Poyraz Güler 30971

Bensu Gümüş 28870

This report outlines the implementation of a CafeShop Python application for managing menu items and customer reviews in a NoSQL database using MongoDB Atlas. The system allows users to interact with the database through a terminal-based interface, enabling users creating collections, reading data, inserting data, deleting data, and updating data.

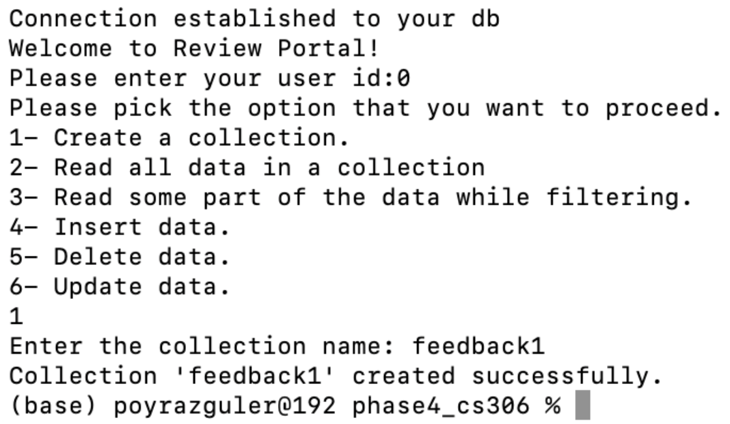
Implementation Details

The application uses the `pymongo` library to connect to MongoDB Atlas.

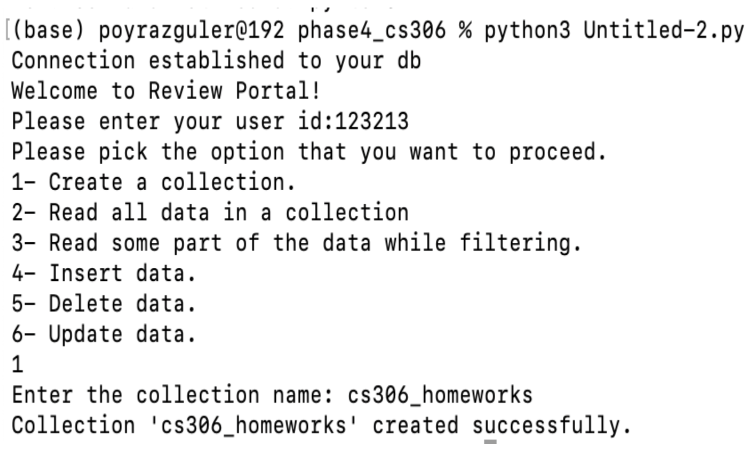
Functions are defined for each CRUD (Create, Read, Update, Delete) operation. The operations are performed on different collections: "Waiters", "MenuItems" and "Reviews." for menu items and service.

The main function serves as the entry point for user interaction. It shows the user with a menu where they may select from several options like reading, writing, updating, removing, and creating a collection.

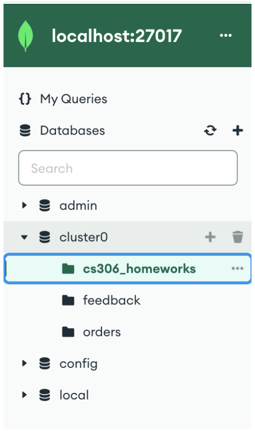
We can create a new collection in database



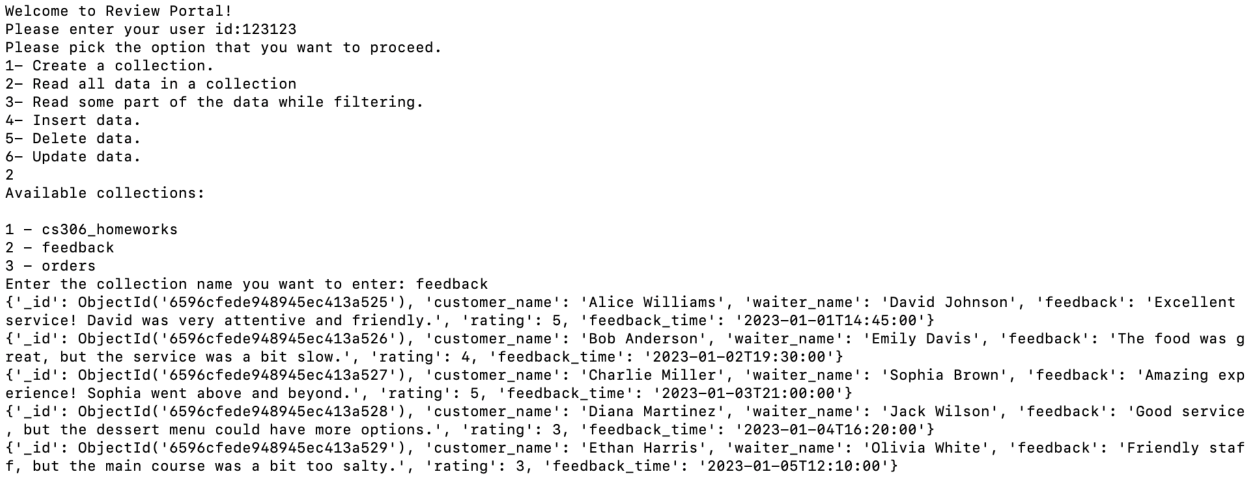
```
Connection established to your db
Welcome to Review Portal!
Please enter your user id:0
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
1
Enter the collection name: feedback1
Collection 'feedback1' created successfully.
(base) poyrazguler@192 phase4_cs306 % 
```



