

#01

test> db.test.help()

Collection Class:

aggregate	Calculates aggregate values for the data in a collection or a view.
bulkWrite	Performs multiple write operations with controls for order of execution.
count	Returns the count of documents that would match a query.
find() query	Returns the count of documents that match the query for a collection or view.
countDocuments	Removes all documents that match the filter from a collection.
for a collection or view.	Removes a single document from a collection.
deleteMany	Finds the distinct values for a specified field across a single collection or view and returns the results in an array.
collection.	Returns the count of all documents in a collection or view.
deleteOne	Selects documents in a collection or view.
distinct	Modifies and returns a single document.
a single collection or view and returns the results in an array.	Selects documents in a collection or view.
estimatedDocumentCount	Renames a collection.
view.	Deletes a single document based on the filter and sort criteria, returning the deleted document.
find	Modifies and replaces a single document based on the filter and sort criteria.
findAndModify	Updates a single document based on the filter and sort criteria.
findOne	Inserts a document or documents into a collection.
renameCollection	Inserts multiple documents into a collection.
findOneAndDelete	Inserts a document into a collection.
criteria, returning the deleted document.	Checks if a collection is capped.
findOneAndReplace	Removes documents from a collection.
filter and sort criteria.	Replaces a single document within the collection based on the filter.
findOneAndUpdate	Modifies an existing document or documents in a collection.
criteria.	Updates all documents that match the specified filter.
insert	Updates a single document within the collection based on the filter.
insertMany	Compacts structured encryption data.
insertOne	calls {convertToCapped:'coll', size:maxBytes} command.
isCapped	Creates one or more indexes on a collection.
remove	Creates one index on a collection.
replaceOne	Creates one index on a collection.
on the filter.	Returns an array that holds a list of documents that identify and describe the existing indexes on the collection.
update	Alias for getIndexes. Returns an array that holds a list of documents that identify and describe the existing indexes on the collection.
collection.	Alias for getIndexes. Returns an array that holds a list of documents that identify and describe the existing indexes on the collection.
updateMany	Updates all documents that match the specified filter for a collection.
for a collection.	Updates a single document within the collection based on the filter.
updateOne	Modifies an existing document or documents in a collection.
on the filter.	Updates all documents that match the specified filter.
compactStructuredEncryptionData	Updates a single document within the collection based on the filter.
convertToCapped	Compacts structured encryption data.
createIndexes	Creates one or more indexes on a collection.
createIndex	Creates one index on a collection.
ensureIndex	Creates one index on a collection.
getIndexes	Returns an array that holds a list of documents that identify and describe the existing indexes on the collection.
getIndexSpecs	Alias for getIndexes. Returns an array that holds a list of documents that identify and describe the existing indexes on the collection.
list of documents that identify and describe the existing indexes on the collection.	Alias for getIndexes. Returns an array that holds a list of documents that identify and describe the existing indexes on the collection.
getIndices	Alias for getIndexes. Returns an array that holds a list of documents that identify and describe the existing indexes on the collection.
list of documents that identify and describe the existing indexes on the collection.	Return an array of key patterns for indexes defined on a collection.
getIndexKeys	Drops the specified index or indexes (except the index on the _id field) from a collection.
collection	Drops or removes the specified index from a collection.
dropIndexes	Reports the total size used by the indexes on a collection.
on the _id field) from a collection.	Rebuilds all existing indexes on a collection.
dropIndex	
collection.	
totalIndexSize	
collection.	
reIndex	

getDB	Get current database.
getMongo	Returns the Mongo object.
dataSize	This method provides a wrapper around the size output
of the collStats (i.e. db.collection.stats()) command.	
storageSize	The total amount of storage allocated to this
collection for document storage.	
totalSize	The total size in bytes of the data in the collection
plus the size of every index on the collection.	
drop	Removes a collection or view from the database.
exists	Returns collection infos if the collection exists or
null otherwise.	
getFullName	Returns the name of the collection prefixed with the
database name.	
getName	Returns the name of the collection.
runCommand	Runs a db command with the given name where the first
param is the collection name.	
explain	Returns information on the query plan.
stats	Returns statistics about the collection.
latencyStats	returns the \$latencyStats aggregation for the
collection. Takes an options document with an	optional boolean 'histograms' field.
initializeOrderedBulkOp	Initializes an ordered bulk command. Returns an
instance of Bulk	
initializeUnorderedBulkOp	Initializes an unordered bulk command. Returns an
instance of Bulk	
getPlanCache	Returns an interface to access the query plan cache
for a collection. The interface provides methods to view and clear the query plan cache.	
mapReduce	Calls the mapReduce command
validate	Calls the validate command. Default full value is
false	
getShardVersion	Calls the getShardVersion command
getShardDistribution	Prints the data distribution statistics for a sharded
collection.	
getShardLocation	Returns a document containing the shards where this
collection is located as well as whether the collection itself is sharded.	
watch	Opens a change stream cursor on the collection
hideIndex	Hides an existing index from the query planner.
unhideIndex	Unhides an existing index from the query planner.
analyzeShardKey	Returns metrics for evaluating a shard key. That is,
'key' can be a candidate shard key for an unsharded or sharded collection, or the current shard key	
for a sharded collection.	
configureQueryAnalyzer	Starts or stops collecting metrics about reads and
writes against an unsharded or sharded collection.	
checkMetadataConsistency	Returns a cursor with information about metadata
inconsistencies	
getSearchIndexes	Returns an array that holds a list of documents that
identify and describe the existing search indexes on the collection.	
createSearchIndex	Creates one search indexes on a collection
createSearchIndexes	Creates one or more search indexes on a collection
dropSearchIndex	Drops or removes the specified search index from a
collection.	
updateSearchIndex	Updates the sepecified search index.