

Library Of CRUD

1. Create Operation:

- Create a new database named "library" and switch to it.
- Create a collection named "books" with fields: title, author, and published_year.
- Insert a document into the "books" collection with the details of your favorite book.

Solution:

```
test> use library;
switched to db library
library> db.createCollection("books");
{ ok: 1 }
library> db.books.insertOne({
...   title: "The Silent Patient",
...   author: "Alex Michaelides",
...   published_year: 2019
... });
{
  acknowledged: true,
  insertedId: ObjectId('676c0cdd9a978b3fa8893bf8')
}
```

2.Read Operation:

- Retrieve all documents from the "books" collection.
- Find and display only the documents where the author is "J.K. Rowling".
- Fetch and display the document with the earliest published year.

Solution:

```
library> db.books.find().toArray();
[
  {
    _id: ObjectId('676c0cdd9a978b3fa8893bf8'),
    title: 'The Silent Patient',
    author: 'Alex Michaelides',
    published_year: 2019
  }
]
library> db.books.find({ author: "J.K. Rowling" }).toArray();
[]
library> db.books.find().sort({ published_year: 1 }).limit(1);
[
  {
    _id: ObjectId('676c0cdd9a978b3fa8893bf8'),
    title: 'The Silent Patient',
    author: 'Alex Michaelides',
    published_year: 2019
  }
]
```

3. Update Operation:

- Update the published year of the book with the title "The Catcher in the Rye" to the current year.
- Add a new field "genre" with the value "Mystery" to all documents in the "books" collection.

Solution:

```
library> db.books.insertOne({
...   title: "The Catcher in the Rye",
...   author: "J.D. Salinger",
...   published_year: 1951
... });
{
  acknowledged: true,
  insertedId: ObjectId('676c0f3a9a978b3fa8893bfa')
}
library>

library> db.books.updateOne(
...   { title: "The Catcher in the Rye" },
...   { $set: { published_year: new Date().getFullYear() } }
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
library>

library> db.books.updateMany(
...   {},
...   { $set: { genre: "Mystery" } }
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 1,
  upsertedCount: 0
}
library> |
```

4. delete Operation:

- Remove the document with the title "1984" from the "books" collection.
- Delete all documents from the "books" collection where the published year is before 2000.

Solution:

```
library> db.books.deleteOne(
...   { title: "1984" }
... );
{ acknowledged: true, deletedCount: 0 }
library>

library> db.books.deleteMany(
...   { published_year: { $lt: 2000 } }
... );
{ acknowledged: true, deletedCount: 1 }
library> |
```

5. Advanced Query:

- Find and display the top 3 recently published books from the "books" collection.
- Retrieve documents from the "books" collection where the title contains the word "MongoDB" or "NoSQL".

```
library> db.books.find().sort({ published_year: -1 }).limit(3);
[
  {
    _id: ObjectId('676c0ed39a978b3fa8893bf9'),
    title: 'The Catcher in the Rye',
    author: 'J.D. Salinger',
    published_year: 2024,
    genre: 'Mystery'
  },
  {
    _id: ObjectId('676c0cdd9a978b3fa8893bf8'),
    title: 'The Silent Patient',
    author: 'Alex Michaelides',
    published_year: 2019,
    genre: 'Mystery'
  }
]
library> db.books.find(
...   title: { $regex: "MongoDB|NoSQL", $options: "i" }
... });
```