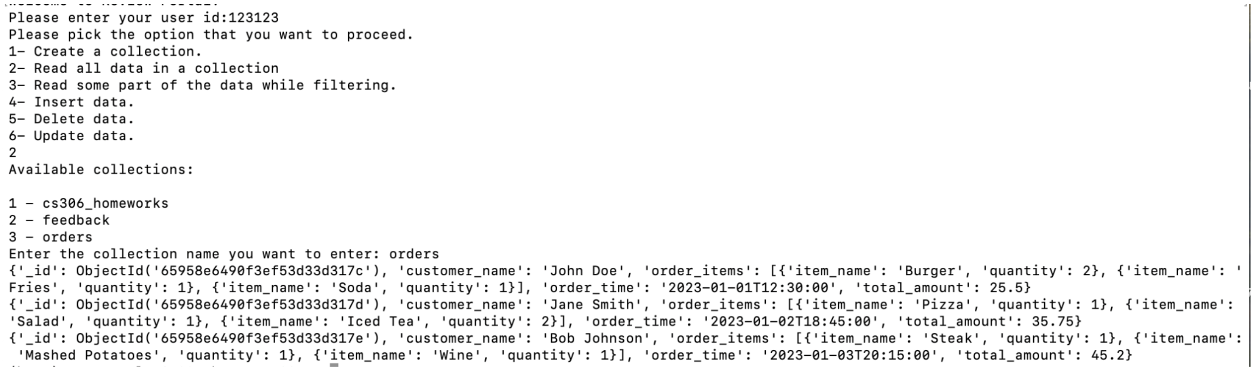
```
(base) poyrazguler@192 phase4_cs306 % python3 Untitled-2.py
Connection established to your db
Welcome to Review Portal!
Please enter your user id:123213
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
1
Enter the collection name: cs306_homeworks
Collection 'cs306_homeworks' created successfully.
```



Secondly we can read all the data from our previously created databases.



