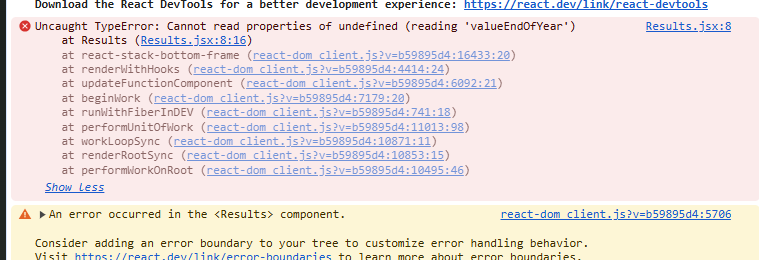
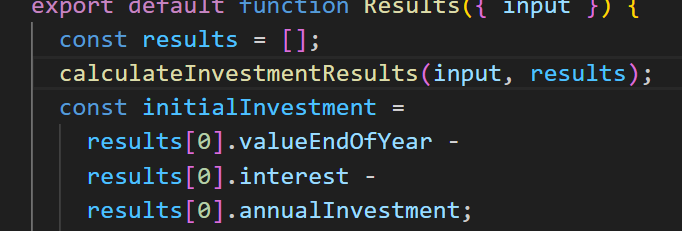
Debugging:

The moment I enter 0 in Duration I get this error in Console  


Uncaught TypeError: Cannot read properties of undefined (reading 'valueEndOfYear')

at Results (Results.jsx:8:16)

Browser failed to read properties of an undefined value of ‘valueEndOfYear’. we need to search where we are accessing the value of ‘valueEndOfYear’ property

We are accessing the value in Results.jsx file  


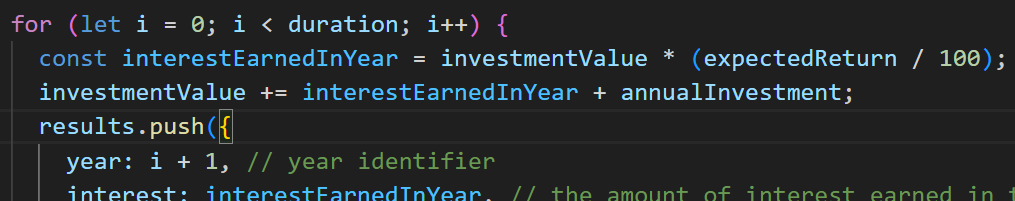
After the error message we have the stack trace: after the list of code executions, it led to that error

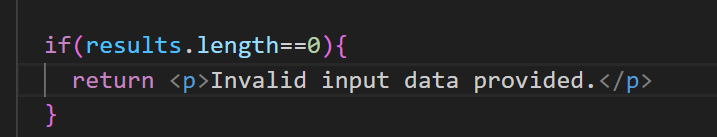
We can see -> Results.jsx:8:16

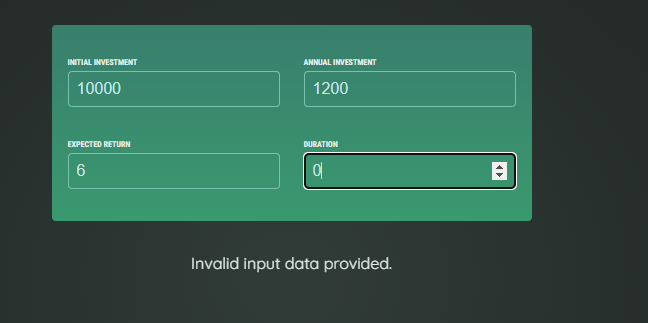
results[0] seemed to yield an undefined value

Therefore we need to go to the place where ‘results’ is populated

Go to calculateInvestmentResults(input, results);

