Bookstore Inventory Management

Problem Statement

Create a stored procedure to manage the inventory of a bookstore. The stored procedure should be able to add new books, update the details of existing books, delete books, and retrieve the details of books based on specific criteria.

Requirements:

1. **Database Schema:**
    - **Table: Books**
        - BookID (INT, Primary Key, Auto Increment)
        - Title (VARCHAR(255))
        - Author (VARCHAR(255))
        - Genre (VARCHAR(100))
        - Price (DECIMAL(10, 2))
        - Stock (INT)
        - PublishedDate (DATE)
2. **Stored Procedure: ManageBooks**
    - **Parameters:**
        - @Action (VARCHAR(10)): The action to be performed ('ADD', 'UPDATE', 'DELETE', 'GET')
        - @BookID (INT): The ID of the book (required for 'UPDATE' and 'DELETE' actions)
        - @Title (VARCHAR(255)): The title of the book (required for 'ADD' and 'UPDATE' actions)
        - @Author (VARCHAR(255)): The author of the book (required for 'ADD' and 'UPDATE' actions)
        - @Genre (VARCHAR(100)): The genre of the book (required for 'ADD' and 'UPDATE' actions)
        - @Price (DECIMAL(10, 2)): The price of the book (required for 'ADD' and 'UPDATE' actions)
        - @Stock (INT): The stock of the book (required for 'ADD' and 'UPDATE' actions)
        - @PublishedDate (DATE): The published date of the book (required for 'ADD' and 'UPDATE' actions)

- @SearchTerm (VARCHAR(255)): The search term for retrieving books (required for 'GET' action)

- **Functionality:**
  - **ADD:** Insert a new book record into the Books table with the provided details.
  - **UPDATE:** Update the existing book record identified by @BookID with the provided details.
  - **DELETE:** Delete the book record identified by @BookID.
  - **GET:** Retrieve the book records that match the @SearchTerm in the title or author fields.