

## Observer Pattern continued

- Observer pattern is useful when you want to model behavior that happens multiple times. (e.g capture user input)
- The OS interrupts different events when something is pressed or triggered . (e.g a keyboard) This interrupts programs like node.js which interrupts the observer. When the observer gets interrupted by node.js, it puts the call in the libuv.
- If we want to listen to every keypress instead of waiting for a return key we need to switch to raw mode.
- Notice this also uses the Observer pattern, listening for keypress events.
- Once in raw, it is possible to listen to specific keypresses with the listener callback having two arguments:
- Keyboard types:
  - process.stdin
  - index

Quick reminder of the spread operator; You can use it in functions and you can use it with objects as well.

- This is usefull when passing parameters. It allows you to pass as many objects/functions in a syntax like this {function1, object1, etc..}