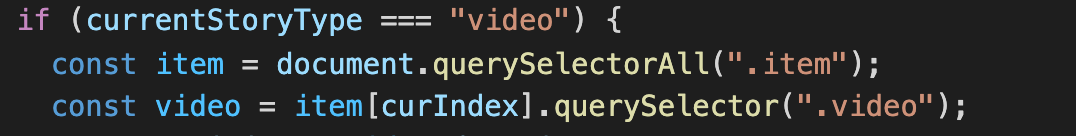
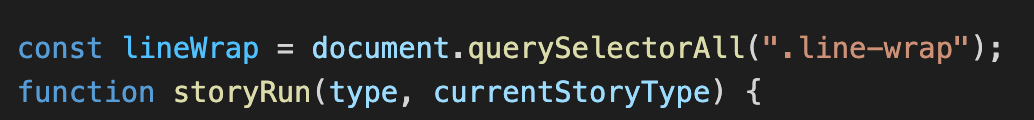
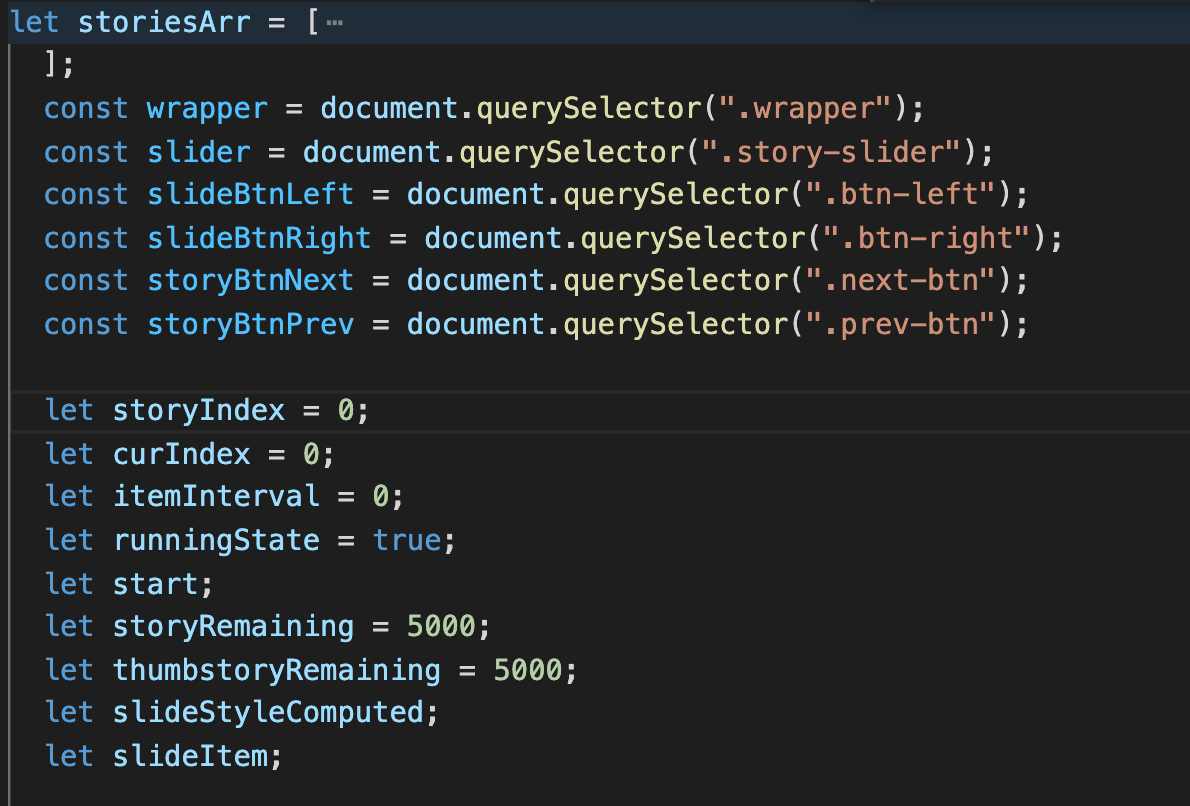
