Which command shows the current database you are working in?

- A) show database
- B) db
- C) current database
- D) show currentdb

12. How do you exit the MongoDB shell?

- A) quit
- B) exit
- C) quit()
- D) exit()

13. What does the `db.collection.findOne()` command do?
- A) Finds and returns all documents in a collection
 - B) Finds and returns the first document that matches the query
 - C) Finds and returns a single random document
 - D) Finds and returns the last document that matches the query
14. Which command will provide information about the indexes on a collection named `orders`?
- A) `db.orders.showIndexes()`
 - B) `db.orders.getIndexes()`
 - C) `db.orders.listIndexes()`
 - D) `db.orders.indexes()`
15. How do you view the structure of a document in the `customers` collection?
- A) `db.customers.describe()`
 - B) `db.customers.viewSchema()`
 - C) `db.customers.findOne()`
 - D) `db.customers.structure()`

ADB Unit – 3

MongoDB databases:

1. In MongoDB, what is a database?
- A) A collection of documents.
 - B) A collection of collections.
 - C) A JSON file containing data.
 - D) A structured table with rows and columns.
2. Which command is used to switch to a specific database in MongoDB shell?
- A) `USE <database_name>`
 - B) `SELECT DATABASE <database_name>`
 - C) `SWITCH TO <database_name>`
 - D) `DB <database_name>`
3. How does MongoDB organize data?
- A) In tables with rows and columns.
 - B) In collections of documents.
 - C) In key-value pairs.
 - D) In graph structures.
4. What is the default database in MongoDB?
- A) `admin`

- B) `local`
- C) `test`
- D) `default`

5. What happens when you insert a document into a collection that does not exist in MongoDB?

- A) MongoDB creates the collection and inserts the document.
- B) MongoDB throws an error.
- C) MongoDB inserts the document into the `system` collection.
- D) MongoDB ignores the document insertion.

6. Which of the following methods is used to drop a database in MongoDB?

- A) `DROP DATABASE <database\_name>`
- B) `DELETE DATABASE <database\_name>`
- C) `DB.<database\_name>.drop()`
- D) `REMOVE DATABASE <database\_name>`

7. In MongoDB, how are collections analogous to in a relational database?

- A) They are like rows.
- B) They are like fields.
- C) They are like tables.
- D) They are like databases.

8. What is the role of the `\_id` field in MongoDB?

- A) It is a mandatory field for defining relationships.
- B) It is automatically generated for each document and serves as a unique identifier.
- C) It specifies the database name.
- D) It denotes the primary key of the collection.

MCQs on Creating and Dropping Databases in MongoDB

1. Which command is used to create a new database in MongoDB?

- A) CREATE DATABASE dbName
- B) use dbName
- C) db.createDatabase(dbName)
- D) new database dbName

2. What happens if you run `use myNewDatabase` in the MongoDB shell and the database does not exist?

- A) An error is thrown
- B) The command is ignored
- C) A new database is created and switched to
- D) A prompt to create the database appears

3. How do you check the list of all databases in MongoDB?
- A) show collections
 - B) list databases
 - C) db.showDatabases()
 - D) show dbs
4. Which command drops the currently selected database?
- A) drop database
 - B) db.drop()
 - C) db.dropDatabase()
 - D) db.deleteDatabase()
5. Before dropping a database, which command should you run to ensure you are in the correct database?
- A) db.show()
 - B) show dbs
 - C) db.getName()
 - D) use dbName
6. What is the command to switch to a database named `testDB`?
- A) use database testDB
 - B) db.switch(testDB)
 - C) switch testDB
 - D) use testDB
7. If you want to drop a database named `inventory`, which command should you run after switching to that database?
- A) db.drop("inventory")
 - B) db.inventory.drop()
 - C) db.dropDatabase()
 - D) db.inventory.dropDatabase()
8. Which command verifies that a database has been successfully dropped?
- A) db.getName()
 - B) show databases
 - C) show collections
 - D) db.verifyDrop()
9. When using the `use` command to create a new database, when is the actual database file created on disk?
- A) Immediately after running `use dbName`
 - B) After inserting the first document into a collection
 - C) After creating the first collection
 - D) After running `db.createDatabase(dbName)`

10. How do you delete a collection named `products` from the `shopDB` database?
- A) `db.shopDB.products.drop()`
 - B) `use shopDB; db.products.drop()`
 - C) `use shopDB; drop collection products`
 - D) `db.products.removeAll()`
11. If you have switched to a new database using the `use` command but have not yet created any collections, what will the `show dbs` command display?
- A) It will show the new database
 - B) It will not show the new database
 - C) It will throw an error
 - D) It will show an empty database list
12. What is the command to rename a database in MongoDB?
- A) `rename database oldName to newName`
 - B) `db.renameDatabase(oldName, newName)`
 - C) MongoDB does not support renaming databases directly
 - D) `db.getSiblingDB('oldName').rename('newName')`
13. How do you check the size of the current database?
- A) `db.size()`
 - B) `db.stats().dataSize`
 - C) `db.getSize()`
 - D) `db.dataSize()`
14. Which command is used to drop all the collections in the current database?
- A) `db.dropCollections()`
 - B) `db.clear()`
 - C) `db.dropAll()`
 - D) `db.getCollectionNames().forEach(function(c) { db[c].drop(); })`
15. What should you do if you need to safely back up a database before dropping it?
- A) Use the `db.backup()` command
 - B) Use `mongodump` to create a backup
 - C) Copy the database files manually
 - D) Use `db.export()`

