

Comprehension Questions

CRUD Endpoints

1. What does the acronym CRUD stand for, and what is the purpose of CRUD endpoints in an API?
2. Which HTTP methods are typically used for each CRUD operation? List the method and the corresponding operation.
3. Why should you use nouns rather than verbs in RESTful endpoint URLs?
4. What status code should be returned when a new resource is successfully created, and what should the response body contain?
5. Describe the difference between a full update and a partial update of a resource. Which HTTP methods are used for each?
6. List two best practices for designing RESTful API endpoints.
7. What should happen if a client tries to access a resource that does not exist? What status code should be returned?
8. Why is it important to validate request data and handle errors gracefully in API endpoints?

Pagination in MongoDB

1. What problem does pagination solve in applications with large datasets?
2. Explain how the `skip()` and `limit()` methods are used together to implement pagination in MongoDB.
3. If you want to display page 4 with a page size of 15, how many documents should you skip, and what would the query look like?
4. Why is it recommended to create indexes on fields used for filtering or sorting in paginated queries?
5. What is a potential drawback of using `skip()` with large offsets, and what alternative technique is suggested for better performance?
6. How can you count the total number of documents that match a filter to provide pagination metadata to users?
7. Give an example of how you might add filtering to a paginated MongoDB query (e.g., retrieving only bookings with status “booked”).

Error Handling

1. Why is robust error handling important in API development, especially when interacting with services like MongoDB Atlas?
2. What information does the `ApiErrorObject` struct contain in the Atlas Go SDK?
3. How can you check for a specific HTTP status code (such as 404) when handling errors in the Atlas Go SDK?
4. What is the purpose of using custom error classes in error handling?
5. Give an example of how to mock an error for testing purposes in the Atlas Go SDK.
6. What is the difference between client errors (4xx) and server errors (5xx), and how should each be handled in an API?
7. List two best practices for error handling in API development