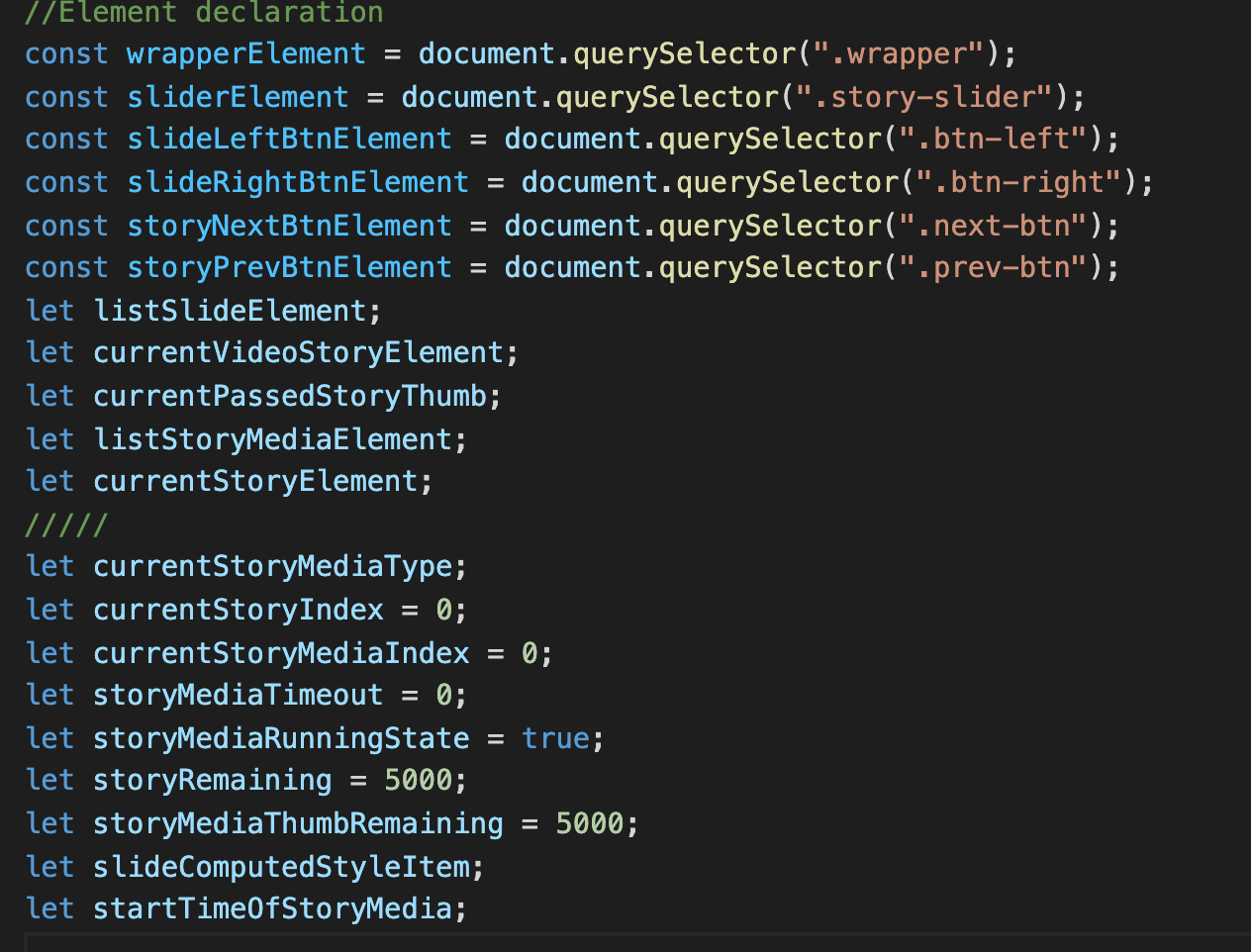
**Refactoring Old Assignment : “ Facebook Story ” assignment.**

