

MongoDB Array Operations Explained =====

1. \$push – Add an element to an array ----- Adds a new value to an array field.

Example: db.students.updateOne({ name: "Ali" }, { \$push: { subjects: "Math" } })

Before: { name: "Ali", subjects: ["Science"] } After: { name: "Ali", subjects: ["Science", "Math"] }

----- \$each – Push multiple values -----

Used with \$push to insert multiple items.

db.students.updateOne({ name: "Ali" }, { \$push: { subjects: { \$each: ["English", "History"] } } })

----- \$addToSet – Add only if it doesn't already exist

----- Prevents duplicates.

db.students.updateOne({ name: "Ali" }, { \$addToSet: { subjects: "Science" } })

----- \$pop – Remove first or last element

----- 1 removes last, -1 removes first.

db.students.updateOne({ name: "Ali" }, { \$pop: { subjects: 1 } })

----- \$pull – Remove specific value(s)

----- Removes elements that match a condition.

db.students.updateOne({ name: "Ali" }, { \$pull: { subjects: "Math" } })

----- \$push with \$slice – Limit array length

----- Adds items and keeps only the last N.

db.students.updateOne({ name: "Ali" }, { \$push: { scores: { \$each: [95, 87], \$slice: -3 } } })

----- \$position – Insert at specific index

----- Used with \$each.

db.students.updateOne({ name: "Ali" }, { \$push: { subjects: { \$each: ["Geography"], \$position: 1 } } })

----- \$set – Update element at a specific index

----- db.students.updateOne({ name: "Ali" }, { \$set: { "subjects.1": "Biology" } })

----- \$ – Positional Operator -----

Updates first matching array element.

db.students.updateOne({ name: "Ali", "scores.subject": "Math" }, { \$set: { "scores.\$.marks": 90 } })

----- \$elemMatch – Match conditions inside arrays

----- db.students.find({ scores: { \$elemMatch: { subject: "Math", marks: { \$gte: 80 } } } })

----- \$size – Match arrays by length

----- db.students.find({ subjects: { \$size: 3 } })

----- \$all – Match arrays containing all given values

----- db.students.find({ subjects: { \$all: ["Math", "Science"] } })

----- Positional Operator (\$) Explained

----- Updates the first array element that matches a condition.

Example: db.students.updateOne({ name: "Ali", "scores.subject": "Science" }, { \$set: { "scores.\$.marks": 85 } })

----- \$elemMatch Explained -----

Ensures all conditions apply to the same array element.

Example: db.students.find({ scores: { \$elemMatch: { subject: "Math", marks: { \$gte: 80 } } } })

Projection example: db.students.find({ name: "Ali" }, { scores: { \$elemMatch: { subject: "Math" } } })

Output: { name: "Ali", scores: [{ subject: "Math", marks: 85 }] }

----- Key Points Summary ----- - \$push

adds elements, \$slice limits size. - \$set updates or creates fields. - \$ adds positional updates to first matching array element. - \$elemMatch ensures conditions apply to the same element. - \$all checks if array contains all given values. - \$size matches array length.