

Prof R. Madana Mohana



BIG DATA ANALYTICS

MongoDB

Insert Document

<https://www.youtube.com/c/RASINENIMADANAMOHANA>

MongoDB: Insert Document

MongoDB: *Insert Document*

The insert() Method:

To insert data into MongoDB collection, we need to use MongoDB's **insert()** or **save()** method.

Syntax:

The basic syntax of **insert()** command is as follows:

```
>db.COLLECTION_NAME.insert(document)
```

MongoDB: *Insert Document*

The insert() Method: *Example*

```
> show dbs
admin          0.000GB
config        0.000GB
local         0.000GB
mydb          0.000GB
sampleDB     0.000GB
test         0.000GB
```

MongoDB: *Insert Document*

The insert() Method: *Example*

```
> use test
```

```
switched to db test
```

```
> show collections
```

```
bdacollection
```

```
mongodbcollection
```

```
mycollection
```

```
mycollection1
```

```
>
```

MongoDB: *Insert Document*

The insert() Method: *Example*

```
> db.courses.insert({
...   ...   ...   _id : ObjectId("507f191e810c19729de860ea"),
...   ...   ...   title: "Big Data Analytics",
...   ...   ...   description: "Core course for IT",
...   ...   ...   by: "IT Department",
...   ...   ...   semester: "VII Sem IT-1",
...   ...   ...   tags: ['mongodb', 'database', 'NoSQL'],
...   ...   ...   studentsenrolled: 68
...   ...   ...   })
WriteResult({ "nInserted" : 1 })
```

MongoDB: *Insert Document*

The insert() Method: *Example*

```
> show collections
```

```
bdacollection
```

```
courses
```

```
mongodbcollection
```

```
mycollection
```

```
mycollection1
```

```
>
```

MongoDB: *Insert Document*

The insert() Method: *Example*

```
Command Prompt - mongo
> db.courses.insert({
... .. _id : ObjectId("507f191e810c19729de860ea"),
... .. title: "Big Data Analytics",
... .. description: "Core course for IT",
... .. by: "IT Department",
... .. semester: "VII Sem IT-1",
... .. tags: ['mongodb', 'database', 'NoSQL'],
... .. studentsenrolled: 68
... .. })
WriteResult({ "nInserted" : 1 })
> show collections
bdacollection
courses
mongodbcollection
mycollection
mycollection1
> _
```


MongoDB: *Insert Document*

The insert() Method:

- If the **collection** doesn't exist in the database, then MongoDB will **create** this **collection** and then **insert** a **document** into it.
- In the **inserted document**, if we don't specify the **_id** parameter, then MongoDB assigns a **unique ObjectId** for this document.

MongoDB: *Insert Document*

The insert() Method:

- **_id** is **12 bytes** hexadecimal number unique for every document in a collection.
- **12 bytes** are divided as follows:

_id: ObjectId(4 bytes timestamp, 3 bytes machine id, 2 bytes process id, 3 bytes incrementer)

MongoDB: *Insert Document*

The insert() Method:

We can also pass an **array of documents** into the **insert()** method as shown below:

```
> db.createCollection("blogspost")
```

```
{ "ok" : 1 }
```

```
> show collections
```

```
bdacollection
```

```
blogspost
```

```
courses
```

```
mongodbcollection
```

```
mycollection
```

```
mycollection1
```

```
>
```

MongoDB: *Insert Document*

The insert() Method:

We can also pass an **array of documents** into the **insert()** method as shown below:

```
> db.createCollection("blogspost")
```

```
{ "ok" : 1 }
```

```
> show collections
```

```
bdacollection
```

```
blogspost
```

```
courses
```

```
mongodbcollection
```

```
mycollection
```

```
mycollection1
```

```
>
```

MongoDB: *Insert Document*

The insert() Method:

```
> db.blogspost.insert([
...   {
...     title: "Big Data Analytics",
...     description: "Chapter-3 is MongoDB",
...     by: "IT Department",
...     url: "https://www.mongodb.com/",
...     tags: ["mongodb", "database", "NoSQL"],
...     likes: 100
...   },
```

MongoDB: *Insert Document*

The insert() Method:

```
... .. {  
... .. title: "Big Data Analytics Lab",  
... .. description: "Big Data Analytics lab is related to BDA  
Course",  
... .. by: "IT Department",  
... .. url: "https://www.mongodb.com/",  
... .. tags: ["mongodb", "database", "NoSQL"],  
... .. likes: 50,  
... .. }
```

MongoDB: *Insert Document*

The insert() Method:

```
... .. comments: [  
... .. {  
... ..   user: "user1",  
... ..   message: "My first comment",  
... ..   dateCreated: new Date(2022,11,10,2,35),  
... ..   like: 10  
... .. }  
... .. ]  
... .. }  
... .. ]  
... .. ])
```

MongoDB: *Insert Document*

The insert() Method:

```
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 2,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
>
```


MongoDB: *Insert Document*

The insert() Method:

```
Command Prompt - mongo
> db.blogspot.insert([
... .. {
... .. title: "Big Data Analytics",
... .. description: "Chapter-3 is MongoDB",
... .. by: "IT Department",
... .. url: "https://www.mongodb.com/",
... .. tags: ["mongodb", "database", "NoSQL"],
... .. likes: 100
... .. },
... .. {
... .. title: "Big Data Analytics Lab",
... .. description: "Big Data Analytics lab is related to BDA Course",
... .. by: "IT Department",
... .. url: "https://www.mongodb.com/",
... .. tags: ["mongodb", "database", "NoSQL"],
... .. likes: 50,
... .. comments: [
... .. {
... .. user: "user1",
... .. message: "My first comment",
... .. dateCreated: new Date(2022,11,10,2,35),
... .. like: 10
... .. }
... .. ]
... .. }
... .. ]
... .. ])
```

MongoDB: *Insert Document*

The insert() Method:

```
Command Prompt - mongo
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 2,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
>
```

MongoDB: *Insert Document*

The insert() Method:

- To **insert** the **document** we can use **db.post.save(document)** also.
- If we don't specify **_id** in the **document** then **save()** method will work same as **insert()** method.
- If we specify **_id** then it will replace whole data of document containing **_id** as specified in **save()** method.

