

## MongoDB Error & Status Codes – Detailed Developer Guide

This document explains commonly encountered MongoDB error codes, their causes, real-world scenarios, and recommended handling strategies when building backend applications using Flask, FastAPI, or other REST frameworks.

### 1. Common MongoDB Error Codes

| Error Code | Error Name                | Description                    | Common Cause                      | Recommended Handling        |
|------------|---------------------------|--------------------------------|-----------------------------------|-----------------------------|
| 2          | BadValue                  | Invalid value or argument      | Wrong query or invalid data type  | Validate input before query |
| 13         | Unauthorized              | Permission denied              | Missing role or privilege         | Return HTTP 401/403         |
| 18         | AuthenticationFailed      | Authentication failed          | Invalid credentials               | Verify username/password    |
| 26         | NamespaceNotFound         | Collection not found           | Wrong collection name             | Handle as 404 Not Found     |
| 50         | ExceededTimeLimit         | Query timeout                  | Long-running query                | Optimize query / add index  |
| 59         | CommandNotFound           | Invalid MongoDB command        | Unsupported or misspelled command | Correct command usage       |
| 66         | ImmutableField            | Immutable field update attempt | Updating _id field                | Remove immutable field      |
| 85         | IndexOptionsConflict      | Index conflict                 | Duplicate index creation          | Drop or modify index        |
| 11000      | DuplicateKeyError         | Duplicate unique key           | Unique index violation            | Return HTTP 409 Conflict    |
| 121        | DocumentValidationFailure | Schema validation failed       | Document violates schema          | Fix request payload         |

## 2. MongoDB Operation Status Indicators

MongoDB does not use HTTP status codes internally. Instead, it returns operation result objects that indicate success or failure.

| Operation | Success Indicator     | Failure Indicator              |
|-----------|-----------------------|--------------------------------|
| Insert    | inserted_id present   | DuplicateKeyError / WriteError |
| Find      | Cursor with documents | Empty cursor / exception       |
| Update    | matched_count > 0     | matched_count = 0              |
| Delete    | deleted_count > 0     | deleted_count = 0              |

## 3. HTTP Status Code Mapping (MongoDB + REST API)

| HTTP Code | Meaning               | Usage Scenario          |
|-----------|-----------------------|-------------------------|
| 200       | OK                    | Successful GET / UPDATE |
| 201       | Created               | Document inserted       |
| 400       | Bad Request           | Invalid input / JSON    |
| 401       | Unauthorized          | Authentication failed   |
| 403       | Forbidden             | Permission denied       |
| 404       | Not Found             | Record not found        |
| 409       | Conflict              | Duplicate key error     |
| 422       | Unprocessable Entity  | Validation error        |
| 500       | Internal Server Error | Database failure        |

Note: Always log full MongoDB errors internally and expose only sanitized error messages to API consumers.