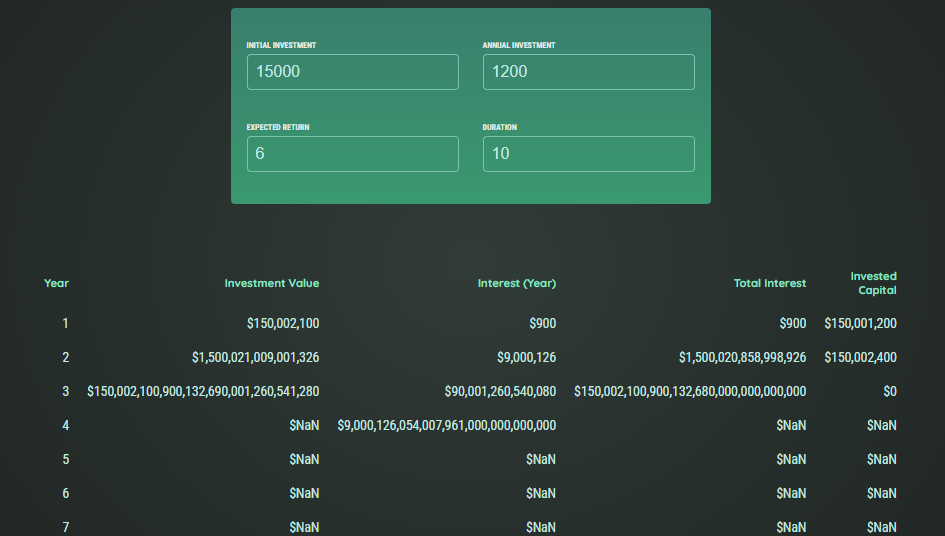
  
we push new elements in this ‘for’ loop but this ‘for’ loop doesn’t seem to execute in the first place. We think that’s because of ‘duration=0’. if ‘for’ loop doesn’t get executed the code will fail. To fix we added an ‘if’ check



No error message now  


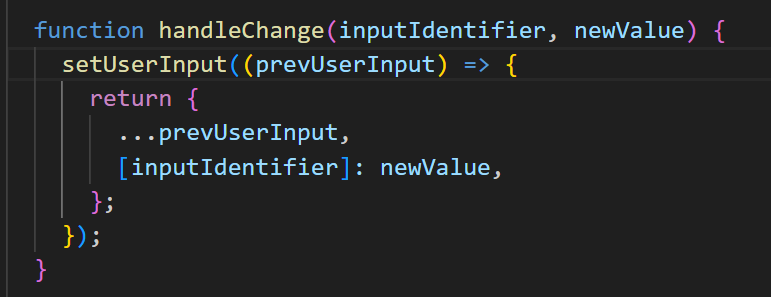
Logical errors:

If I change ‘Initial Investment’ to 15000 then it shows $NaN

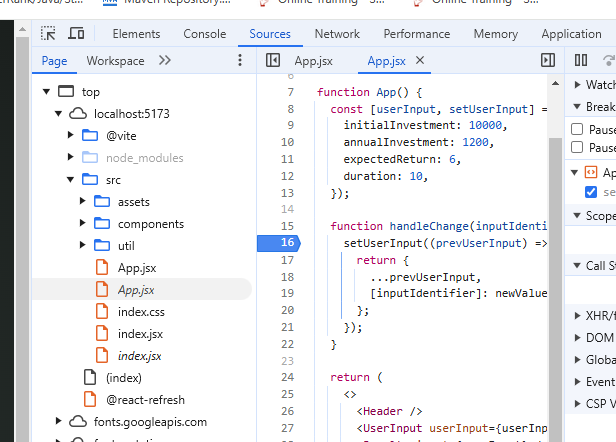


When I re-load the page everything seems to work fine so it is not from the ‘calculateInvestmentResults(input, results)’. it breaks only if we change the value. So it could be related to how we fetch the user input in ‘App.jsx’