MongoDB: *Insert Document*

The insertOne() method:

- If we need to **insert only one document** into a **collection** we can use this method.

Syntax:

The basic syntax of **insertOne()** command is as follows:

```
>db.COLLECTION_NAME.insertOne(document)
```

MongoDB: *Insert Document*

The insertOne() method: *Example*

Following example creates a new collection named

Employee and inserts a document using the **insertOne()** method.

MongoDB: *Insert Document*

The insertOne() method: *Example*

```
> db.createCollection("Employee")
```

```
{ "ok" : 1 }
```

```
> show collections
```

```
Employee
```

```
bdacollection
```

```
blogspost
```

```
courses
```

```
mongodbcollection
```

```
mycollection
```

```
mycollection1
```

```
>
```

MongoDB: *Insert Document*

The insertOne() method: *Example*

```
> db.Employee.insertOne(
...   {
...   First_Name: "R Madana",
...   Last_Name: "Mohana",
...   Date_Of_Birth: "1981-06-15",
...   e_mail: "rmmnaidu@gmail.com",
...   phone: "0123456789"
...   })
{
  "acknowledged" : true,
  "insertedId" : ObjectId("634bfe4112c650f5e845ddc9")
}
>
```

MongoDB: *Insert Document*

The insertOne() method: *Example*

```
Command Prompt - mongo
> db.Employee.insertOne(
...   {
...     First_Name: "R Madana",
...     Last_Name: "Mohana",
...     Date_Of_Birth: "1981-06-15",
...     e_mail: "rmmnaidu@gmail.com",
...     phone: "0123456789"
...   })
{
  "acknowledged" : true,
  "insertedId" : ObjectId("634bfe4112c650f5e845ddc9")
}
>
```


MongoDB: *Insert Document*

The insertMany() method:

- We can **insert** multiple **documents** using the **insertMany()** method.
- To use this method we need to pass an **array of documents**.

MongoDB: *Insert Document*

The insertMany() method: *Syntax*

The basic syntax of **insertMany()** command is as follows:

```
>db.COLLECTION_NAME.insertMany(document)
```

MongoDB: *Insert Document*

The insertMany() method: *Example*

Following example inserts three different documents into the **Employee** collection using the **insertMany()** method.

MongoDB: *Insert Document*

The insertMany() method: *Example*

```
> db.Employee.insertMany(  
... .. [  
... .. {  
... .. First_Name: "R Madana",  
... .. Last_Name: "Mohana",  
... .. Date_Of_Birth: "1981-06-15",  
... .. e_mail: "rmmnaidu@gmail.com",  
... .. phone: "0123456789"  
... .. },
```

MongoDB: *Insert Document*

The insertMany() method: *Example*

```
... .. {  
... .. First_Name: "Madan",  
... .. Last_Name: "Mohan",  
... .. Date_Of_Birth: "1980-12-26",  
... .. e_mail: "r2431663@yahoo.com",  
... .. phone: "9876543210"  
... .. },
```

MongoDB: *Insert Document*

The insertMany() method: *Example*

```
... .. {  
... .. First_Name: "Madanamohana",  
... .. Last_Name: "Rasineni",  
... .. Date_Of_Birth: "1982-06-20",  
... .. e_mail: "madanmohananar@gmail.com",  
... .. phone: "9440793154"  
... .. }  
... .. ]  
... .. )
```

MongoDB: *Insert Document*

The insertMany() method: *Example*

```
{
    "acknowledged" : true,
    "insertedIds" : [
        ObjectId("634c03ef12c650f5e845ddca"),
        ObjectId("634c03ef12c650f5e845ddcb"),
        ObjectId("634c03ef12c650f5e845ddcc")
    ]
}
```

MongoDB: *Insert Document*

The insertMany() method: *Example*

```
Command Prompt - mongo
> db.Employee.insertMany(
... .. [
... .. {
... ..   First_Name: "R Madana",
... ..   Last_Name: "Mohana",
... ..   Date_Of_Birth: "1981-06-15",
... ..   e_mail: "rmmnaidu@gmail.com",
... ..   phone: "0123456789"
... .. },
... .. {
... ..   First_Name: "Madan",
... ..   Last_Name: "Mohan",
... ..   Date_Of_Birth: "1980-12-26",
... ..   e_mail: "r2431663@yahoo.com",
... ..   phone: "9876543210"
... .. },
... .. {
... ..   First_Name: "Madanamohana",
... ..   Last_Name: "Rasineni",
... ..   Date_Of_Birth: "1982-06-20",
... ..   e_mail: "madanmohana@gmail.com",
... ..   phone: "9440793154"
... .. }
... .. ]
... .. )
```


MongoDB: *Insert Document*

The insertMany() method: *Example*

Command Prompt - mongo

```
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("634c03ef12c650f5e845ddca"),
    ObjectId("634c03ef12c650f5e845ddcb"),
    ObjectId("634c03ef12c650f5e845ddcc")
  ]
}
```