Old code:



New code:



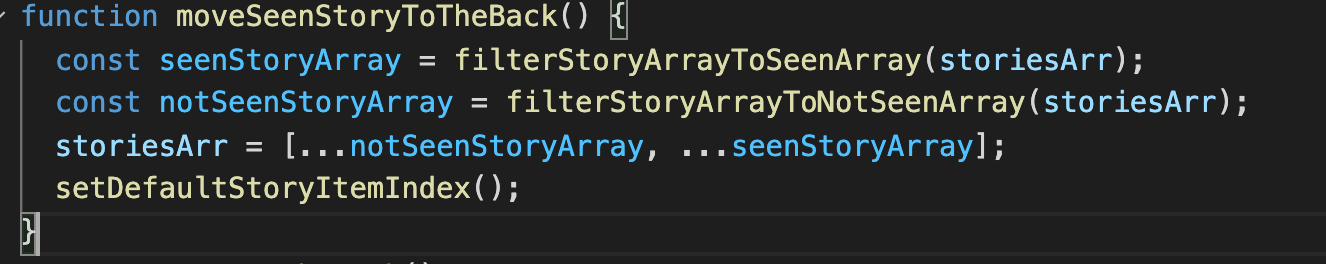
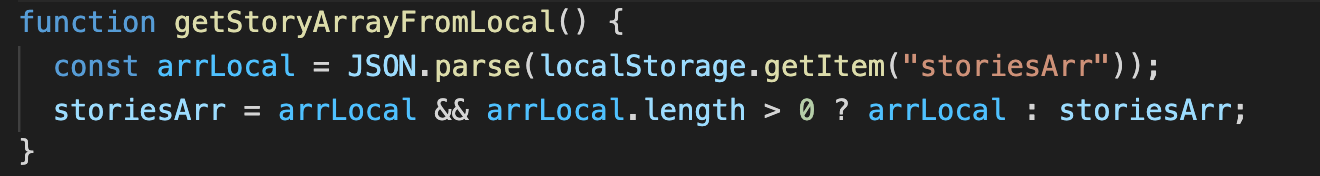
- In the old code i declared global variables in different places. It’s hard to find and the code look very messy. I move all global variables to the top to easy use ,reuse and I change the variable’s name to get more understandable about that variable meaning.

- I use “Element” to describe a element variable. So I can access exactly the variable, easier to code or debug.

Old code :



New code :



- This function i separate it to each function and name it a meaningful name so i can know what inside this function is doing.

Old code :

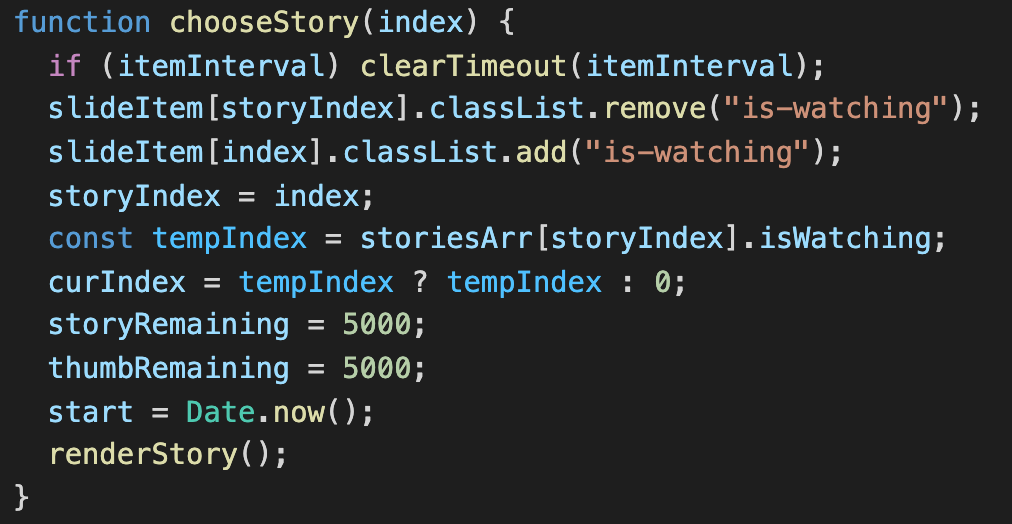


New code :