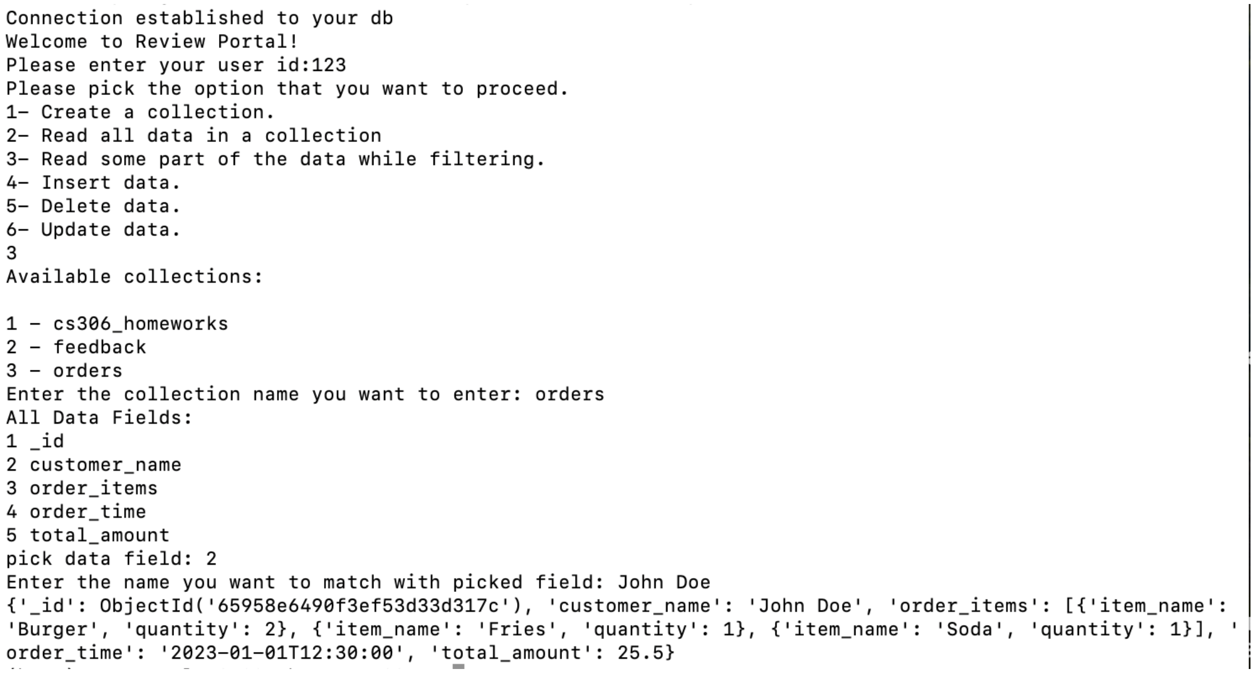
```
Welcome to Review Portal!
Please enter your user id:123123
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
2
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to enter: feedback
{'_id': ObjectId('6596cfede948945ec413a525'), 'customer_name': 'Alice Williams', 'waiter_name': 'David Johnson', 'feedback': 'Excellent
service! David was very attentive and friendly.', 'rating': 5, 'feedback_time': '2023-01-01T14:45:00'}
{'_id': ObjectId('6596cfede948945ec413a526'), 'customer_name': 'Bob Anderson', 'waiter_name': 'Emily Davis', 'feedback': 'The food was g
reat, but the service was a bit slow.', 'rating': 4, 'feedback_time': '2023-01-02T19:30:00'}
{'_id': ObjectId('6596cfede948945ec413a527'), 'customer_name': 'Charlie Miller', 'waiter_name': 'Sophia Brown', 'feedback': 'Amazing exp
erience! Sophia went above and beyond.', 'rating': 5, 'feedback_time': '2023-01-03T21:00:00'}
{'_id': ObjectId('6596cfede948945ec413a528'), 'customer_name': 'Diana Martinez', 'waiter_name': 'Jack Wilson', 'feedback': 'Good service
, but the dessert menu could have more options.', 'rating': 3, 'feedback_time': '2023-01-04T16:20:00'}
{'_id': ObjectId('6596cfede948945ec413a529'), 'customer_name': 'Ethan Harris', 'waiter_name': 'Olivia White', 'feedback': 'Friendly staf
f, but the main course was a bit too salty.', 'rating': 3, 'feedback_time': '2023-01-05T12:10:00'}
```



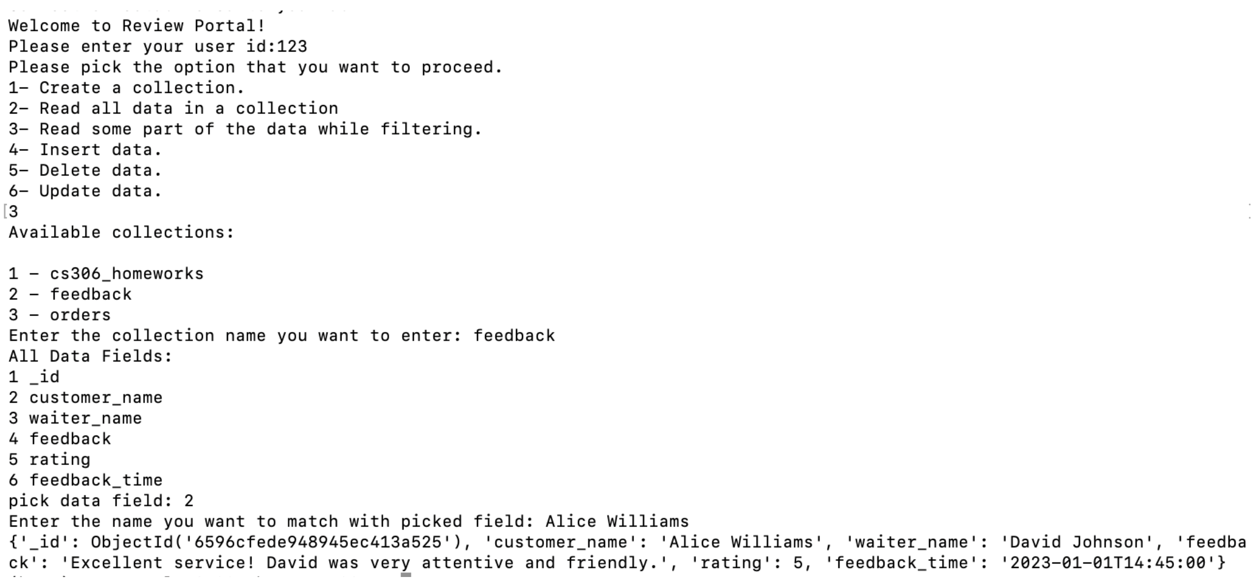
```
Please enter your user id:123123
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
2
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to enter: orders
{'_id': ObjectId('65958e6490f3ef53d33d317c'), 'customer_name': 'John Doe', 'order_items': [{'item_name': 'Burger', 'quantity': 2}, {'item_name': '
Fries', 'quantity': 1}, {'item_name': 'Soda', 'quantity': 1}], 'order_time': '2023-01-01T12:30:00', 'total_amount': 25.5}
{'_id': ObjectId('65958e6490f3ef53d33d317d'), 'customer_name': 'Jane Smith', 'order_items': [{'item_name': 'Pizza', 'quantity': 1}, {'item_name':
'Salad', 'quantity': 1}, {'item_name': 'Iced Tea', 'quantity': 2}], 'order_time': '2023-01-02T18:45:00', 'total_amount': 35.75}
{'_id': ObjectId('65958e6490f3ef53d33d317e'), 'customer_name': 'Bob Johnson', 'order_items': [{'item_name': 'Steak', 'quantity': 1}, {'item_name':
 'Mashed Potatoes', 'quantity': 1}, {'item_name': 'Wine', 'quantity': 1}], 'order_time': '2023-01-03T20:15:00', 'total_amount': 45.2}
```

