JavaScript as an Object Oriented Programming Language

Note: Make your own interface for the following tasks.

1. **Create your own object that contains a list of numerical sequence, with the following details**
   - **Your constructor takes 3 parameters to define start, end of list and step**
   - **The list should be private and filled with private method**
   - **You can create getter and setter for the list if needed**
   - **Allow the user to apply the following functionality to his created sequence**
     - **Append or prepend a new value (you have to ensure that you are pushing sequential value otherwise through exception)**
     - **Dequeue or pop a value,**
     - **display all its content,**
   - **all of the properties should be defined using accessor descriptor, prevent them from being deleted, iterated or being modified.**
   - **you can add any property you need.**

2. **Display the area and perimeter of an object created from using Rectangle Constructor that has width and height properties and 2 extended methods (using prototype property) for calculating area, perimeter.**

   Override .toString() function to display a message declaring the width, height, area and perimeter of the created object.

   Create Class Property that counts numbers of created objects and Class method to retrieve it.

3. **Make proper updates in your previous code of generating Rectangle object that should inherit from Shape Constructor, and create your**

Square constructor that inherits from Rectangle.
Prevent creating any object from shape, allow creation of only rectangles and square
Bonus: allow creation of only one square and one rectangle

4. Create your box object that contains books objects, ensure that you can 1) count # of books inside box

2) delete any of these books in box according to book name or type.

3) create book object and add it to box object content property

4) use .toString() to tell its dimensions and how books are stored in it.

5) implement .valueof() so that if there is more than one box object we can get total books in these boxes by adding the i.e. box1 has 5 books while box2 has 2 books, box1 + box2 should return 7

Note:

- **there is no inheritance**
- **using of global variables, Class methods and properties isn't allowed.**
- **box object has the following properties:**
  height, width, length, numOfBooks, volume, material, content.
  The content property contains an array books
- **book object has the following properties:**
  title, numofChapters, author, numofPages, publisher, numofCopies
- **you can define any function needed for both box and book objects**