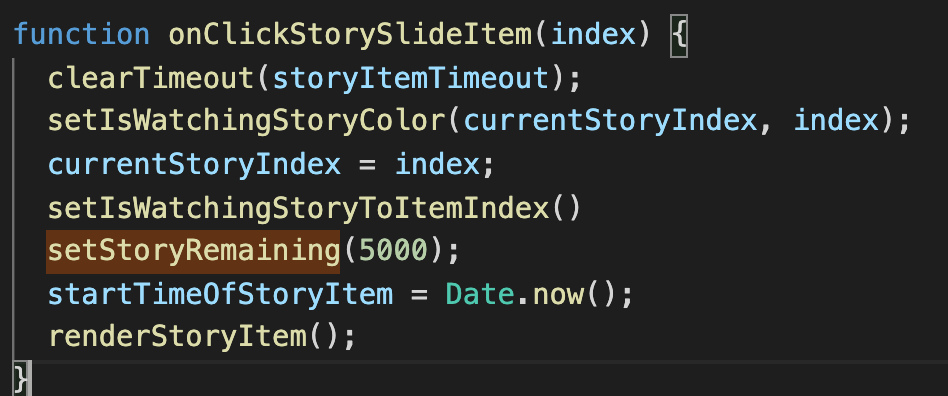


- These 2 function i change it to a more clearly name so we can know what this function does.

Old code :



New code :



- I change this function to a better name. Some lines in the function i use it multiple time.

To easier coding i wrap it into a function to reuse and then i delete some redundancy condition.

Old code :



New code :

