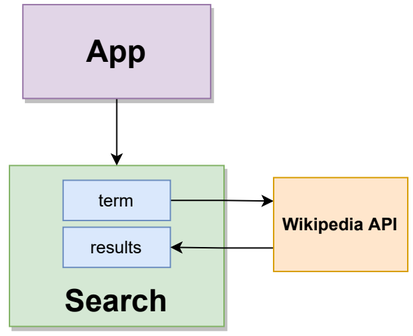
Page 1
localhost/list
Accordion
List
Dropdown
Translate
Search
programming
Programming
Programming is the process of building a computer program
Program
A program is a collection code code that is executed on a computer
Functional Programming
Functional programming is a style of programming


App
term
results
Wikipedia API
Search

en.wikipedia.org/w/api.php?action=query&list=search&format=json&origin=*&srsearch=programming

Wikipedia API

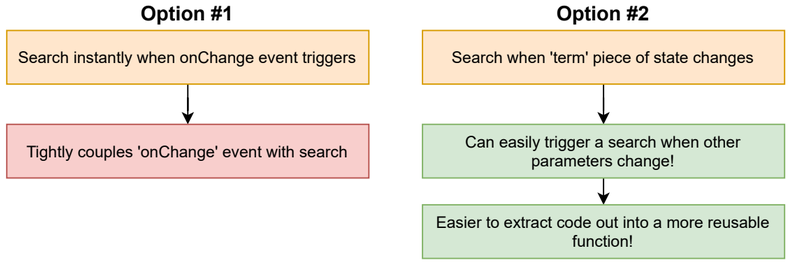
When Do We Search?

Option #1

- User types in input
- onChange event handler called
- We take value from input and make request to API
- ....
- Get response
- Update 'results' piece of state
- Component rerenders, we show list of results

Option #2

- User types in input
- onChange event handler called
- Update 'term' piece of state
- Component rerenders
- *We add code to detect that 'term' has changed!*
- Make request to API
- ....
- Get response
- Update 'results' piece of state
- Component rerenders, we show list of results

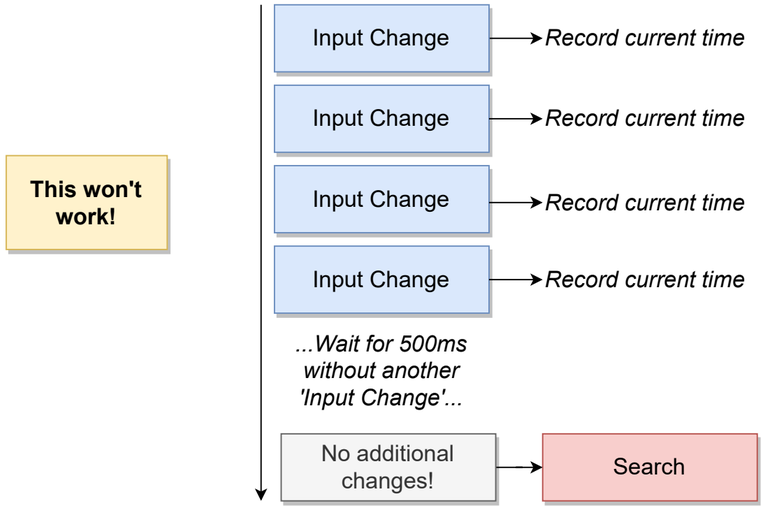

Option #1
Search instantly when onChange event triggers
Tightly couples 'onChange' event with search
Option #2
Search when 'term' piece of state changes
Can easily trigger a search when other parameters change!
Easier to extract code out into a more reusable function!

The 'useEffect' Hook

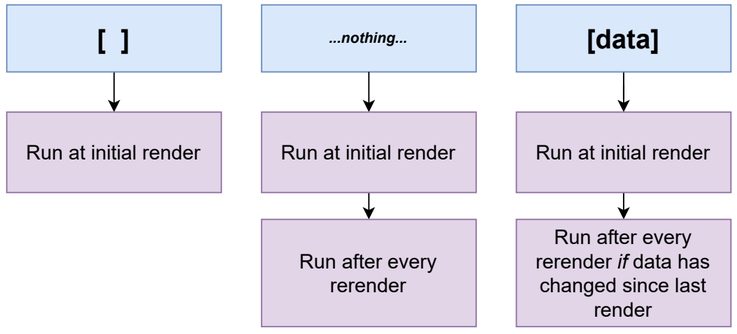
Allows function components to use *something like* lifecycle methods

