Notes on tasks and what is needed to do for current project:

2-9-2019:

* Currently I need to create a types objects that I’m going to pass the data
* What will need to do in my project:
  + Added Types and helpers to my project
* I need to add a Board and fake Buttons for history just show that the app can shrink and grow in a flex box, as well I will have to start setting up Redux to output squares I believe.
  + In the moment I’m getting Redux setup. And connecting the data to my components which will output Header and Board
  + When connected, I will make sure that both components scale properly and are in a flex container
  + What to think about moving forward? Well when I’m wanting to interact with this data and change, I will have to create a action-creator and action will call a dispatch then a copy of the object so it’s mutable

I have created a action-creator, still need to create a action and update the reducer.

Notes on redux with examples below :

What things will I need to use redux in my app?

* Npm I redux, react-redux, redux-thunk, redux-logger

How to use combineReducers in redux?

* **import { combineReducers } from 'redux'**

How to use createStore method from redux?

* **import { createStore } from 'redux'**
* How parameters doe’s createStore use?
  1. Reducer, or combineReducers(list of reducers here)
  2. preloadedState: such as default state, but you don’t have to pass that there
  3. [enhancer]: or you can add applyMiddleWare() method with can take many enhancers

How to use and do action-creators:

* Action-creators simply create an action like so below

function addTodo(text) {

return {

type: ADD\_TODO,

text

}

}

* This is a function that returns a object, that a store.dispatch(action-here)

How to use connect() in your react-app?

How to create and use a reducer in redux:

import { VisibilityFilters } from './actions'

const initialState = {

visibilityFilter: VisibilityFilters.SHOW\_ALL,

todos: []

}

function todoApp(state = initialState, action) {

switch (action.type) {

case SET\_VISIBILITY\_FILTER:

return Object.assign({}, state, {

visibilityFilter: action.filter

})

default:

return state

}

}

When you have many reducers, use combineReducers to combine many reducers into 1 and that is inputted into createStore method, example below.

import { combineReducers } from 'redux'

const todoApp = combineReducers({

visibilityFilter,

todos

})

export default todoApp