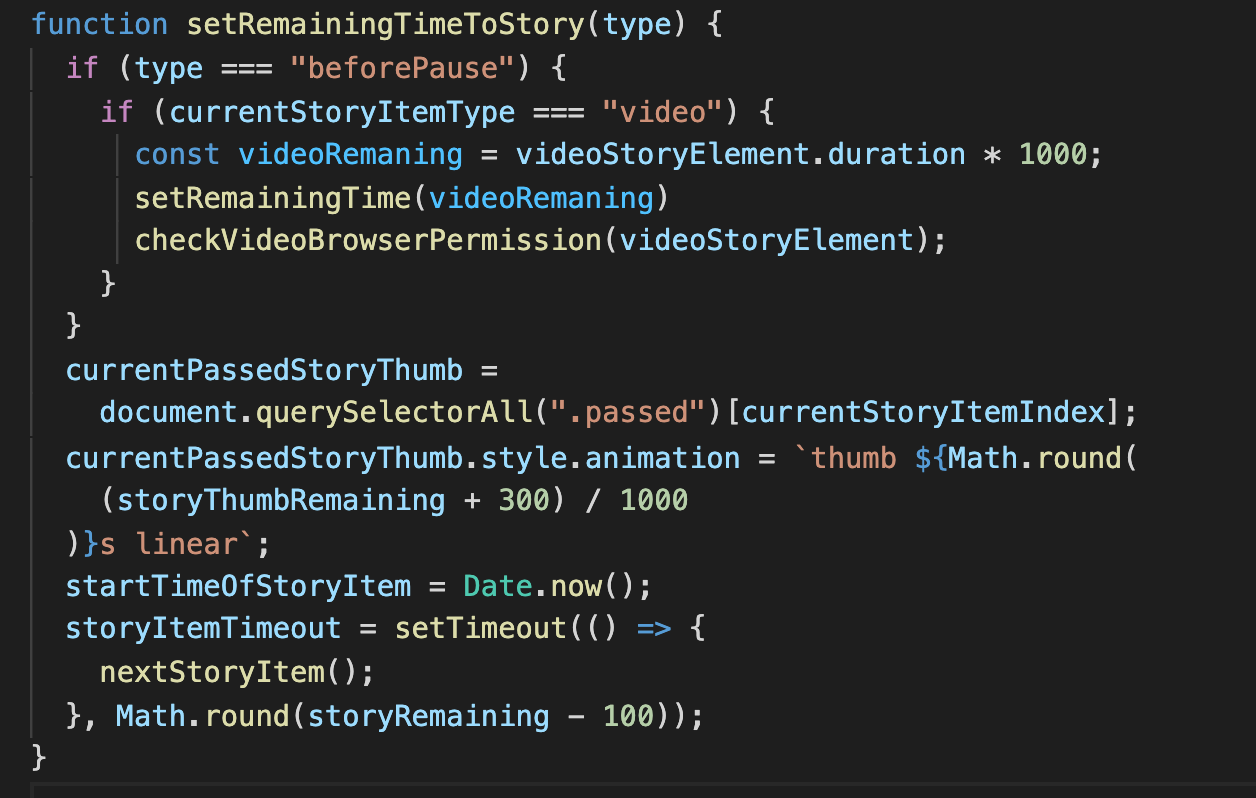


- I change this function name to a more meaningful name so i can know what this function does

Old code :



New code :

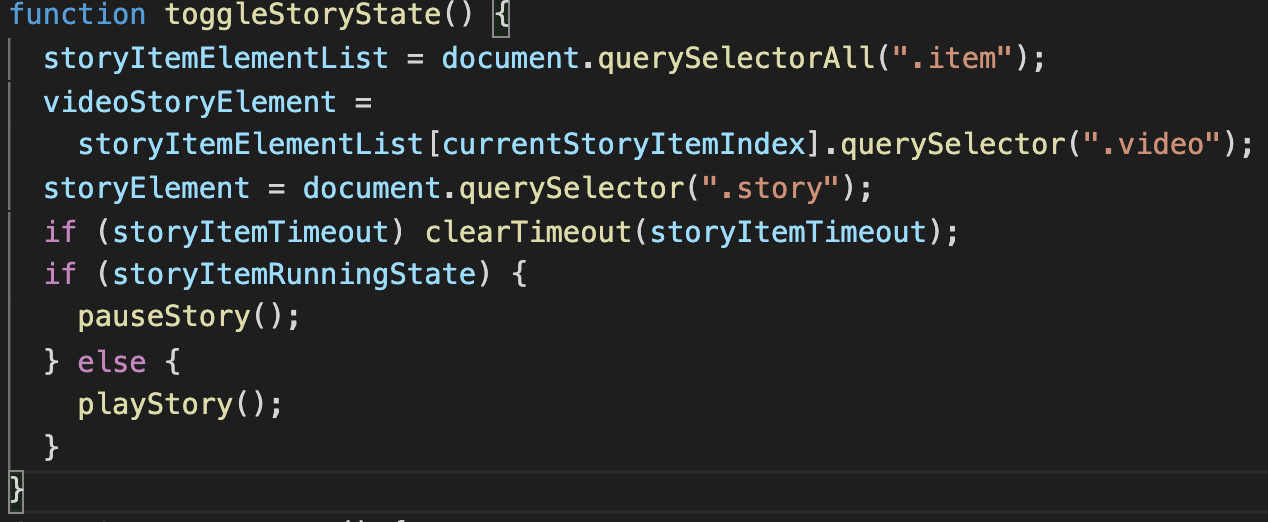


- When someone read this function, they probably don’t understand the Promise inside the function. So i wrap it into a function and name it clearly so if someone read it they can know what those code were written for.