This part of code might be the reason

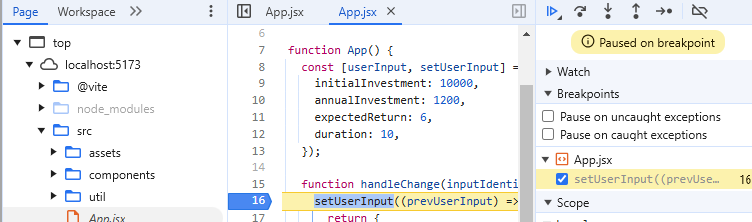


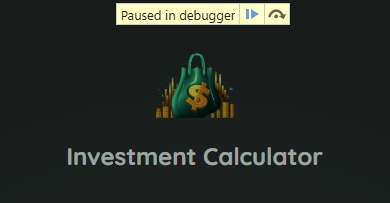
Go to ‘Sources’ tab and place a breakpoint by clicking on the linenumber



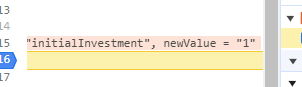
The code execution will stop at the breakpoint

Now we can see it is paused on the breakpoint as soon as we change ‘Initial Investment’

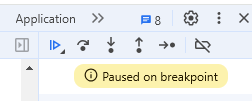


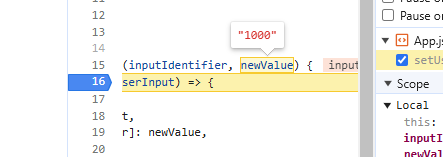


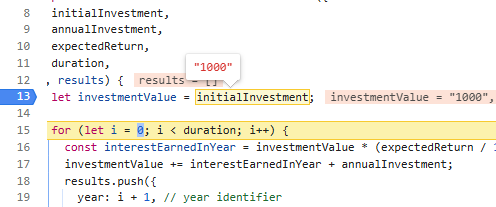
If you scroll you can see the new value entered



We can click on ‘Step over’ button to see how the control is flowing

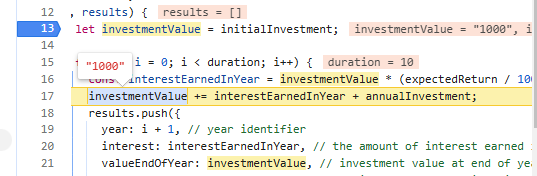


We can clearly see when we hover over newValue that it is a String  


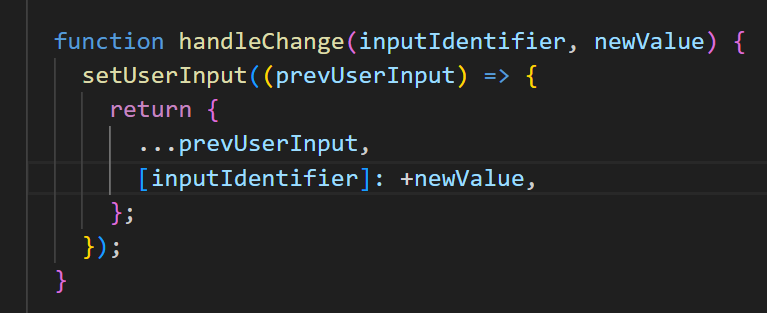
Now I added another breakpoint in calculateInvestmentResults() function  


Here the ‘initialInvestment’ is “1000” string

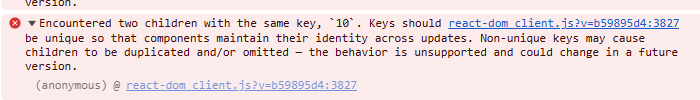
In Line 17, initialInvestment value is “1000” string



