ARRAY IMPLEMENTATION OF STACK

```
class Stack:
    def __init__(self):
        self.stack = []

    def push(self, book_title):
        """Add a book title to the stack."""
        self.stack.append(book_title)
        print(f"Book '{book_title}' added to the stack.")

    def pop(self):
        """Remove and return the top book title from the stack."""
        if self.is_empty():
            print("Stack is empty. No book to pop.")
            return None
        return self.stack.pop()

    def peek(self):
        """Return the top book title without removing it."""
        if self.is_empty():
            print("Stack is empty. No book to peek.")
            return None
        return self.stack[-1]

    def is_empty(self):
        """Check if the stack is empty."""
        return len(self.stack) == 0

    def display_stack(self):
        """Display all book titles in the stack."""
        if self.is_empty():
            print("Stack is empty.")
            return
        print("Current Stack of Books:")
        for title in reversed(self.stack):
            print(f"- {title}")

# Example Usage
if __name__ == "__main__":
    book_stack = Stack()

    # Adding books to the stack
    book_stack.push("The Catcher in the Rye")
    book_stack.push("To Kill a Mockingbird")
    book_stack.push("1984")
```

```
# Displaying the stack
book_stack.display_stack()

# Peeking at the top book
top_book = book_stack.peek()
if top_book:
    print(f"Top book on the stack: '{top_book}'")

# Removing books from the stack
print(f"Popped book: '{book_stack.pop()}'")
print(f"Popped book: '{book_stack.pop()}'")

# Displaying the stack after pops
book_stack.display_stack()

# Attempting to pop from an empty stack
book_stack.pop()
book_stack.pop()  # Should indicate stack is empty
```