Third option is to display some of the information based on a query

In our system we ask which field the user wants to run the query on. The query we provide checks for the equality of values. So if user picks customer name and enters a name we provide all documents which have the customer name that the user provided.

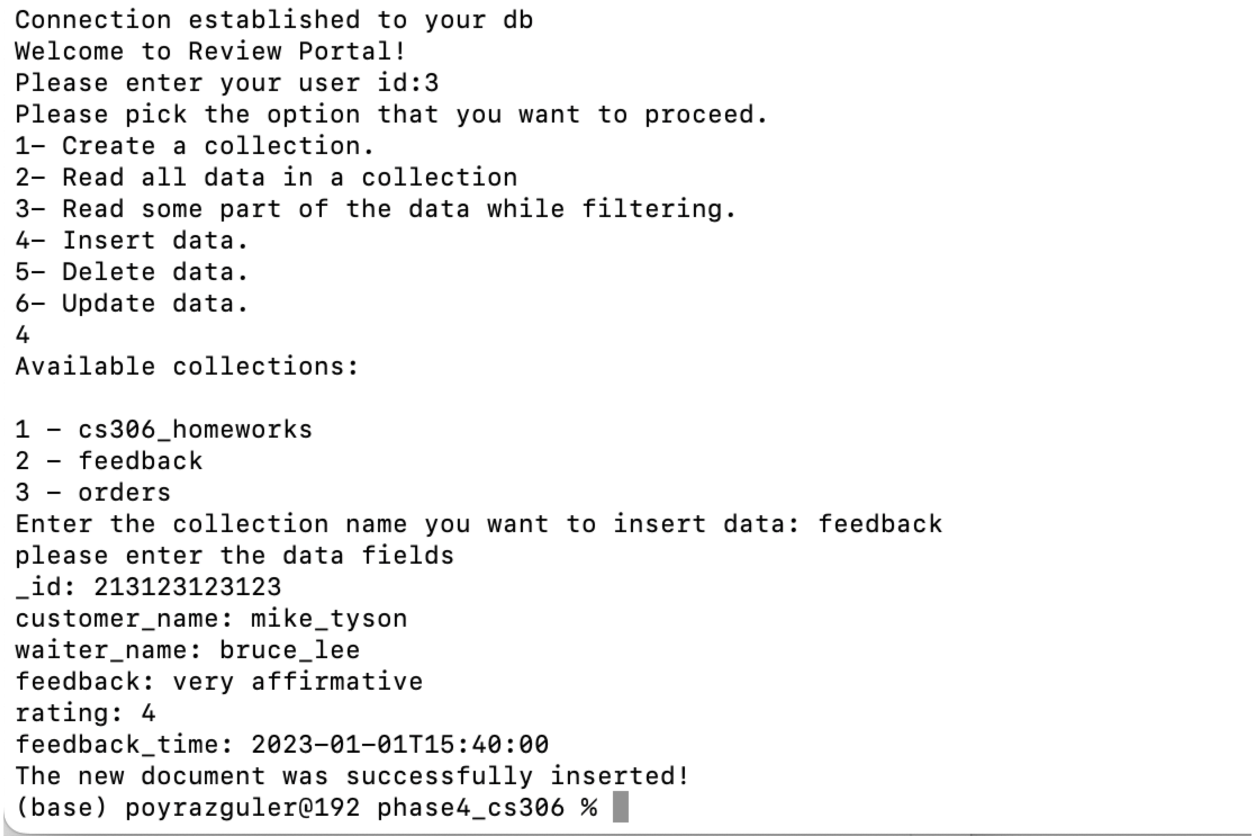


```
Connection established to your db
Welcome to Review Portal!
Please enter your user id:123
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
3
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to enter: orders
All Data Fields:
1 _id
2 customer_name
3 order_items
4 order_time
5 total_amount
pick data field: 2
Enter the name you want to match with picked field: John Doe
{'_id': ObjectId('65958e6490f3ef53d33d317c'), 'customer_name': 'John Doe', 'order_items': [{'item_name':
'Burger', 'quantity': 2}, {'item_name': 'Fries', 'quantity': 1}, {'item_name': 'Soda', 'quantity': 1}], '
order_time': '2023-01-01T12:30:00', 'total_amount': 25.5}
```