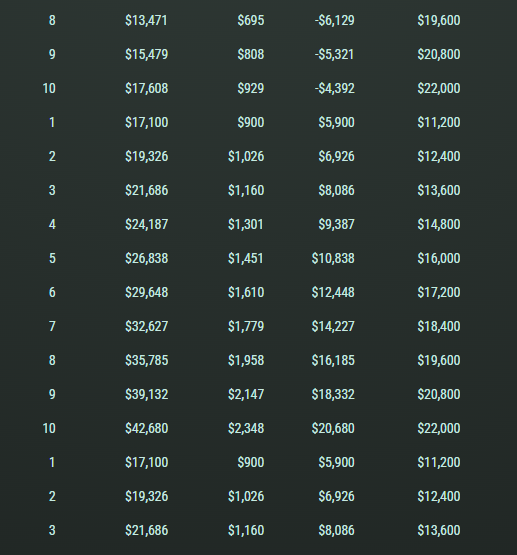
To make sure input value is stored as a number we add a ‘+’ sign before



Another React feature that can help to catch and fix some kind of errors

I replace the ‘Results.jsx’ file  


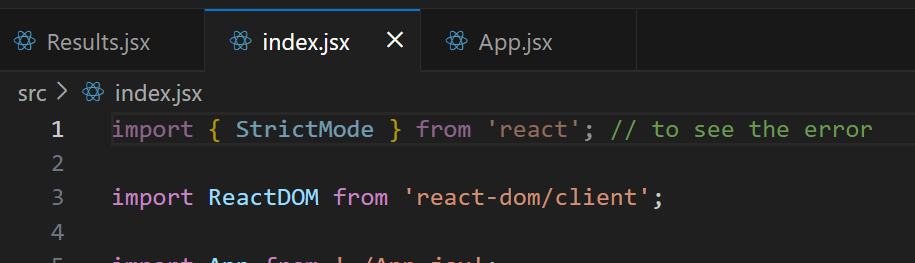
I get this new error. It’s about two children having the same key

Also this list keeps getting longer and longer instead of being replaced  


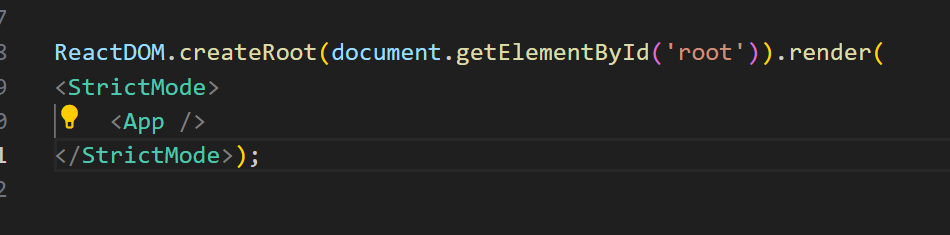
We see this problem only when we edit the inputs