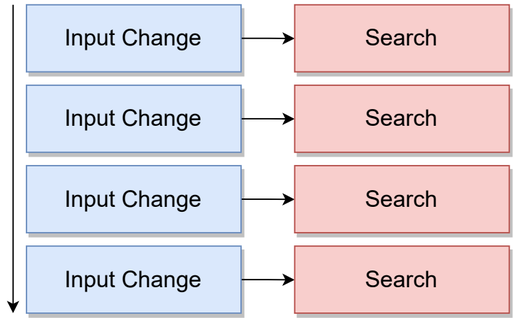
We configure the hook to run some code automatically in one of three scenarios

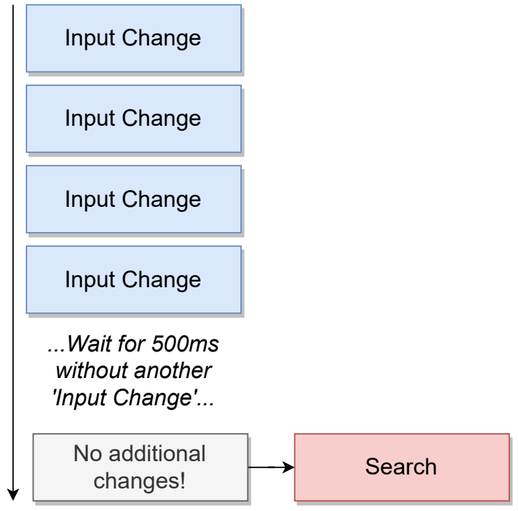
1	When the component is rendered *for the first time only*
2	When the component is rendered ***for the first time and whenever it rerenders***
3	When the component is rendered ***for the first time and (whenever it rerenders and some piece of data has changed)***

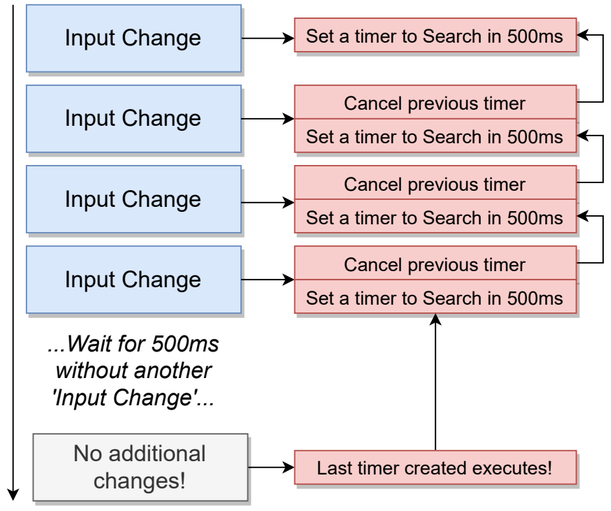

This won't work!
Input Change
Record current time
Input Change
Record current time
Input Change
Record current time
Input Change
Record current time
...Wait for 500ms without another 'Input Change'...
No additional changes!
Search