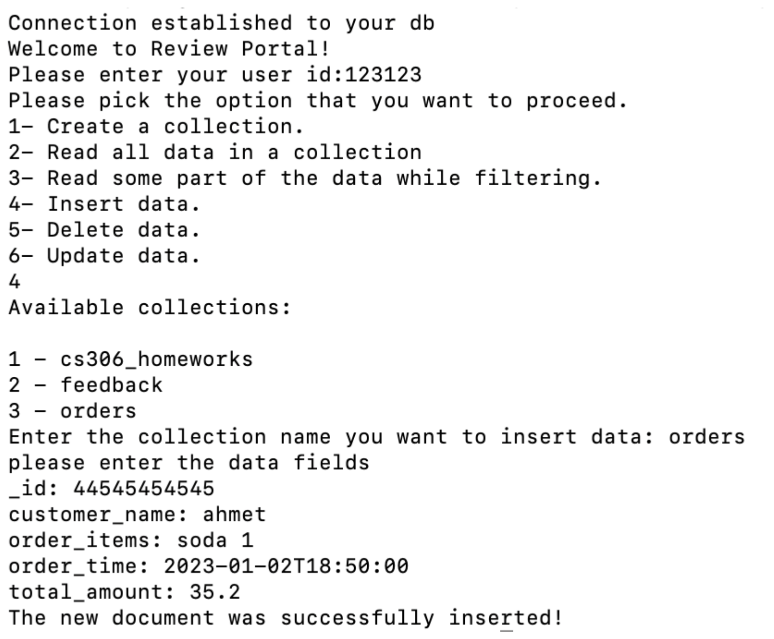


```
Welcome to Review Portal!
Please enter your user id:123
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
3
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to enter: feedback
All Data Fields:
1 _id
2 customer_name
3 waiter_name
4 feedback
5 rating
6 feedback_time
pick data field: 2
Enter the name you want to match with picked field: Alice Williams
{'_id': ObjectId('6596cfede948945ec413a525'), 'customer_name': 'Alice Williams', 'waiter_name': 'David Johnson', 'feedba
ck': 'Excellent service! David was very attentive and friendly.', 'rating': 5, 'feedback_time': '2023-01-01T14:45:00'}
```

The Fourth option asks for values for already existing fields based on the collection you want to insert a document to. After receiving these information. A document is added to the collection.

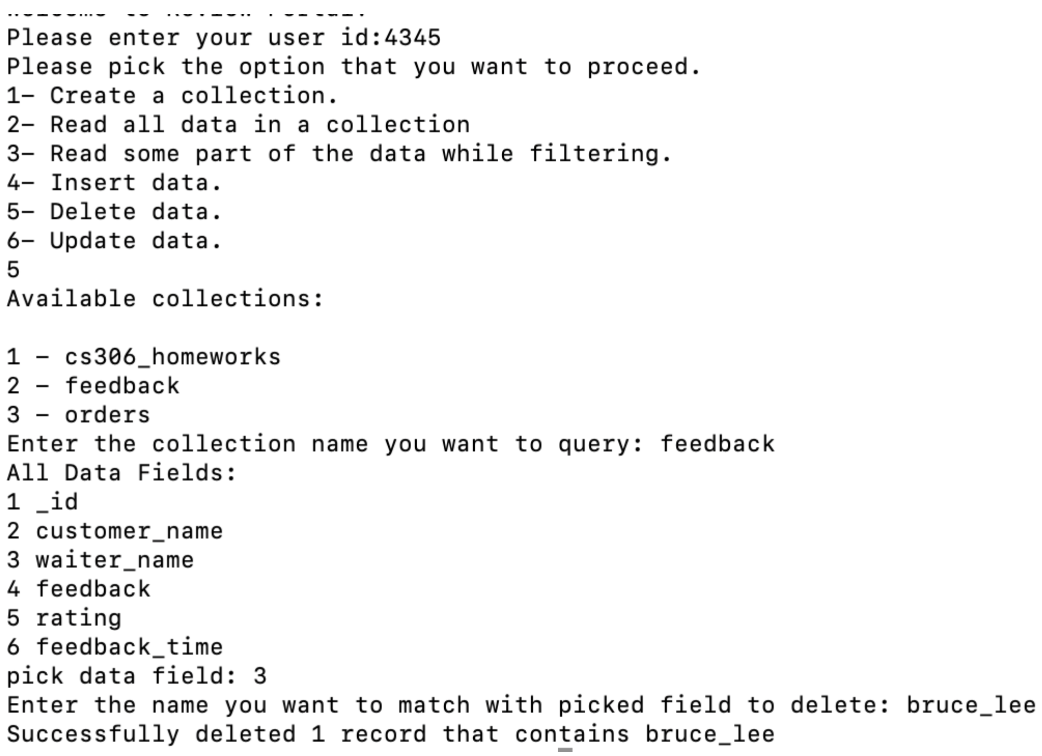


```
Connection established to your db
Welcome to Review Portal!
Please enter your user id:3
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
4
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to insert data: feedback
please enter the data fields
_id: 213123123123
customer_name: mike_tyson
waiter_name: bruce_lee
feedback: very affirmative
rating: 4
feedback_time: 2023-01-01T15:40:00
The new document was successfully inserted!
(base) poyrazguler@192 phase4_cs306 % 
```

