**The central store**

The central store stores states (values of properties, that are changing all the time, that are being changes from multiple places, and are dislplayed in multiple places). One property must have one value, in all the places. Imagine it as a big store with shelfs.

In this example, we are storing two ways, two types of changing data:

1. Everything counter related: we are counting the clicks from multiple places, and displaying them in multiple places
2. Two way binding stuff: just an example how to connect two way binding with central store
3. store//this is the folder that contains all the store shit
   1. modules//modules is one way how we can separete different stuff in the central store, by topic (if it is a counter related stuff, or value related stuff)
      1. counter.js//this is the central store for all the counter related stuff. We have here all the mutations, actions, getters... for the counter stuff. Don’t forget, this is complicated, because we are using two ways of organising the central store (only now, for learning purposes), but in real life, we should use only one way. Either by modules, or by actions/getters/mutations/etc..
      2. value.js//this is basically empty, because the value topic was sorted in a different way.
   2. actions.js//... for the value stuff
   3. getters.js//... for the value stuff
   4. mutations.js//... for the value stuff
   5. store.js//basically, this is the CENTRAL STORE. But, since we have too many data, we migrated a lot of data to other files in the store folder. However, all the files in the store folder are connected this this store.js, and are merged together as one big central store.