App-breaking Mistakes that were fixed

* You should have expected fridgeList to contain 2, not 3 items
  + You started with an empty fridge
  + Then added apples and pineapples (no oranges)
  + You should expect it’s length to be at most 2 (not 3)
* Your return statement in checkRecipe was inside the for loop so it returned before you were done checking all items
* The word “this” used inside a spec will refer to the anonymous function inside the ‘it()’ function
  + If you are gonna do this (and you REALLY,REALLY shouldn’t) use the fat-arrow method of defining anonymous functions ‘ () => {}’
* In the checkRecipe function there was more than 1 place where you had “return something.push(r)”
  + Huge NO NO
  + And javascript won’t warn you about it because it will think you want to return the exit code of the functions’ execution
  + You never return push()
    - push() is an action (to add something to an array)
    - When you return something you are essentially sending it to the block of code that called your function
    - So you can’t send an action
    - That’s like you telling your buddy “ Hey , I’m gonna send you a walk ”

WARNING!!

If you have 9 oranges in your fridge and the recipe asks for 3 oranges, the fridge list will have 3 oranges.

If you intended for it to work this way, then cool. If not, you need to also check how many oranges are in the fridge in addition to the ones demanded by the recipe

General Feedback

* + A function should be as self-contained as possible:
    - There is no reason for the sList and fList to not be declared inside your function since they are not used anywhere else
      * Also, the next time your function runs, the stuff in sList from the previous run will still be there and you will wind up with the results of both tests combined
    - The checkRecipe function is too big. If you have a function doing 3 things, that is a sign that it should be separated into smaller functions
      * If you find yourself having to do a loop within a loop, that’s generally a bad sign
        + It becomes hard to track exactly where you are in the program

Which might be what caused you to have your return statement in the wrong place

* + - You don’t need to declare a class each time you want to send an object
      * For a simple object that has only 2 attributes, no methods, and is never used anywhere else in your program, an object literal declaration should be enough
      * Kim wrote this on the board for you
      * response = { listOfFridge: fridgeListArray, shoppingList: shoppingListArray } ; return response
      * But change the variable names