#01
D:\ts\nosql.mongodb>mongosh --help

```
  $ mongosh [options] [db address] [file names (ending in .js or .mongodb)]

  Options:

    -h, --help                                Show this usage information
    -f, --file [arg]                          Load the specified mongosh script
        --host [arg]                          Server to connect to
        --port [arg]                          Port to connect to
        --build-info                          Show build information
        --version                             Show version information
        --quiet                               Silence output from the shell during the connection
process
        --shell                               Run the shell after executing files
        --nodb                                Don't connect to mongod on startup - no 'db address'
[arg] expected
        --norc                                Will not run the '.mongoshrc.js' file on start up
        --eval [arg]                          Evaluate javascript
        --json[=canonical|relaxed]            Print result of --eval as Extended JSON, including
errors
        --retryWrites[=true|false]            Automatically retry write operations upon transient
network errors (Default: true)

  Authentication Options:

    -u, --username [arg]                      Username for authentication
    -p, --password [arg]                      Password for authentication
        --authenticationDatabase [arg]        User source (defaults to dbname)
        --authenticationMechanism [arg]       Authentication mechanism
        --awsIamSessionToken [arg]            AWS IAM Temporary Session Token ID
        --gssapiServiceName [arg]             Service name to use when authenticating using
GSSAPI/Kerberos
        --sspiHostnameCanonicalization [arg]  Specify the SSPI hostname canonicalization (none or
forward, available on Windows)
        --sspiRealmOverride [arg]             Specify the SSPI server realm (available on Windows)

  TLS Options:

        --tls                                 Use TLS for all connections
        --tlsCertificateKeyFile [arg]         PEM certificate/key file for TLS
        --tlsCertificateKeyFilePassword [arg] Password for key in PEM file for TLS
        --tlsCAFile [arg]                     Certificate Authority file for TLS
        --tlsAllowInvalidHostnames            Allow connections to servers with non-matching
hostnames
        --tlsAllowInvalidCertificates         Allow connections to servers with invalid certificates
        --tlsCertificateSelector [arg]        TLS Certificate in system store (Windows and macOS
only)
        --tlsCRLFile [arg]                    Specifies the .pem file that contains the Certificate
Revocation List
        --tlsDisabledProtocols [arg]          Comma separated list of TLS protocols to disable
[TLS1_0,TLS1_1,TLS1_2]
        --tlsFIPSMode                         Enable the system TLS library's FIPS mode

  API version options:

        --apiVersion [arg]                    Specifies the API version to connect with
        --apiStrict                           Use strict API version mode
        --apiDeprecationErrors                Fail deprecated commands for the specified API version

  FLE Options:

        --awsAccessKeyId [arg]                AWS Access Key for FLE Amazon KMS
```

```
        --awsSecretAccessKey [arg]            AWS Secret Key for FLE Amazon KMS
        --awsSessionToken [arg]               Optional AWS Session Token ID
        --keyVaultNamespace [arg]             database.collection to store encrypted FLE parameters
        --kmsURL [arg]                        Test parameter to override the URL of the KMS endpoint

  OIDC auth options:

        --oidcFlows[=auth-code,device-auth]    Supported OIDC auth flows
        --oidcRedirectUri[=url]               Local auth code flow redirect URL
[http://localhost:27097/redirect]
        --oidcTrustedEndpoint                 Treat the cluster/database mongosh as a trusted
endpoint
        --oidcIdTokenAsAccessToken            Use ID tokens in place of access tokens for auth
        --oidcDumpTokens[=mode]               Debug OIDC by printing tokens to mongosh's output
[redacted|include-secrets]
        --oidcNoNonce                         Don't send a nonce argument in the OIDC auth request

  DB Address Examples:

        foo                                   Foo database on local machine
        192.168.0.5/foo                       Foo database on 192.168.0.5 machine
        192.168.0.5:9999/foo                  Foo database on 192.168.0.5 machine on port 9999
        mongodb://192.168.0.5:9999/foo        Connection string URI can also be used

  File Names:

        A list of files to run. Files must end in .js and will exit after unless --shell is
specified.

  Examples:

        Start mongosh using 'ships' database on specified connection string:
        $ mongosh mongodb://192.168.0.5:9999/ships

  For more information on usage: https://mongodb.com/docs/mongodb-shell.
```