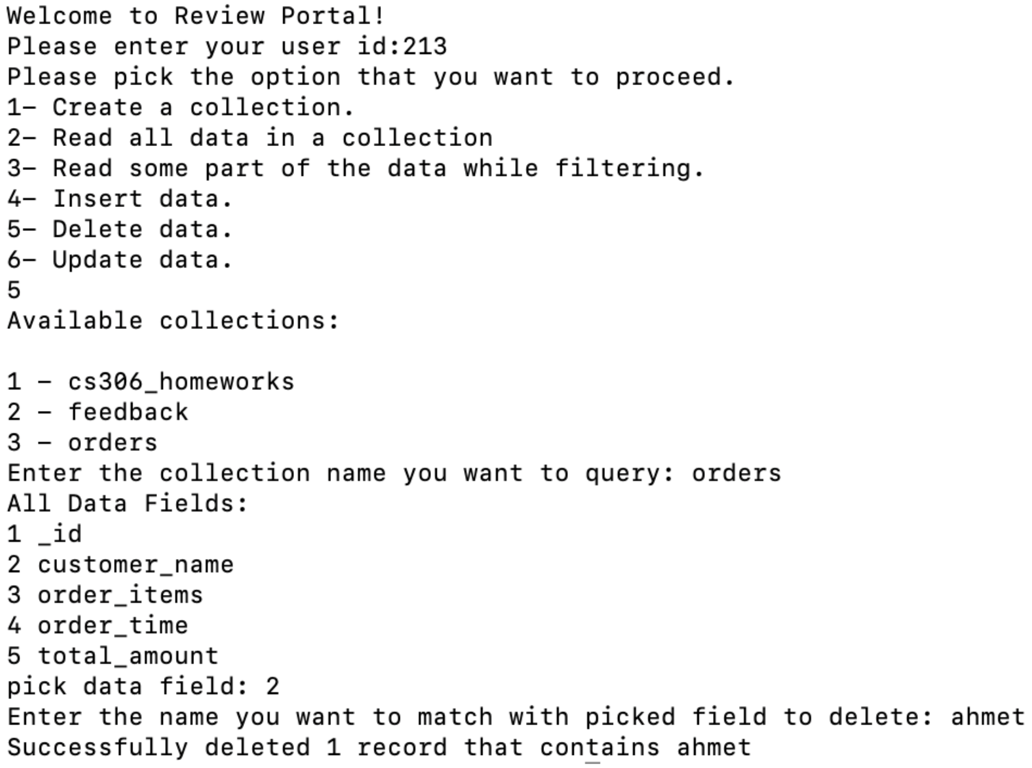


```
Connection established to your db
Welcome to Review Portal!
Please enter your user id:123123
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
4
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to insert data: orders
please enter the data fields
_id: 44545454545
customer_name: ahmet
order_items: soda 1
order_time: 2023-01-02T18:50:00
total_amount: 35.2
The new document was successfully inserted!
```

Fifth option is to Delete Data based on a query. We ask the user which field s/he want to run the query on? Then based on the input we delete all records that fit.