We could see this error right when the App starts using the ‘Strict’ mode

We import ‘StrictMode’ in index.jsx file

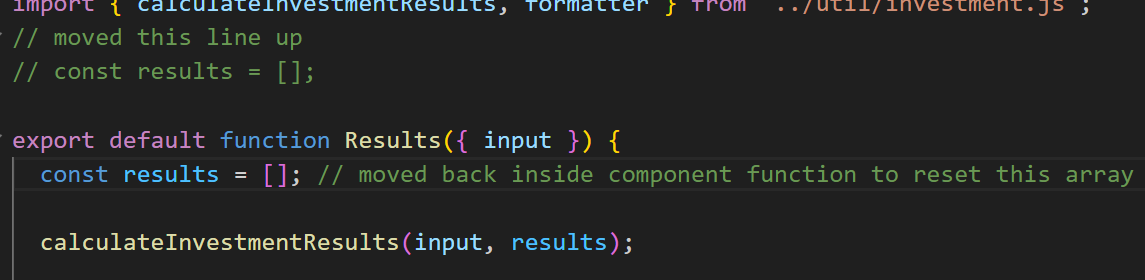


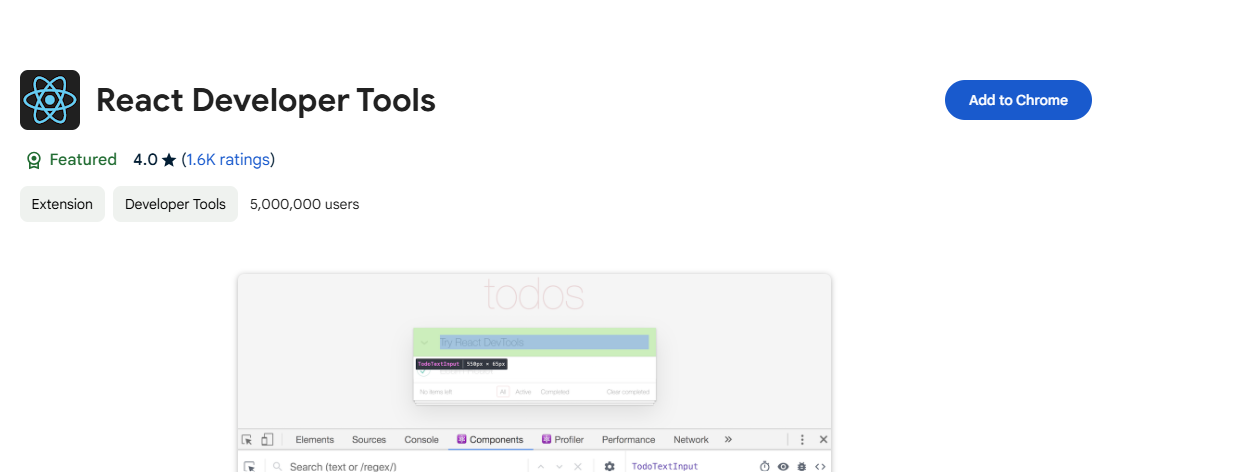
‘StrictMode’ is a React component that’s meant to be wrapped around other components example ‘App’ component



We could wrap any other component as well. ‘StrictMode’ executes every component twice instead of just once and only does that only during development not in Production at all. It doesn’t even tell me what the problem is but it exaggerates the problem. Here the problem is, results array is created in the top then more and more elements are added without resetting the array

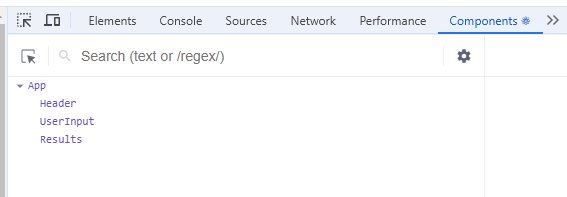
This fixes the error





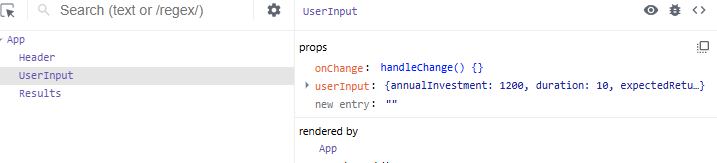
Now go to ‘Inspect’ we see a new tab ‘Profiler’. this is for improving React performance

Click on ‘Components’ tab:



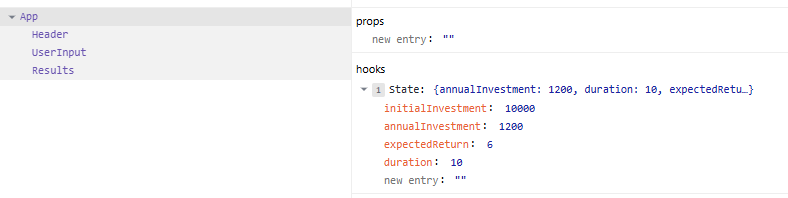
If you hover over ‘UserInput’, it will highlight the respective component. If you click, you will learn more about component.

We can see what props it accepts



userInput is an object with following properties

If you click on the ‘App’ component, we see the State hook



If the component manages ‘States’ we can see them under hooks

It uses State hook to create one State value and we can edit that value as well

Components is a good tool to see the component tree, which component controls which part of UI, also we can edit component tree.