

Starting & accessing	
arangod /path/to/my/db	start server
arangodconsolelog error /path/to/my/db	start emergency console (do not use with a db which has a server attached to it!)
http://localhost:8529/_admin/html/index.html	access admin front end in browser
arangosh	start ArangoDB shell

arangod frequently used options	
log	set log level: fatal, error, warning, info, debug, trace
server.http-port <i>host:port</i>	set address and port for HTTP-Clients
daemon	run as daemon/background process

Database methods in arangosh	
dbcreate(collection-name, properties)	create collection (with properties)
dbcreateEdgeCollection(collection-name, properties)	create an edge collection (with properties)
dbcollection(collection-name collection-id)	get collection
dbcollections()	list all collections
db.collection-name	get a collection by name
dbdrop(collection-name collection-id)	drop collection with indexes
dbtruncate(collection-name collection-id)	remove collection, keep indexes

Collection methods in arangosh	
collection.drop()	drop collection with indexes
collection.truncate()	remove documents, keep indexes
collection.properties()	get all document properties
collection.properties(properties)	change property
collection.figures()	get all collection figures
collection.load()	load collection into memory
collection.unload()	start to unload a collection
collection.rename(new-name)	rename collection to new-name

Document methods in arangosh	
collection.document(document)	get document by identifier
collection.save(data)	create new document
collection.replace(document, data)	replace existing document
collection.update(document)	partially update
collection.remove(document)	remove document
dbdocument(document document-handle)	get document by identifier handle
dbreplace(document document-handle,data)	replace existing document
- 1 , , ,	

dbupdate(document)	partially update document
dbremove(document)	remove document

Edges in arangosh	
edge-collection.save(from, to, document)	save new edge
edge-collection.edges(vertex)	find edges from (outbound) to (inbound) vertex
edge-collection.inEdges(vertices)	find all edges ending in (inbound)
edge-collection.outEdges(vertices)	find all edges starting from (outbound)

Queries in arangosh	
collection.all()	select all documents and return cursor
collection.any()	select a random document
collection.byExample( <i>example</i> )	select all documents that matches the given example
collection.firstExample(example)	select the first document that matches the given example
collection.range(attribute,left,right)	select all documents with attribute >= left and < right
collection.removeByExample(example)	remove all documents that match the example
collection. replaceByExample	remove all documents that match the example
(example, newValue)	
collection. updateByExample	remove all documents that match the example
(example, newValue)	
collection.count()	returns the number of living documents in the collection
collection.toArray()	convert the collection into an array of documents (not for production!)

Geo Queries in arangosh	
collection.near( <i>latitude,longitude</i> )	get documents near the given coordinates
$collection. within ({\it latitude, longitude, distance})$	get all documents within a radious of <i>distance</i> around the given coordinates
collection.geo(location)	the next near or within operator will use the specific geo-spatial index

Pagination in arangosh	
query.limit(number)	limits a result to the first number documents
query.skip(number)	skips the first number documents

Sequential Access And Cursors in	arangosh
query.hasNext()	returns true if the cursor still has documents
query.next()	advance cursor
query.dispose()	free resources associated with a cursor
query.count()	returns number of documents in the result set