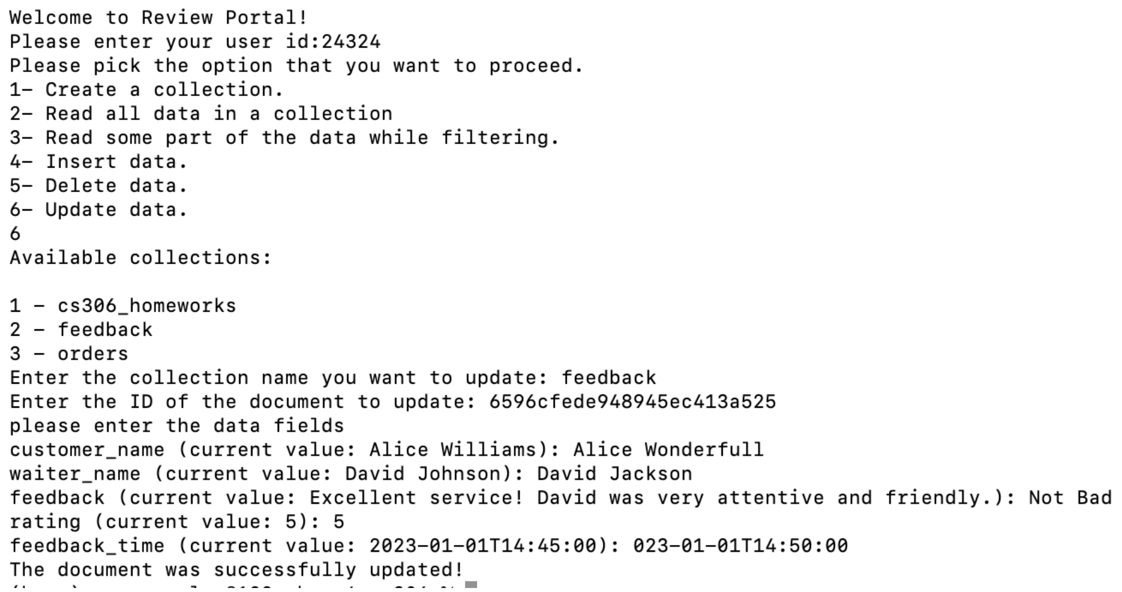


```
Welcome to Review Portal!
Please enter your user id:4345
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
5
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to query: feedback
All Data Fields:
1 _id
2 customer_name
3 waiter_name
4 feedback
5 rating
6 feedback_time
pick data field: 3
Enter the name you want to match with picked field to delete: bruce_lee
Successfully deleted 1 record that contains bruce_lee
```

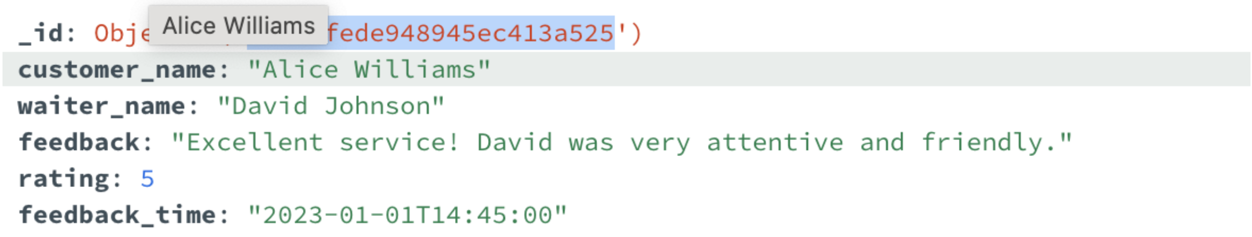


```
Welcome to Review Portal!
Please enter your user id:213
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
5
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to query: orders
All Data Fields:
1 _id
2 customer_name
3 order_items
4 order_time
5 total_amount
pick data field: 2
Enter the name you want to match with picked field to delete: ahmet
Successfully deleted 1 record that contains ahmet
```

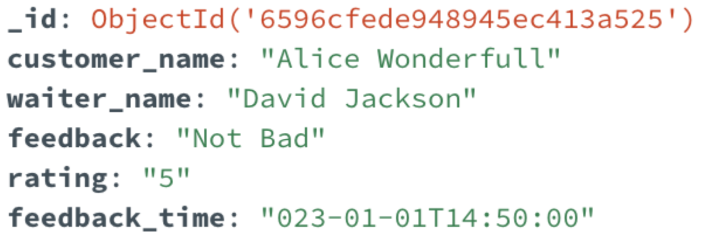
In option 6 we expect user to pick which document to update based on the Id s/he provides. Then we ask the user to provide new values to these fields.



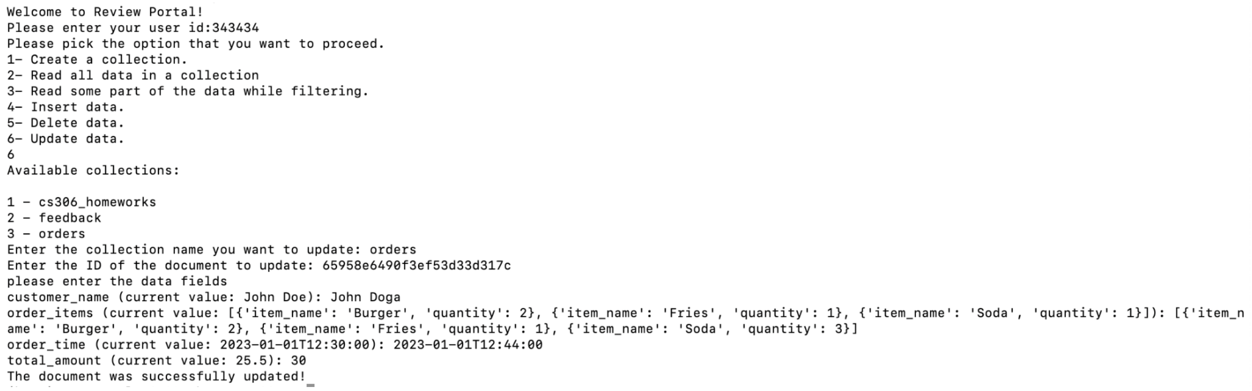
```
Welcome to Review Portal!
Please enter your user id:24324
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
6
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to update: feedback
Enter the ID of the document to update: 6596cfede948945ec413a525
please enter the data fields
customer_name (current value: Alice Williams): Alice Wonderfull
waiter_name (current value: David Johnson): David Jackson
feedback (current value: Excellent service! David was very attentive and friendly.): Not Bad
rating (current value: 5): 5
feedback_time (current value: 2023-01-01T14:45:00): 023-01-01T14:50:00
The document was successfully updated!
```



