// Step 1: Use the appropriate database

use Assignment5;

// Step 2: Create the books collection and insert sample data

db.books.drop(); // Drop the collection if it exists for a fresh start

// Insert sample data into the books collection

db.books.insertMany([

{ title: "Book A", author: "Author 1", genre: "Fiction", price: 15.99, published\_date: new Date("2022-01-10") },

{ title: "Book B", author: "Author 2", genre: "Non-Fiction", price: 12.99, published\_date: new Date("2021-05-15") },

{ title: "Book C", author: "Author 3", genre: "Mystery", price: 20.00, published\_date: new Date("2020-07-20") },

{ title: "Book D", author: "Author 1", genre: "Fiction", price: 25.00, published\_date: new Date("2023-02-10") },

{ title: "Book E", author: "Author 4", genre: "Sci-Fi", price: 18.50, published\_date: new Date("2019-08-30") },

{ title: "Book F", author: "Author 2", genre: "Non-Fiction", price: 30.00, published\_date: new Date("2020-03-25") },

{ title: "Book G", author: "Author 3", genre: "Mystery", price: 22.00, published\_date: new Date("2022-10-15") },

{ title: "Book H", author: "Author 5", genre: "Sci-Fi", price: 27.99, published\_date: new Date("2021-12-05") },

{ title: "Book I", author: "Author 1", genre: "Fiction", price: 19.99, published\_date: new Date("2018-11-20") },

{ title: "Book J", author: "Author 6", genre: "Non-Fiction", price: 15.00, published\_date: new Date("2017-06-12") }

]);

// Step 3: Perform aggregation operations

// 1. Find the average price of all books

print("1. Average Price of All Books:");

db.books.aggregate([

{

$group: {

\_id: null,

averagePrice: { $avg: "$price" }

}

}

]).pretty();

// 2. Find the count of books in each genre

print("2. Count of Books in Each Genre:");

db.books.aggregate([

{

$group: {

\_id: "$genre",

count: { $sum: 1 }

}

},

{

$sort: { count: -1 } // Optional: sort by count in descending order

}

]).pretty();

// 3. For each genre, find the most expensive book

print("3. Most Expensive Book in Each Genre:");

db.books.aggregate([

{

$group: {

\_id: "$genre",

mostExpensiveBook: { $max: { price: "$price", title: "$title" } }

}

}

]).pretty();

// 4. Find the authors who have written the maximum number of books

print("4. Authors Who Have Written the Maximum Number of Books:");

db.books.aggregate([

{

$group: {

\_id: "$author",

count: { $sum: 1 }

}

},

{

$sort: { count: -1 }

},

{

$limit: 1 // Optional: Get the author with the maximum count only

}

]).pretty();

// 5. Sort the books by published\_date in descending order

print("5. Books Sorted by Published Date in Descending Order:");

db.books.find().sort({ published\_date: -1 }).pretty();

// 6. Sort the price in ascending order

print("6. Books Sorted by Price in Ascending Order:");

db.books.find().sort({ price: 1 }).pretty();

// 7. Create an index on the title of the book

print("7. Creating Index on Title of the Book:");

db.books.createIndex({ title: 1 });

// Describe the index details

print("Index Details:");

db.books.getIndexes();