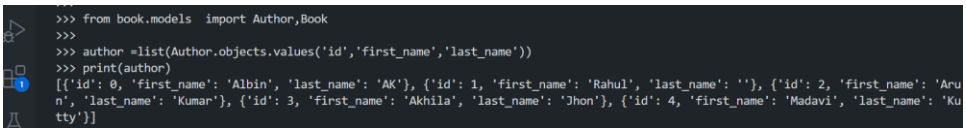
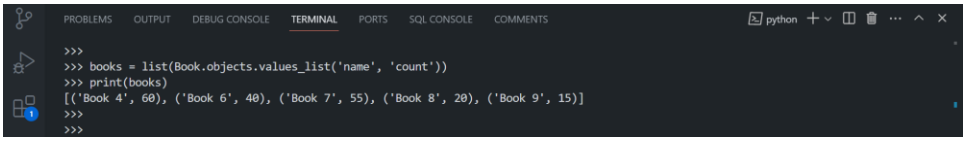
Note - avoid N+1 query problems 1. Using the shell plus and the django project you created before, run these queries to:

Fetch all the objects of authors with only id, first name and last name

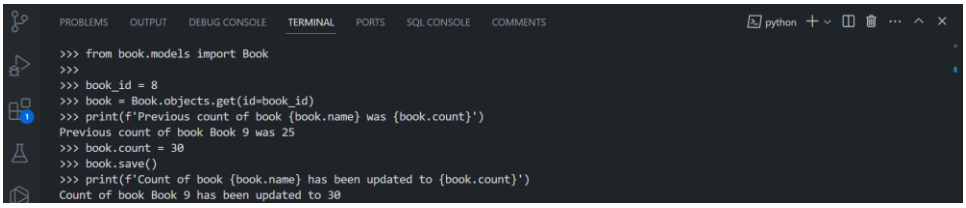


Fetch all the objects of books as a list that contains name

Fetch all the objects of books as a tuple that contains name and count



Fetch an object from books table using id and update the count.



Fetch an object from the books using name and update the author of that book

```
>>> from book.models import Book, Author
>>>
>>> book_name = "Book 4"
>>> try:
...
...     book = Book.objects.get(name=book_name)
...     new_author_id = 3
...     book.author_id= new_author_id
...     book.save()
...     print(f"The author of the book '{book_name}' has been updated to '{ new_author_id}'.")
... except Book.DoesNotExist:
...     print(f"Error: Book '{book_name}' does not exist.")
... except Author.DoesNotExist:
...     print(f"Error: Author '{book.author_id}' does not exist.")
... except Exception as e:
...     print(f"An error occurred: {e}")
...
The author of the book 'Book 4' has been updated to '3'.
```

Fetch all books from the table and show the name of the books and full name of authors as a list of dictionaries

```
>>> from book.models import Book
>>>
>>> books_with_authors = Book.objects.values('name', 'author__first_name', 'author__last_name')
>>>
>>> for book in books_with_authors:
...     author_name = f"{book['author__first_name']} {book['author__last_name']}"
...     book['author name'] = author_name
...     del book['author__first_name']
...     del book['author__last_name']
...
>>> books_list = list(books_with_authors)
>>>
>>> print(books_list)
[{'name': 'Book 4', 'author name': 'Akhila Jhon'}, {'name': 'Book 6', 'author name': 'Rahul '}, {'name': 'Book 7', 'author name': 'Akhi
la Jhon'}, {'name': 'Book 8', 'author name': 'Akhila Jhon'}, {'name': 'Book 9', 'author name': 'Madavi Kutty'}]
```

Fetch all the authors from the table and show the name of authors with the list of names of the books of that author in a list of dictionaries format