MCQs on Adding Batch of Documents, Finding Documents, and Working with Documents in MongoDB

1. Which command is used to insert multiple documents into a collection in MongoDB?
- A) `db.collection.insertMany()`
 - B) `db.collection.insertBatch()`
 - C) `db.collection.addMany()`
 - D) `db.collection.insert()`

2. How do you find all documents in a collection named `employees`?

- A) db.employees.findAll()
- B) db.employees.getAll()
- C) db.employees.find({})
- D) db.employees.select()

3. Which operator is used to match documents where a field is greater than a specified value?

- A) \$lt
- B) \$gte
- C) \$gt
- D) \$lte

4. How do you update multiple documents in a collection named `products`?

- A) db.products.updateAll()
- B) db.products.updateMany()
- C) db.products.modifyAll()
- D) db.products.changeMany()

5. Which command removes a single document that matches a given filter from the `orders` collection?

- A) db.orders.removeOne()
- B) db.orders.deleteOne()
- C) db.orders.dropOne()
- D) db.orders.remove({})

6. How do you retrieve only the `name` and `age` fields from documents in the `users` collection?

- A) db.users.find({}, {name: 1, age: 1})
- B) db.users.find({name: 1, age: 1})
- C) db.users.select({name: 1, age: 1})
- D) db.users.get({name: 1, age: 1})

7. Which method allows you to add a new field to all documents in a collection named `inventory`?

- A) db.inventory.addField()
- B) db.inventory.set()
- C) db.inventory.updateMany({}, {\$set: {newField: "value"}})
- D) db.inventory.modifyAll({}, {newField: "value"})

8. How can you find documents in the `students` collection where the `grade` field is either "A" or "B"?

- A) db.students.find({grade: {\$in: ["A", "B"]}})
- B) db.students.find({grade: ["A", "B"]})
- C) db.students.find({grade: {\$or: ["A", "B"]}})
- D) db.students.find({grade: {\$any: ["A", "B"]}})

9. Which command do you use to insert a single document into the `customers` collection?

- A) db.customers.addOne()
- B) db.customers.insertOne()
- C) db.customers.addDocument()
- D) db.customers.newDocument()

10. How do you replace an entire document in the `books` collection?

- A) db.books.updateOne()
- B) db.books.modifyOne()
- C) db.books.replaceOne()
- D) db.books.changeOne()

11. Which method would you use to update a specific field in documents that match a filter in the `sales` collection?

- A) db.sales.modifyMany()
- B) db.sales.set()
- C) db.sales.updateField()
- D) db.sales.updateMany()

12. How do you sort the results of a query on the `events` collection by the `date` field in descending order?

- A) db.events.find().sort({date: -1})
- B) db.events.find().orderBy({date: -1})
- C) db.events.sortBy({date: -1})
- D) db.events.find().order({date: -1})

13. Which command finds the first document in the `logs` collection that matches the query?

- A) db.logs.findOne()
- B) db.logs.findFirst()
- C) db.logs.getFirst()
- D) db.logs.find().limit(1)

14. How do you remove all documents from the `archive` collection that match a specific filter?

- A) db.archive.deleteAll()
- B) db.archive.remove({})
- C) db.archive.deleteMany()
- D) db.archive.clear()

15. How do you add a batch of documents to a collection named `transactions`?

- A) db.transactions.insertMany([{...}, {...}, {...}])
- B) db.transactions.addBatch([{...}, {...}, {...}])
- C) db.transactions.insertBatch([{...}, {...}, {...}])

- D) db.transactions.addMany([{...}, {...}, {...}])

MCQs on Modifying Documents, Finding Selected Fields, and Removing Documents in MongoDB

1. Which command is used to update a single document in MongoDB?
 - A) db.collection.updateOne()
 - B) db.collection.modifyOne()
 - C) db.collection.changeOne()
 - D) db.collection.editOne()
2. How do you update multiple documents in a collection named `products` where the `category` is "electronics"?
 - A) db.products.updateMany({category: "electronics"}, {\$set: {status: "available"}})
 - B) db.products.modifyAll({category: "electronics"}, {\$set: {status: "available"}})
 - C) db.products.update({category: "electronics"}, {\$set: {status: "available"}})
 - D) db.products.updateDocuments({category: "electronics"}, {\$set: {status: "available"}})
3. Which operator is used to set a new value to a field in a document?
 - A) \$push - B) \$set - C) \$add - D) \$inc
4. How do you increment the value of a field `count` by 1 in a document within the `visits` collection?
 - A) db.visits.updateOne({}, {\$set: {count: 1}})
 - B) db.visits.modify({}, {\$inc: {count: 1}})
 - C) db.visits.updateOne({}, {\$inc: {count: 1}})
 - D) db.visits.change({}, {\$increment: {count: 1}})
5. Which command removes a single document that matches a filter in the `orders` collection?
 - A) db.orders.removeOne()
 - B) db.orders.deleteOne()
 - C) db.orders.dropOne()
 - D) db.orders.remove({})
6. How do you remove all documents from a collection named `logs`?
 - A) db.logs.deleteAll()
 - B) db.logs.remove({})
 - C) db.logs.deleteMany({})
 - D) db.logs.clear()
7. How do you retrieve only the `name` and `email` fields from documents in the `users` collection?
 - A) db.users.find({}, {name: 1, email: 1})
 - B) db.users.find({name: 1, email: 1})
 - C) db.users.select({name: 1, email: 1})

