**Question 4 – MongoDB aggregation pipeline**

1. *Show how to use the MongoDB aggregation pipeline to query the data.*

**Using the MongoDB Aggregation Pipeline**

Is a sequence of stages, each taking the output of the previous stage as its input and transforming the document stream.  
Common stages include:

$match

* Filters documents (like **find()**)

$project

* Shapes each document by including, excluding, or computing fields.

EXAMPLE: all books by a given author

* **$match** filters documents whose authors array contains the given ObjectId.
* **$project** returns only the book’s title and isbn.

$unwind

* Deconstructs an array field into multiple documents.

$group

* Groups documents by a specified key and applies accumulator expressions **($sum, $avg,** etc.).

EXAMPLE: average page count by genre

* **$unwind** flattens the **genres** array.
* **$group** computes the average **page\_count** for each genre.

$lookup

* Performs a left-outer join with another collection.

EXAMPLE: merges each **order** with its matching **customer** document

$graphLookup

* Recursively joins documents (useful for tree- or graph-structured data).

EXAMPLE: for each employee, it will recursively follow their **manager\_id** links up the chain

* Hany for tree-pattern queries

1. *Compare selected queries from your 3 assignments.*

**Comparing Selected Queries Across Assignments**

Explain code

the aggregation pipeline replaces JOIN + GROUP BY in SQL

with a sequence of stages ($lookup, $unwind, $group, etc.),

in GraphQL/Neo4j you often rely on automatic resolvers or custom @cypher directives to achieve the same end.