

# GridFS in MongoDB

GridFS is the MongoDB Specification for storing and collecting large files such as images, audio, video files, etc. It is a type of file system to store files but its data is stored inside MongoDB collections. .

## Introducing GridFS

**GridFS** is the MongoDB Specification for storing and collecting large files such as images, audio, video files, etc. It is a type of file system to store files but its data is stored inside MongoDB collections. . GridFS has the ability to store files even when its size is larger than the size limit of 16 MB of the Document. .

GridFS divides a file into chunk and stores each piece of data in a separate document, its maximum size is 255k.

By default, GridFS uses two **fs.files** and **fs.chunks** Collection to store file metadata and chunk. Each Chunk is identified by the unique **\_id** ObjectId field. **fs.files** acts as a parent Document. The **files\_id** field in fs.chunks links Chunk to its parent Document.

Below is the sample Document of **fs.files** collection:

```
{ "filename" : "test.txt" , "chunkSize" : NumberInt ( 261120 ) , "uploadDate" : :
```

Document determines the file name, chunk size, update date, and length.

Below is the sample Document of fs.chunks document:

```
{ "files_id" : ObjectId ( "534a75d19f54bfec8a2fe44b" ) , "n" : NumberInt ( 0 ) ,
```

## Add File to GridFS

Now, we will store an mp3 file by using GridFS with the **put** command. We will use the **mongofiles.exe** utility **located** in the bin directory of the MongoDB installation directory.

Open the command prompt, navigate to mongofiles.exe in the bin directory of the MongoDB installation directory and compose the following code:

```
> mongofiles . exe - d gridfs put song . mp3
```

Here, **gridfs** is the name of the Database in which the **file** will be stored. If this Database does not exist, MongoDB will automatically create a new Document. **song.mp3** is the name of the uploaded file. To view the document's file in Database, you use the find query.

```
> db . fs . files . find ( )
```

The above command will result:

```
{ _id : ObjectId ( '534a811bf8b4aa4d33fdf94d' ), filename : "song.mp3" , chu
```

We can also see all Chunk in the fs.chunks collection related to the file stored with the following code, using the document id returned from the previous query.

```
> db . fs . chunks . find ( { files_id : ObjectId ( '534a811bf8b4aa4d33fdf94d' )
```

For me, the query returns 40 documents, meaning that all mp3 documents are divided into 40 chunk.

## According to Tutorialspoint

Last lesson: Auto-Increment Sequence in MongoDB

Next article: Capped Collection in MongoDB

You finished reading the article "**GridFS in MongoDB**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.