- D) db.users.get({name: 1, email: 1})
8. Which command adds a new field to all documents in a collection named `inventory`?
- A) db.inventory.addField()
 - B) db.inventory.set()
 - C) db.inventory.updateMany({}, {\$set: {newField: "value"}})
 - D) db.inventory.modifyAll({}, {newField: "value"})
9. What command would you use to find documents in the `students` collection where the `grade` is "A" or "B" and only retrieve the `name` and `grade` fields?
- A) db.students.find({grade: {\$in: ["A", "B"]}}, {name: 1, grade: 1})
 - B) db.students.find({grade: ["A", "B"]}, {name: 1, grade: 1})
 - C) db.students.find({grade: {\$or: ["A", "B"]}}, {name: 1, grade: 1})
 - D) db.students.find({grade: {\$any: ["A", "B"]}}, {name: 1, grade: 1})
10. Which method would you use to update a specific field in documents that match a filter in the `sales` collection?
- A) db.sales.modifyMany()
 - B) db.sales.set()
 - C) db.sales.updateField()
 - D) db.sales.updateMany()
11. How do you sort the results of a query on the `events` collection by the `date` field in descending order?
- A) db.events.find().sort({date: -1})
 - B) db.events.find().orderBy({date: -1})
 - C) db.events.sortBy({date: -1})
 - D) db.events.find().order({date: -1})
12. Which command finds the first document in the `logs` collection that matches the query?
- A) db.logs.findOne()
 - B) db.logs.findFirst()
 - C) db.logs.getFirst()
 - D) db.logs.find().limit(1)
13. How do you remove a single document from the `archive` collection that matches a specific filter?
- A) db.archive.deleteOne()
 - B) db.archive.remove({})
 - C) db.archive.deleteMany({})
 - D) db.archive.clear()
14. Which method would you use to update a field in all documents in the `employees` collection where the `department` is "HR"?

- A) `db.employees.updateMany({department: "HR"}, {$set: {status: "active"}})`
- B) `db.employees.modify({department: "HR"}, {$set: {status: "active"}})`
- C) `db.employees.update({department: "HR"}, {$set: {status: "active"}})`
- D) `db.employees.updateDocuments({department: "HR"}, {$set: {status: "active"}})`

15. How do you add a new field `createdAt` with the current date to all documents in the `users` collection?

- A) `db.users.updateMany({}, {$set: {createdAt: new Date()}})`
- B) `db.users.addField({}, {createdAt: new Date()})`
- C) `db.users.modify({}, {$set: {createdAt: new Date()}})`
- D) `db.users.update({}, {createdAt: new Date()})`

MCQs on Indexing in MongoDB

1. What is the primary purpose of indexing in MongoDB?

- A) To reduce storage requirements
- B) To increase the performance of search queries
- C) To enable distributed transactions
- D) To provide data encryption

2. Which command is used to create an index on the `username` field in the `users` collection?

- A) `db.users.addIndex({username: 1})`
- B) `db.users.index({username: 1})`
- C) `db.users.createIndex({username: 1})`
- D) `db.users.newIndex({username: 1})`

3. How do you create a compound index on the `firstName` and `lastName` fields in the `employees` collection?

- A) `db.employees.createCompoundIndex({firstName: 1, lastName: 1})`
- B) `db.employees.createIndex({firstName: 1, lastName: 1})`
- C) `db.employees.addIndex({firstName: 1, lastName: 1})`
- D) `db.employees.index({firstName: 1, lastName: 1})`

4. Which of the following is the correct command to drop an index on the `email` field in the `customers` collection?

- A) `db.customers.dropIndex({email: 1})`
- B) `db.customers.removeIndex({email: 1})`
- C) `db.customers.deleteIndex({email: 1})`
- D) `db.customers.indexDrop({email: 1})`

5. What does the `db.collection.getIndexes()` command do?

- A) It lists all the documents in a collection
- B) It lists all the indexes on a collection
- C) It deletes all indexes on a collection
- D) It updates all indexes on a collection

6. How do you create a unique index on the `email` field in the `users` collection?
- A) `db.users.createUniqueIndex({email: 1})`
 - B) `db.users.createIndex({email: 1}, {unique: true})`
 - C) `db.users.addUniqueIndex({email: 1})`
 - D) `db.users.index({email: 1}, {unique: true})`
7. Which command creates a text index on the `description` field in the `products` collection?
- A) `db.products.createTextIndex({description: "text"})`
 - B) `db.products.createIndex({description: "text"})`
 - C) `db.products.textIndex({description: "text"})`
 - D) `db.products.addTextIndex({description: "text"})`
8. How do you find documents in the `articles` collection that contain the word "MongoDB" in a text indexed field?
- A) `db.articles.find({$text: {$search: "MongoDB"}})`
 - B) `db.articles.find({text: "MongoDB"})`
 - C) `db.articles.find({description: "MongoDB"})`
 - D) `db.articles.search({$text: {$query: "MongoDB"}})`
9. What is the purpose of a hashed index in MongoDB?
- A) To provide full-text search capabilities
 - B) To ensure unique values in a field
 - C) To distribute documents more evenly across a sharded cluster
 - D) To compress the indexed data
10. Which command creates an index on the `orderDate` field in the `orders` collection with an expiration time of 3600 seconds?
- A) `db.orders.createTTLIndex({orderDate: 1}, {expireAfterSeconds: 3600})`
 - B) `db.orders.createIndex({orderDate: 1}, {expireAfterSeconds: 3600})`
 - C) `db.orders.addTTLIndex({orderDate: 1}, {expireAfterSeconds: 3600})`
 - D) `db.orders.index({orderDate: 1}, {expireAfterSeconds: 3600})`
11. How do you create a sparse index on the `phoneNumber` field in the `contacts` collection?
- A) `db.contacts.createIndex({phoneNumber: 1}, {sparse: true})`
 - B) `db.contacts.addSparseIndex({phoneNumber: 1})`
 - C) `db.contacts.createSparseIndex({phoneNumber: 1})`
 - D) `db.contacts.index({phoneNumber: 1}, {sparse: true})`
12. Which command drops all indexes on the `employees` collection except for the `\_id` index?
- A) `db.employees.dropAllIndexes()`
 - B) `db.employees.dropIndexes()`