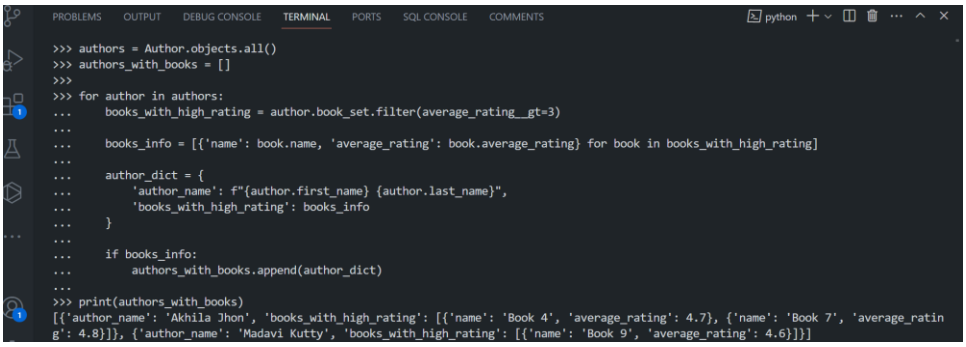
```
>>> from book.models import Author
>>>
>>> authors = Author.objects.all()
>>> authors_with_books = []
>>> for author in authors:
...
...     book_names = list(author.book_set.values_list('name', flat=True))
...     author_dict = {
...         'author_name': f"{author.first_name} {author.last_name}",
...         'books': book_names
...     }
...
...
...     authors_with_books.append(author_dict)
...
>>> print(authors_with_books)
[{'author_name': 'Albin AK', 'books': []}, {'author_name': 'Rahul ', 'books': ['Book 6']}, {'author_name': 'Arun Kumar', 'books': []},
{'author_name': 'Akhila Jhon', 'books': ['Book 4', 'Book 7', 'Book 8']}, {'author_name': 'Madavi Kutty', 'books': ['Book 9']}]
```

Fetch all the books from the table and show the name, price and count of the books in a list of dictionary format inside the queryset output

```
>>> from book.models import Book
>>>
>>> books = Book.objects.values('name', 'price', 'count')
>>> books_list = list(books)
>>> print(books_list)
[{'name': 'Book 4', 'price': Decimal('250.00'), 'count': 60}, {'name': 'Book 6', 'price': Decimal('350.00'), 'count': 40}, {'name': 'Bo
ok 7', 'price': Decimal('400.00'), 'count': 55}, {'name': 'Book 8', 'price': Decimal('450.00'), 'count': 23}, {'name': 'Book 9', 'price
': Decimal('500.00'), 'count': 30}]>>>
>>>
```

Fetch an author from the table and increase the price with 100 for all the books of that author

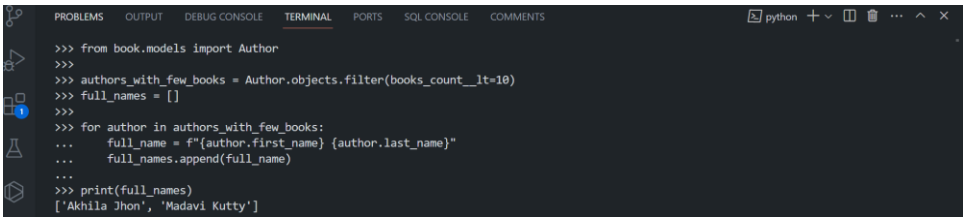
Fetch all the authors from the table and show the name of authors with the list of names of the books of that author that have an average rating greater than 3 in a list of dictionaries format



Fetch all the authors from the table and show the name of authors with the list of names of the books of that author that have an average rating less than or equal to 3 in a list of dictionaries format

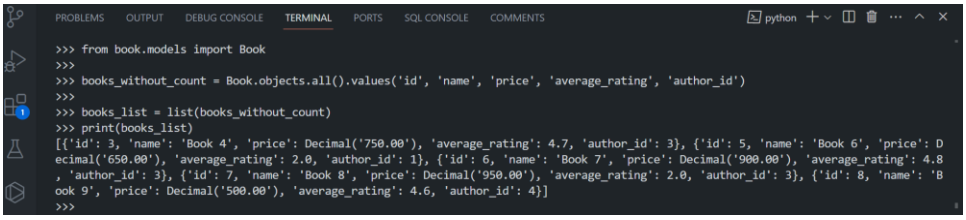
Fetch all the books from the table that has an average rating as 3

Fetch all the authors from the table that have books_count less than 10



Fetch all the authors from the table and update the books_count value with respect to the exact number of books connected to that author

Fetch all the books from the table and avoid count field while fetching the objects



Fetch all the books from the table and avoid count field while fetching the objects