

# BlogPosts Collection

Create a collection by the name blogPosts and it has 3 fields id, title and comments. In the collection the comments field is an array which consists of user details. Each collection

consists of two user details inside the comments array- user name and text

Demonstrate the following

1. Adding an element into array
2. Display second element
3. Display size of the array
4. Display first two elements of the array
5. Update the document with id 4 and replace the element present in 1st index position of the array with another array

```
Atlas atlas-6navq6-shard-0 [primary] DB> db.blogPosts.insertOne({
...   title: "Another Sample Title",
...   comments: [
...     { userName: "Emily", text: "I found this really helpful, thanks!" }
...   ]
... });
{
  acknowledged: true,
  insertedId: ObjectId('660bd09749e8e6b3e924e25f')
}
Atlas atlas-6navq6-shard-0 [primary] DB> db.blogPosts.find()
[
  {
    _id: ObjectId('60f9f1877b89380f68d885a5'),
    title: 'Sample Title',
    comments: [
      { userName: 'John', text: 'This is a great article!' },
      {
        userName: 'Emily',
        text: 'I found this really helpful, thanks!'
      }
    ]
  },
  {
    _id: ObjectId('660bd00749e8e6b3e924e25e'),
    title: 'Another Sample Title',
    comments: [
      {
        userName: 'Emily',
        text: 'I found this really helpful, thanks!'
      }
    ]
  },
  {
    _id: ObjectId('660bd09749e8e6b3e924e25f'),
    title: 'Another Sample Title',
    comments: [
      {
        userName: 'Emily',
        text: 'I found this really helpful, thanks!'
      }
    ]
  }
]
Atlas atlas-6navq6-shard-0 [primary] DB>
```

```

Atlas atlas-6mavq6-shard-0 [primary] DB> db.blogPosts.aggregate([
.. { $unwind: "$comments" }, // Unwind the comments array
.. { $project: { _id: 0, commentText: "$comments.text" } } // Project only the text
.. ]).forEach(function(doc) {
..   printjson(doc);
.. })

commentText: 'This is a great article!'

commentText: 'I found this really helpful, thanks!'

commentText: 'I found this really helpful, thanks!'

commentText: 'I found this really helpful, thanks!'

```

```

Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.insertOne({_id:1, title:
"Introduction to MongoDB", comments: [ { userName: "Alice", text: "Great article!" }, {
userName: "Bob", text: "Looking forward to more content." } ] })
{ acknowledged: true, insertedId: 1 }

```

```

Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.insertOne({_id:2, title:
"Advanced MongoDB Techniques", comments: [ { userName: "Charlie", text: "Very
informative." }, { userName: "David", text: "Helped me a lot!" } ] })
{ acknowledged: true, insertedId: 2 }

```

```

Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.insertOne({_id:3, title:
"MongoDB Performance Optimization", comments: [ { userName: "Eve", text: "I have
a question." }, { userName: "Frank", text: "This is exactly what I needed!" } ] })
{ acknowledged: true, insertedId: 3 }

```

```

Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.update(
... { _id: ObjectId("document_id") },
... { $push: { comments: { userName: "John", text: "This is a new comment." } } }
... )

```

BSONError: input must be a 24 character hex string, 12 byte Uint8Array, or an integer

```

Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.update( { _id:1 }, {
$push: { comments: { userName: "John", text: "This is a new comment." } } } )

```

DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.

```

{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

```

}
Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.find({}, { comments: {
$slice: [1, 1] }})
[
  {
    _id: 1,
    title: 'Introduction to MongoDB',
    comments: [ { userName: 'Bob', text: 'Looking forward to more content.' } ]
  },
  {
    _id: 2,
    title: 'Advanced MongoDB Techniques',
    comments: [ { userName: 'David', text: 'Helped me a lot!' } ]
  },
  {
    _id: 3,
    title: 'MongoDB Performance Optimization',
    comments: [ { userName: 'Frank', text: 'This is exactly what I needed!' } ]
  }
]
Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.aggregate([
... { $unwind: "$comments" }, // Unwind the comments array
... { $project: { _id: 0, commentText: "$comments.text" } } // Project only the text field
... ])
[
  { commentText: 'Great article!' },
  { commentText: 'Looking forward to more content.' },
  { commentText: 'This is a new comment.' },
  { commentText: 'Very informative.' },
  { commentText: 'Helped me a lot!' },
  { commentText: 'I have a question.' },
  { commentText: 'This is exactly what I needed!' }
]
Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.aggregate([
... { $project: { commentCount: { $size: "$comments" } } }
... ])
[
  { _id: 1, commentCount: 3 },
  { _id: 2, commentCount: 2 },
  { _id: 3, commentCount: 2 }
]
Atlas atlas-12eb3b-shard-0 [primary] MY_DB>
Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.aggregate([
... { $project: { firstTwoComments: { $slice: ["$comments", 2] } } }

```

```

... ])
[
  {
    _id: 1,
    firstTwoComments: [
      { userName: 'Alice', text: 'Great article!' },
      { userName: 'Bob', text: 'Looking forward to more content.' }
    ]
  },
  {
    _id: 2,
    firstTwoComments: [
      { userName: 'Charlie', text: 'Very informative.' },
      { userName: 'David', text: 'Helped me a lot!' }
    ]
  },
  {
    _id: 3,
    firstTwoComments: [
      { userName: 'Eve', text: 'I have a question.' },
      { userName: 'Frank', text: 'This is exactly what I needed!' }
    ]
  }
]
Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.update(
... { _id: ObjectId("document_id") },
... { $set: { "comments.1": { userName: "Alice", text: "Replaced comment." } } }
... )
BSONError: input must be a 24 character hex string, 12 byte Uint8Array, or an
integer
Atlas atlas-12eb3b-shard-0 [primary] MY_DB> db.blogPosts.update( { _id: 3 }, { $set:
{ "comments.1": { userName: "Alice", text: "Replaced comment." } } } )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```