- C) db.employees.removeAllIndexes()
 - D) db.employees.clearIndexes()
13. How do you create a 2dsphere index on the `location` field in the `places` collection?
- A) db.places.create2dsphereIndex({location: "2dsphere"})
 - B) db.places.createIndex({location: "2dsphere"})
 - C) db.places.add2dsphereIndex({location: "2dsphere"})
 - D) db.places.index({location: "2dsphere"})
14. Which command checks the usage statistics of indexes in the `customers` collection?
- A) db.customers.indexStats()
 - B) db.customers.indexUsageStats()
 - C) db.customers.aggregate([{\$indexStats: {}}])
 - D) db.customers.statsIndex()
15. What type of index should you create to ensure that the combination of `firstName` and `lastName` fields is unique in the `users` collection?
- A) Compound index with unique option
 - B) Single field unique index
 - C) Text index
 - D) Hashed index

MCQs on Writing to Shards in MongoDB

1. What is the primary purpose of sharding in MongoDB?
- A) To backup data
 - B) To distribute data across multiple servers
 - C) To create indexes
 - D) To increase query performance without data distribution
2. Which of the following commands enables sharding for a database named `testDB`?
- A) sh.enableSharding("testDB")
 - B) db.enableSharding("testDB")
 - C) db.adminCommand({enableSharding: "testDB"})
 - D) sh.shardDatabase("testDB")
3. How do you shard a collection named `users` in the `testDB` database using the `userID` field as the shard key?
- A) sh.shardCollection("testDB.users", {userID: 1})
 - B) sh.shard("testDB.users", {userID: 1})
 - C) sh.enableSharding("testDB.users", {userID: 1})
 - D) sh.addShardKey("testDB.users", {userID: 1})
4. What is a shard key in MongoDB?
- A) A field used to create an index
 - B) A unique identifier for each document

- C) A field or combination of fields used to distribute documents across shards
 - D) A field used to encrypt documents
5. Which command moves a chunk of data to a different shard in MongoDB?
- A) `sh.moveChunk()`
 - B) `sh.relocateChunk()`
 - C) `sh.splitChunk()`
 - D) `sh.moveData()`
6. How does MongoDB determine the shard to which a document should be written?
- A) Based on the document's size
 - B) Based on the shard key value
 - C) Randomly
 - D) Based on the document's creation time
7. What happens if you try to write a document with a shard key value that falls outside the current range of chunks?
- A) The write operation fails
 - B) MongoDB automatically creates a new chunk
 - C) MongoDB moves an existing chunk
 - D) The document is written to a temporary collection
8. How can you check the shard distribution of a collection?
- A) `db.collection.stats()`
 - B) `db.collection.shardStats()`
 - C) `sh.status()`
 - D) `db.collection.distStats()`
9. Which of the following operations can be used to split an existing chunk in MongoDB?
- A) `sh.splitAt()`
 - B) `sh.splitChunk()`
 - C) `sh.divideChunk()`
 - D) `sh.createChunk()`
10. How do you ensure high availability and data redundancy in a sharded MongoDB cluster?
- A) By using indexes
 - B) By using replica sets for each shard
 - C) By using a single server for each shard
 - D) By using a primary key
11. What command would you use to add a new shard to a MongoDB cluster?
- A) `sh.addShard()`
 - B) `sh.addShardServer()`
 - C) `sh.addServer()`
 - D) `sh.addNode()`

12. Which component in a MongoDB sharded cluster is responsible for routing read and write operations to the appropriate shard?

- A) Config server
- B) Shard server
- C) Query router (mongos)
- D) Replica set

13. How do you balance the distribution of chunks across shards in a MongoDB sharded cluster?

- A) sh.balanceChunks()
- B) sh.startBalancer()
- C) sh.balanceData()
- D) sh.rebalance()

14. What does the `sh.status()` command display?

- A) The status of the MongoDB server
- B) The status of the shards in the cluster
- C) The status of the balancer
- D) Detailed information about the sharded cluster

15. Which of the following is true about the shard key?

- A) It must be unique for each document in the collection
- B) It can be changed after being set
- C) It determines how data is distributed across the shards
- D) It cannot be an indexed field

ADB Unit – 4

Cassandra:

1. Cassandra is best categorized as which type of database?

- A) Relational database.
- B) NoSQL database.
- C) In-memory database.
- D) Key-value store.

2. Which of the following is a characteristic of Cassandra?

- A) Strong consistency.
- B) SQL-based querying.
- C) Horizontal scalability.
- D) Fixed schema.

