**Activity 13:** Book Inventory Manager

**Objective:** Make a simple Python program to manage a book inventory for a library or bookstore.

**What Your Program Must Do:**  
Build a Python program to manage a book inventory with these features:

* **Add Book:**
  + Add a new book with a title, author name, date of publication (e.g., "2020"), and stock amount.
  + Example: "Harry Potter", "J.K. Rowling", "1997", 5 in stock.
  + Make sure date and stock only accept numbers (no letters like "abc").
* **Edit Book:**
  + Change the author, date, or stock of a book already in the inventory.
  + Example: Update "Harry Potter" to "J.K. Rowling", "1998", 10 in stock.
  + Make sure date and stock only accept numbers (no letters).
* **Delete Book:**
  + Remove a book from the inventory.
  + Example: Remove "Harry Potter".
  + Ask “Are you sure?” before deleting it.
* **Search Book:**
  + Search for books by asking if the user wants to use Author Name (AN) or Date of Publication (DoP).
  + Example for AN: Search "J.K. Rowling" and show all her books with title, date, and stock.
  + Example for DoP: Search "1997" and show all books from that year with title, author, and stock.
* **View All Books:**
  + Show every book in the inventory with title, author, date, and stock.
  + Example: "Harry Potter: J.K. Rowling, 1997, 5 left".

**How It Works:**

* Use a function called manage\_inventory().
* Show a menu with these choices:
  + 1 = Add Book
  + 2 = Edit Book
  + 3 = Delete Book
  + 4 = Search Book
  + 5 = View All Books
* Ask for a number (1-5) to pick an action.
* After each action, ask: “Do another action? (yes/no)”
  + “Yes” shows the menu again.
  + “No” stops the program.
* For Add, Edit, Delete, and Search, also ask right after: “Do this again? (yes/no)”
  + Example: “Search another book?”
  + “Yes” repeats that action.
  + “No” moves to “Do another action?”