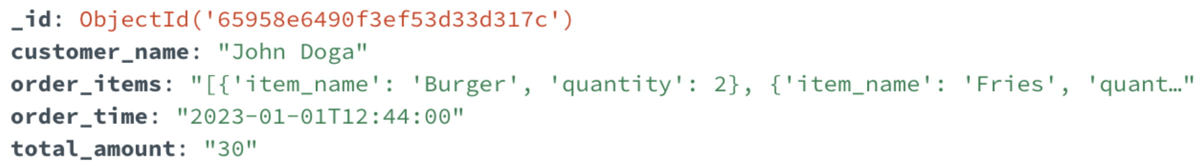
```
_id: Obje                fede948945ec413a525')
customer_name: "Alice Williams"
waiter_name: "David Johnson"
feedback: "Excellent service! David was very attentive and friendly."
rating: 5
feedback_time: "2023-01-01T14:45:00"
```



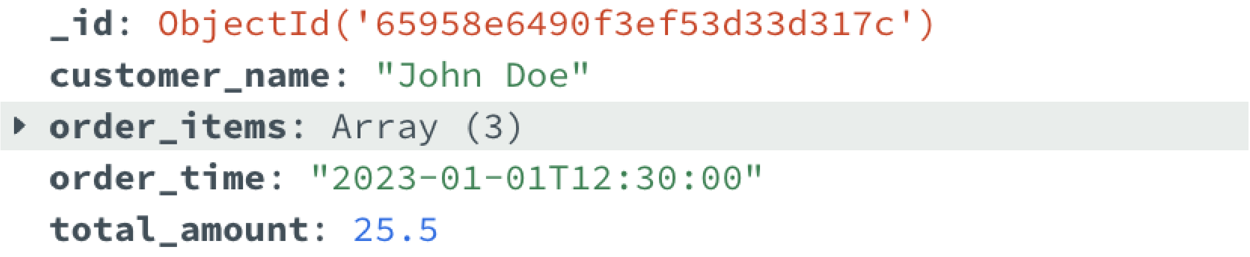
```
_id: ObjectId('6596cfede948945ec413a525')
customer_name: "Alice Wonderfull"
waiter_name: "David Jackson"
feedback: "Not Bad"
rating: "5"
feedback_time: "023-01-01T14:50:00"
```



```
Welcome to Review Portal!
Please enter your user id:343434
Please pick the option that you want to proceed.
1- Create a collection.
2- Read all data in a collection
3- Read some part of the data while filtering.
4- Insert data.
5- Delete data.
6- Update data.
6
Available collections:

1 - cs306_homeworks
2 - feedback
3 - orders
Enter the collection name you want to update: orders
Enter the ID of the document to update: 65958e6490f3ef53d33d317c
please enter the data fields
customer_name (current value: John Doe): John Doga
order_items (current value: [{'item_name': 'Burger', 'quantity': 2}, {'item_name': 'Fries', 'quantity': 1}, {'item_name': 'Soda', 'quantity': 1}]): [{'item_n
ame': 'Burger', 'quantity': 2}, {'item_name': 'Fries', 'quantity': 1}, {'item_name': 'Soda', 'quantity': 3}]
order_time (current value: 2023-01-01T12:30:00): 2023-01-01T12:44:00
total_amount (current value: 25.5): 30
The document was successfully updated!
```



```
_id: ObjectId('65958e6490f3ef53d33d317c')
customer_name: "John Doga"
order_items: "[{'item_name': 'Burger', 'quantity': 2}, {'item_name': 'Fries', 'quant…"
order_time: "2023-01-01T12:44:00"
total_amount: "30"
```



```
_id: ObjectId('65958e6490f3ef53d33d317c')
customer_name: "John Doe"
▸ order_items: Array (3)
order_time: "2023-01-01T12:30:00"
total_amount: 25.5
```

In Conclusion, this Python application provides a terminal-based interface for users to interact with a NoSQL database hosted on MongoDB Atlas, covering essential CRUD operations on collections related to menu items, waiters and customer reviews