3. How does Cassandra achieve fault tolerance?

- A) Through vertical scaling.

- B) By using ACID transactions.
 - C) With data replication across multiple nodes.
 - D) By enforcing strict schema constraints.
4. In Cassandra, data is organized into which primary structure?
- A) Tables.
 - B) Documents.
 - C) Columns.
 - D) Key-value pairs.
5. Which consistency level in Cassandra offers the highest consistency but may impact availability during network partitions?
- A) ONE.
 - B) QUORUM.
 - C) ALL.
 - D) LOCAL_ONE.
6. Cassandra uses which protocol for communication between nodes?
- A) HTTP.
 - B) TCP/IP.
 - C) REST.
 - D) UDP.
7. What is a partition key in Cassandra?
- A) The primary key of a table.
 - B) A unique identifier for each node.
 - C) A column used for data distribution and replication.
 - D) An index for fast lookup.
8. Which of the following is a typical use case for Cassandra?
- A) Online transaction processing (OLTP).
 - B) Complex relational queries.
 - C) Data warehousing.
 - D) High-volume, low-latency applications.

MCQs on The Column-Family Data Model in Cassandra

1. What is a keyspace in Cassandra?
- A) A single row in a table
 - B) A collection of rows in a table
 - C) A namespace for a collection of tables
 - D) A single column in a table
2. Which of the following best describes a column family in Cassandra?
- A) A collection of tables
 - B) A collection of keyspaces
 - C) A collection of rows that share the same primary key
 - D) A collection of rows with a similar schema

3. What is the primary key in a Cassandra table used for?
 - A) To uniquely identify a row within a table
 - B) To encrypt data
 - C) To define the structure of the table
 - D) To create indexes
4. In Cassandra, what is a partition key?
 - A) The first part of the primary key, used to determine which node stores the data
 - B) The second part of the primary key, used for clustering
 - C) A key used to index data
 - D) A key used to replicate data
5. Which command is used to create a new keyspace in Cassandra?
 - A) CREATE TABLE
 - B) CREATE KEYSPACE
 - C) CREATE COLUMNFAMILY
 - D) CREATE SCHEMA
6. Which of the following statements is true about clustering columns in Cassandra?
 - A) They determine how data is partitioned across nodes
 - B) They determine the order of data within a partition
 - C) They must be unique for each row
 - D) They are optional and not used in most cases
7. How do you define a composite key in a Cassandra table?
 - A) By specifying multiple primary keys
 - B) By combining partition key and clustering columns
 - C) By using a single column as the key
 - D) By creating multiple indexes
8. Which of the following is used to define the replication strategy of a keyspace in Cassandra?
 - A) CREATE TABLE
 - B) CREATE KEYSPACE
 - C) CREATE REPLICATION
 - D) CREATE STRATEGY
9. What is the purpose of a "super column" in Cassandra?
 - A) To serve as a primary key
 - B) To group related columns together within a column family
 - C) To index data for faster queries
 - D) To replicate data across nodes
10. Which CQL (Cassandra Query Language) command is used to insert data into a table?
 - A) INSERT INTO
 - B) ADD TO
 - C) PUT INTO
 - D) APPEND TO
11. How do you ensure high availability and fault tolerance in a Cassandra cluster?
 - A) By creating secondary indexes

- B) By using replication strategies
 - C) By increasing the number of nodes
 - D) By partitioning data evenly
12. What is the purpose of a "wide row" in Cassandra?
- A) To store large binary data
 - B) To accommodate a large number of columns for a single row key
 - C) To ensure data consistency
 - D) To distribute data across multiple keyspaces
13. Which of the following best describes a time-series data model in Cassandra?
- A) A model where data is partitioned by time intervals
 - B) A model where each row represents a specific time point
 - C) A model where data is stored as JSON objects
 - D) A model where data is indexed by time
14. What is the purpose of the `WITH CLUSTERING ORDER BY` clause in a CREATE TABLE statement?
- A) To specify the replication strategy
 - B) To define the primary key
 - C) To order rows within a partition
 - D) To set the consistency level
15. How do you delete a column family in Cassandra?
- A) DROP COLUMNFAMILY
 - B) DROP KEYSPACE
 - C) DROP TABLE
 - D) DELETE COLUMNFAMILY

MCQs on Cassandra's Architecture

1. What is the role of a node in a Cassandra cluster?
- A) To act as a master server for managing other nodes
 - B) To store and manage data
 - C) To replicate data across multiple clusters
 - D) To act as a load balancer for incoming requests
2. In Cassandra, what is a "data center"?
- A) A physical location where nodes are housed
 - B) A logical grouping of nodes for replication and fault tolerance
 - C) A collection of tables
 - D) A set of user-defined keyspaces
3. What is the purpose of the gossip protocol in Cassandra?
- A) To synchronize clocks across nodes
 - B) To exchange information about node status and cluster membership
 - C) To perform data backups
 - D) To manage user permissions
4. How does Cassandra ensure data consistency across multiple nodes?