Old code :



New code :

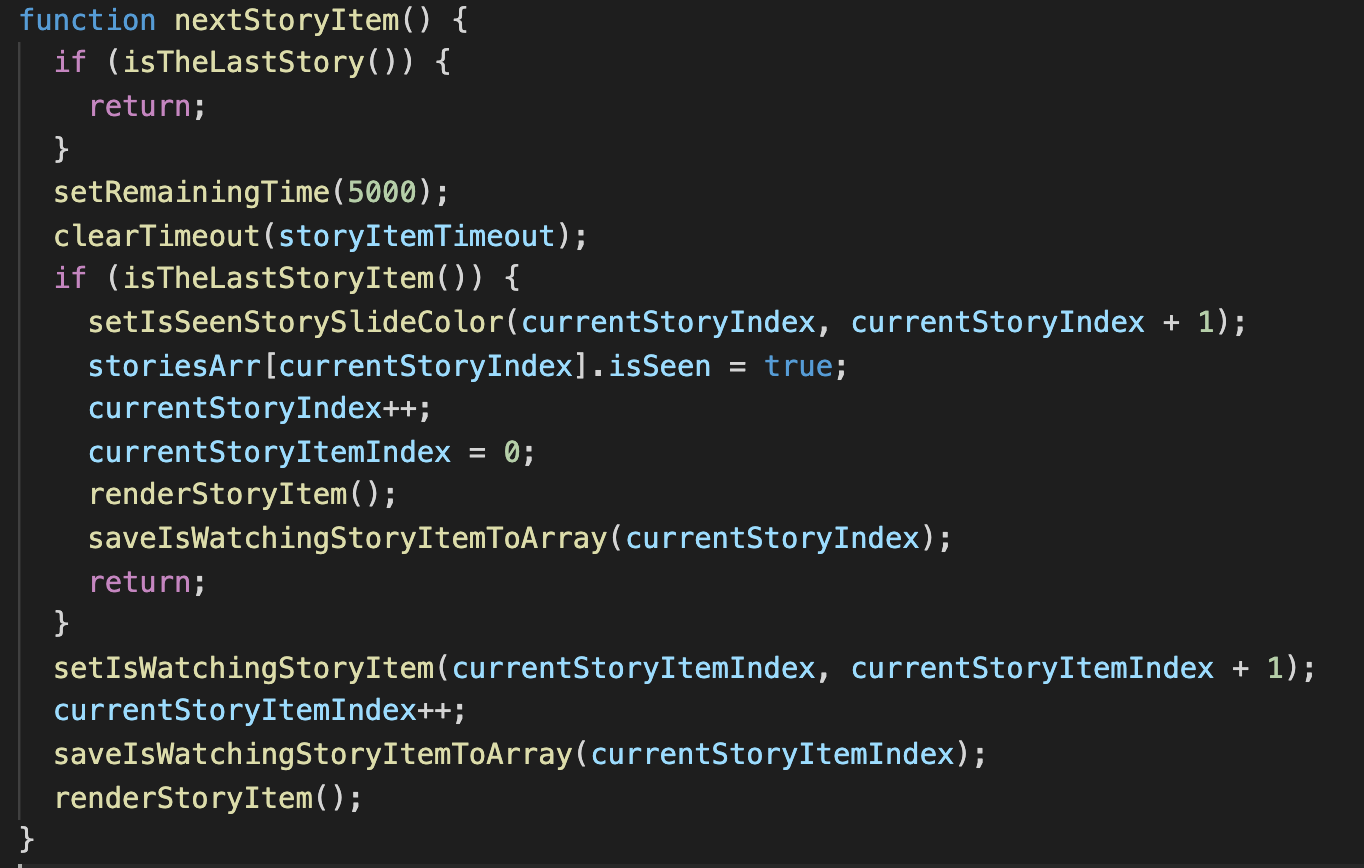


- This function is i wrote it very long and reduplicate some declaration so i write it shorter and separate function i can easier understand the function and easy to control.

Old code :



New code



- This function has a lot of duplicate code so i wrap it into a function for easier reusing.

- When someone read the condition maybe they get confused . So i wrap it into a return function and name it a clean name. They will easier to understand the condition.