# Coding Challenge Week 6 Day 1

I want you peeps to design a stack operation in a UI pattern. I will define the pattern below. Before speaking about the implementation, I want to tell what stack is. A stack is a data structure, which stores data at the top of a rack, and the last element which is placed is the first to get accessed. It is like a stack of plates which you could see in a regular marriage function, where the top most plate is taken first by each person. In layman's terms, the data is added in a vertical fashion, and the top most item is being removed first. Now let us see how you should create it.

Each stack block should contain the following properties

1) Font size - 30px
2) Font - Roboto, Refer here on how to import 3rd party fonts.
3) Font color - red
4) Bold weight
5) Block width - 400px
6) Block height - 50px
7) Border - red 3px
8) Background - yellow
9) Margin top and bottom - 10px

**Please note that the above styles must be applied via JavaScript not by regular CSS.** I have taught this today.

The programming pattern must follow like below.

1) The user must keep on getting asked for a number.

2) Every time he/she enters that number, a new stack block must be added to the top. Say for example if the Stack shelf is empty now, and if user types the number 6, then a new stack block of all the above said styles should be created and placed above. Again if the

user enters the number 8, the stack block 8 should be placed above 6. I must view it visually.

3) If the user types in “enough” only then this infinite process be stopped. Refer here on knowing how to abruptly come out of a process.

4) There is this thing called “**Stack overflow”** which means that once our memory is full of stack blocks, it will scream an error. We will simulate the same here as well. **If the stack blocks are more than 10,** deny the creation of stack block by saying **“Sorry stack limit reached.”**, and come out of the process as well.

5) A button called POP should be present, which makes sure the top most rectangle gets removed from the DOM itself.

The stack shelf in entirety, should somehow look like this.