- A) By using a distributed lock mechanism
- B) By replicating data across nodes and using a quorum-based approach
- C) By enforcing a master-slave architecture
- D) By encrypting data at rest

5. What is a "commit log" in Cassandra?

- A) A log that records all read operations
- B) A log that records all write operations to ensure durability
- C) A log that tracks schema changes
- D) A log that manages user authentication

Ans: B) A log that records all write operations to ensure durability

6. Which component in Cassandra is responsible for handling read and write requests?

- A) Coordinator node
- B) Data center
- C) Seed node
- D) Storage node

7. What is a "seed node" in a Cassandra cluster?

- A) A node used to start up and join a cluster
- B) A node that stores backup data
- C) A node responsible for balancing data across other nodes
- D) A node that performs encryption and decryption

8. What does the term "hinted handoff" refer to in Cassandra?

- A) A process for rebalancing data across nodes
- B) A mechanism to handle failed writes by temporarily storing hints and later replaying them
- C) A method for encrypting data in transit
- D) A technique for synchronizing clocks across nodes

9. In Cassandra, what is a "replica" and how is it managed?

- A) A copy of data stored on a single node; managed by the coordinator node
- B) A copy of data distributed across multiple nodes; managed by the replication strategy
- C) A backup copy of data stored on a disk; managed by the commit log
- D) A copy of data used for encryption; managed by the security manager

10. Which component in Cassandra is responsible for performing data compaction?

- A) The commit log
- B) The Memtable
- C) The SSTable
- D) The Compaction Manager

11. What does the term "eventual consistency" mean in Cassandra?

- A) All nodes must be consistent immediately after a write operation
- B) Data will eventually become consistent across nodes, but there may be temporary inconsistencies
- C) Consistency is enforced using a master-slave architecture
- D) Data is always consistent due to synchronous replication

12. Which of the following best describes Cassandra's write path architecture?
- A) Writes go directly to the commit log, then to Memtables and SSTables
 - B) Writes are first logged in a backup file, then committed to the database
 - C) Writes are handled by a master node and replicated to slave nodes
 - D) Writes are directly applied to disk and then to the cache
13. What is the role of "Memtables" in Cassandra?
- A) To cache data temporarily before it is written to disk
 - B) To manage replication and data distribution
 - C) To handle read and write requests
 - D) To perform data encryption
14. How does Cassandra handle large amounts of data distributed across multiple nodes?
- A) By using a master-slave architecture
 - B) By partitioning data and distributing it across nodes using a consistent hashing algorithm
 - C) By storing all data on a single node and replicating it to other nodes
 - D) By creating multiple databases for different data types
15. What is the function of the "Cassandra Query Language (CQL)"?
- A) To manage cluster configuration
 - B) To perform administrative tasks on nodes
 - C) To interact with and query data in Cassandra
 - D) To handle data replication and consistency

ADB Unit – 5

MongoDB with Python:

1. Which Python library is commonly used to interact with MongoDB?
- A) SQLAlchemy.
 - B) pymongo.
 - C) pyMongoDB.
 - D) mongoDBConnector.
2. How can you connect to a MongoDB server using pymongo in Python?
- A) ``connect(server='localhost', port=27017)``
 - B) ``MongoClient('localhost', 27017)``
 - C) ``mongodb://localhost:27017``
 - D) ``pymongo.connect('localhost:27017')``
3. In pymongo, how do you specify a database to work with?
- A) ``use_database('mydb')``
 - B) ``db = client.mydb``
 - C) ``connect.database('mydb')``
 - D) ``client.connect('mydb')``

4. What MongoDB operation in pymongo allows you to insert one document into a collection?

- A) ``add_one()``
- B) ``insert_one()``
- C) ``add_document()``
- D) ``insert_document()``

5. How do you find documents in a collection that match a specific condition using pymongo?

- A) ``collection.find(condition)``
- B) ``find_document(condition)``
- C) ``collection.fetch(condition)``
- D) ``collection.find_one(condition)``

6. What is the purpose of the ``find_one()`` method in pymongo?

- A) It retrieves the first document in a collection.
- B) It finds all documents in a collection.
- C) It updates a single document in a collection.
- D) It deletes a single document from a collection.

7. How can you update a document in a collection using pymongo?

- A) ``collection.update(document)``
- B) ``update_one(filter, update)``
- C) ``modify_document(filter, update)``
- D) ``update_document(filter, update)``

8. Which pymongo method is used to delete a document from a collection?

- A) ``delete(document)``
- B) ``remove_one(filter)``
- C) ``delete_one(filter)``
- D) ``delete_document(filter)``

Python modules:

1. What is a Python module?

- A) A directory containing Python files.
- B) A built-in function in Python.
- C) A Python file containing functions, classes, and variables.
- D) A Python script used for automation.

2. How do you import a module named ``mymodule`` in Python?

- A) ``import mymodule``
- B) ``use mymodule``
- C) ``require mymodule``
- D) ``from mymodule import *``

3. Which statement is true about the ``from ... import ...`` syntax in Python?

- A) It imports all names from the module into the current namespace.
- B) It imports only specific names from the module into the current namespace.
- C) It imports a specific function from a module.
- D) It imports a module and renames it in the current namespace.

4. What does the `if __name__ == "__main__":` statement in a Python module indicate?
 - A) It checks if the module is being executed as a standalone program.
 - B) It defines a function that is only executed when imported as a module.
 - C) It indicates the end of the module's execution.
 - D) It checks if the module has finished executing.
5. How can you alias a module `mymodule` to `mm` when importing it in Python?
 - A) `import mymodule as mm`
 - B) `use mymodule as mm`
 - C) `require mymodule as mm`
 - D) `from mymodule import as mm`
6. Which Python module is used for interacting with the operating system's file system?
 - A) `os`
 - B) `sys`
 - C) `file`
 - D) `filesystem`
7. What is the purpose of the `__init__.py` file in a Python module directory?
 - A) It contains initialization code for the module.
 - B) It indicates that the directory is a Python package.
 - C) It specifies the entry point of the module.
 - D) It is required for importing modules.
8. How do you uninstall a third-party module `mymodule` installed via `pip` in Python?
 - A) `pip uninstall mymodule`
 - B) `remove mymodule`
 - C) `delete mymodule`
 - D) `uninstall mymodule`

MCQs on Working with Databases with PyMongo

1. What is PyMongo?
 - A) A Python library for MongoDB
 - B) A tool for managing MongoDB clusters
 - C) A command-line interface for MongoDB
 - D) A MongoDB GUI tool
2. Which function is used to connect to a MongoDB server using PyMongo?
 - A) `MongoClient()`
 - B) `connect()`
 - C) `open_connection()`
 - D) `db.connect()`
3. How do you access a specific database named `mydatabase` using PyMongo?
 - A) `client.use('mydatabase')`
 - B) `client.get_database('mydatabase')`
 - C) `client['mydatabase']`
 - D) `client.database('mydatabase')`

4. How do you create a new collection named `mycollection` in a database using PyMongo?
- A) `db.create_collection('mycollection')`
 - B) `db.new_collection('mycollection')`
 - C) `db.add_collection('mycollection')`
 - D) `db.mycollection.create()`
5. Which method is used to insert a single document into a collection using PyMongo?
- A) `collection.add_one(document)`
 - B) `collection.insert(document)`
 - C) `collection.insert_one(document)`
 - D) `collection.save(document)`
6. How do you find all documents in a collection using PyMongo?
- A) `collection.find_all()`
 - B) `collection.query()`
 - C) `collection.find()`
 - D) `collection.get_all()`
7. Which method is used to update a single document in a collection?
- A) `collection.update_one(filter, update)`
 - B) `collection.modify_one(filter, update)`
 - C) `collection.change_one(filter, update)`
 - D) `collection.set_one(filter, update)`
8. How do you delete a single document from a collection using PyMongo?
- A) `collection.remove_one(filter)`
 - B) `collection.delete(filter)`
 - C) `collection.delete_one(filter)`
 - D) `collection.drop_one(filter)`
9. Which method is used to count the number of documents in a collection that match a filter?
- A) `collection.count()`
 - B) `collection.count_documents(filter)`
 - C) `collection.size(filter)`
 - D) `collection.get_count(filter)`
10. How do you sort the documents retrieved from a collection in descending order by a field named `age`?
- A) `collection.find().sort('age', -1)`
 - B) `collection.sort('age', -1)`
 - C) `collection.find().order_by('age', 'desc')`
 - D) `collection.find().sort([('age', 'desc')])`
11. How do you limit the number of documents returned by a query to 5 documents?
- A) `collection.find().limit(5)`
 - B) `collection.query().max_results(5)`
 - C) `collection.find().max(5)`

- D) ``collection.get(5)``
12. Which method is used to retrieve a single document that matches a filter?
- A) ``collection.find_one(filter)``
 - B) ``collection.get_one(filter)``
 - C) ``collection.find_first(filter)``
 - D) ``collection.fetch_one(filter)``
13. How do you create an index on the ``email`` field in a collection named ``users``?
- A) ``collection.create_index([('email', 1)])``
 - B) ``collection.add_index([('email', 1)])``
 - C) ``collection.index([('email', 1)])``
 - D) ``collection.new_index([('email', 1)])``
14. Which method is used to drop a collection from a database using PyMongo?
- A) ``db.drop_collection('mycollection')``
 - B) ``db.remove_collection('mycollection')``
 - C) ``db.delete_collection('mycollection')``
 - D) ``db.drop('mycollection')``
15. How do you ensure that your PyMongo queries are executed with a timeout?
- A) By setting a timeout parameter in the ``find()`` method
 - B) By configuring the connection settings with a timeout value
 - C) By using the ``timeout()`` method on the cursor
 - D) By specifying a timeout in the database configuration file

MCQs on CRUD Operations in PyMongo

1. Which PyMongo method is used to create a new document in a collection?
- A) ``collection.insert()``
 - B) ``collection.add()``
 - C) ``collection.insert_one()``
 - D) ``collection.create()``
2. How do you read all documents from a collection using PyMongo?
- A) ``collection.get_all()``
 - B) ``collection.read()``
 - C) ``collection.find()``
 - D) ``collection.query()``
3. Which method updates multiple documents that match a filter in PyMongo?
- A) ``collection.update()``
 - B) ``collection.modify_many()``
 - C) ``collection.update_many()``
 - D) ``collection.update_all()``
4. How do you delete a single document based on a filter in PyMongo?
- A) ``collection.remove(filter)``
 - B) ``collection.delete(filter)``

- C) ``collection.delete_one(filter)``
 - D) ``collection.drop(filter)``
5. Which PyMongo method allows you to insert multiple documents into a collection?
- A) ``collection.insert_all()``
 - B) ``collection.insert_many()``
 - C) ``collection.add_many()``
 - D) ``collection.create_many()``
6. How do you update a single document in a collection with a new value using PyMongo?
- A) ``collection.update_one(filter, update)``
 - B) ``collection.set_one(filter, update)``
 - C) ``collection.modify_one(filter, update)``
 - D) ``collection.change_one(filter, update)``
7. Which method is used to find and return a single document that matches a filter in PyMongo?
- A) ``collection.find(filter)``
 - B) ``collection.find_one(filter)``
 - C) ``collection.query_one(filter)``
 - D) ``collection.get_one(filter)``
8. How do you replace an existing document in a collection with a new document using PyMongo?
- A) ``collection.replace(filter, new_document)``
 - B) ``collection.update_one(filter, {'$set': new_document})``
 - C) ``collection.replace_one(filter, new_document)``
 - D) ``collection.update(filter, new_document)``
9. Which method is used to count the number of documents in a collection that match a filter?
- A) ``collection.count()``
 - B) ``collection.count_documents(filter)``
 - C) ``collection.size(filter)``
 - D) ``collection.get_count(filter)``
10. How do you limit the number of documents returned by a query to 10 documents?
- A) ``collection.find().limit(10)``
 - B) ``collection.query().max_results(10)``
 - C) ``collection.find().max(10)``
 - D) ``collection.get(10)``
11. Which method is used to sort documents in ascending order by a field named ``age``?
- A) ``collection.find().sort('age', 1)``
 - B) ``collection.query().order_by('age', 'asc')``
 - C) ``collection.sort('age', 1)``
 - D) ``collection.find().order_by('age')``
12. How do you delete multiple documents that match a filter using PyMongo?
- A) ``collection.remove(filter)``
 - B) ``collection.delete(filter)``

- C) `collection.delete_many(filter)`
 - D) `collection.drop(filter)`
13. What is the purpose of the `$set` operator in an update operation?
- A) To add new fields or update existing fields in a document
 - B) To remove fields from a document
 - C) To create a new document
 - D) To replace an entire document
14. Which PyMongo method allows you to perform a partial update to a document?
- A) `collection.update()`
 - B) `collection.update_one(filter, {'$set': fields})`
 - C) `collection.replace_one(filter, new_document)`
 - D) `collection.modify()`
15. How do you perform a bulk write operation (multiple inserts, updates, or deletes) using PyMongo?
- A) `collection.bulk_write(operations)`
 - B) `collection.batch_write(operations)`
 - C) `collection.execute_bulk(operations)`
 - D) `collection.write_many(operations)`

MCQs on Flask API with MongoDB Database

1. Which library is commonly used in Flask to interact with MongoDB?
- A) `Flask-SQLAlchemy`
 - B) `Flask-MongoEngine`
 - C) `Flask-Redis`
 - D) `Flask-MySQL`
2. What is the purpose of `MongoClient` in a Flask application using PyMongo?
- A) To create a new Flask app
 - B) To connect to a MongoDB server
 - C) To define Flask routes
 - D) To configure Flask settings
3. How do you configure a Flask application to connect to a MongoDB database using PyMongo?
- A) By setting `app.config['MONGO_URI']` to the MongoDB URI
 - B) By initializing `MongoClient` directly in the route
 - C) By using `db.connect()`
 - D) By defining a connection in the `__init__.py` file
4. Which method is used to retrieve a collection from the MongoDB database using PyMongo in Flask?
- A) `db.get_collection('mycollection')`
 - B) `db['mycollection']`
 - C) `db.collection('mycollection')`
 - D) `db.get('mycollection')`

5. How do you define a route in Flask to handle a GET request that retrieves data from a MongoDB collection?
- A) Use the `@app.get('/route')` decorator
 - B) Use the `@app.route('/route', methods=['GET'])` decorator
 - C) Use the `@app.get_route('/route')` decorator
 - D) Use the `@app.handle('/route', method='GET')` decorator
6. Which method is used to insert a document into a MongoDB collection using PyMongo in Flask?
- A) `collection.add(document)`
 - B) `collection.insert(document)`
 - C) `collection.insert_one(document)`
 - D) `collection.create(document)`
7. How do you update a document in a MongoDB collection using Flask and PyMongo?
- A) `collection.update(filter, new_values)`
 - B) `collection.update_one(filter, {'$set': new_values})`
 - C) `collection.replace_one(filter, new_values)`
 - D) `collection.modify(filter, new_values)`
8. What is the purpose of the `jsonify` function in Flask?
- A) To convert a dictionary to JSON format
 - B) To create a new Flask route
 - C) To connect to the MongoDB database
 - D) To initialize the Flask application
9. How do you handle a POST request to insert data into a MongoDB collection using Flask?
- A) Use `request.get_json()` to retrieve the data and then `collection.insert_one(data)`
 - B) Use `request.form()` to get the data and then `collection.add(data)`
 - C) Use `request.json()` to get the data and then `collection.create(data)`
 - D) Use `request.data` to get the data and then `collection.insert(data)`
10. How do you delete a document from a MongoDB collection using Flask and PyMongo?
- A) `collection.remove(filter)`
 - B) `collection.delete_one(filter)`
 - C) `collection.drop(filter)`
 - D) `collection.erase(filter)`
11. How do you set up error handling for a Flask route that interacts with MongoDB?
- A) Use the `try-except` block within the route function
 - B) Define error handlers in `config.py`
 - C) Use `Flask-ErrorHandler` extension
 - D) Use `try-catch` within the `__init__.py` file
12. What method is used to find all documents in a MongoDB collection in a Flask route?
- A) `collection.find_all()`
 - B) `collection.get_all()`
 - C) `collection.find()`
 - D) `collection.query_all()`