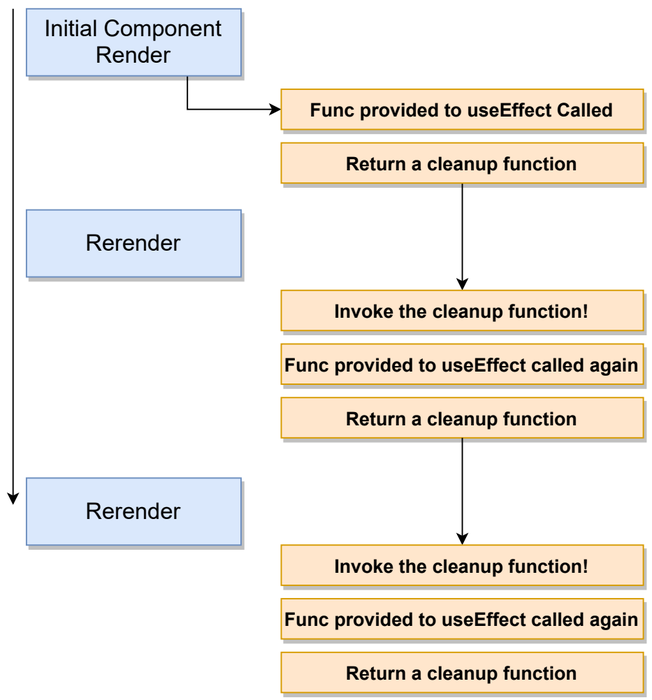
UseEffect Second Arugment

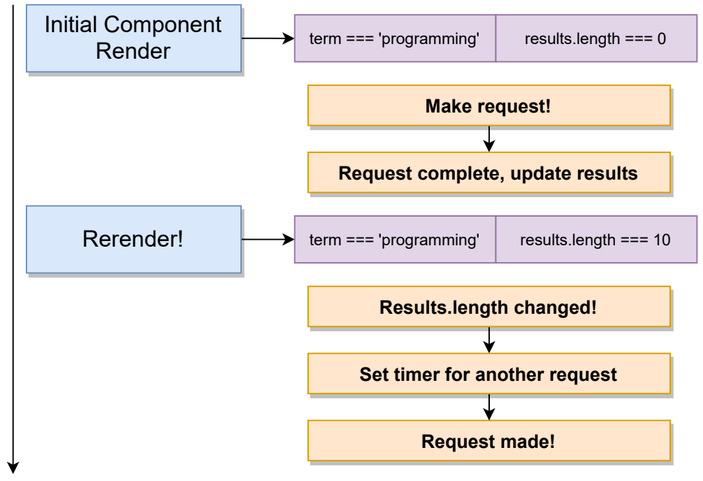


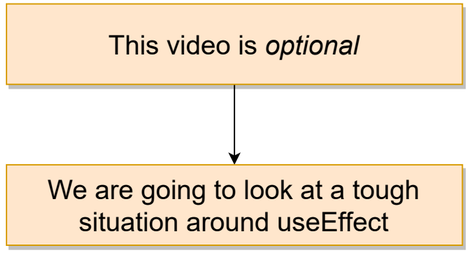

Input Change
Search
Input Change
Search
Input Change
Search
Input Change
Search

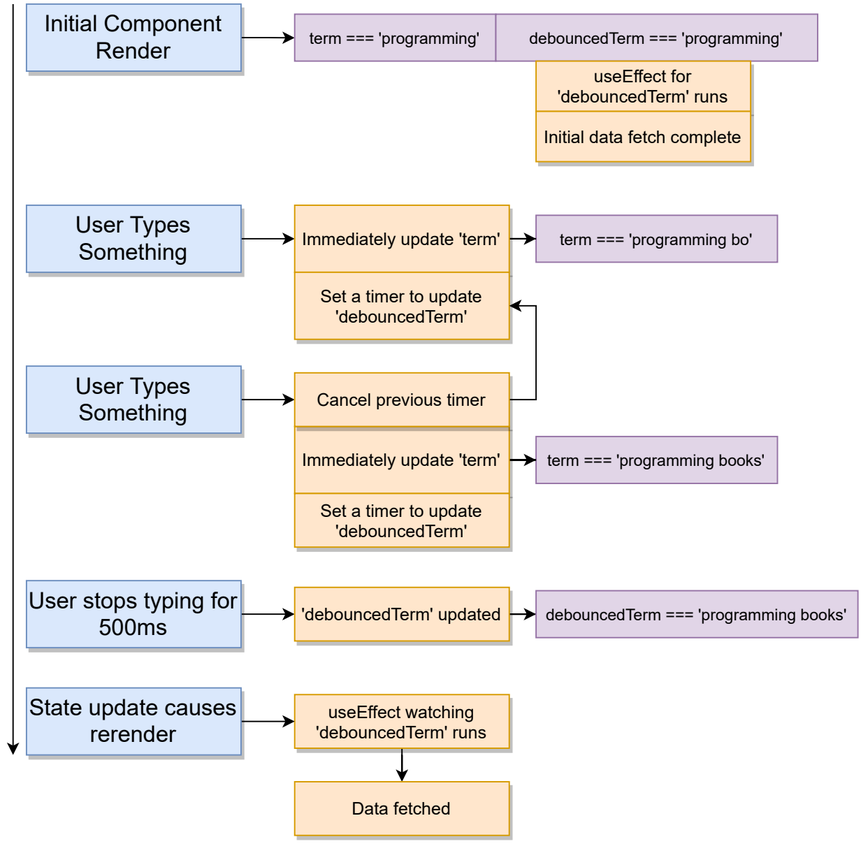

Input Change
Input Change
Input Change
Input Change
...Wait for 500ms without another 'Input Change'...
No additional changes!
Search


Input Change
Set a timer to Search in 500ms
Input Change
Cancel previous timer
Set a timer to Search in 500ms
Input Change
Cancel previous timer
Set a timer to Search in 500ms
Input Change
Cancel previous timer
Set a timer to Search in 500ms
...Wait for 500ms without another 'Input Change'...
No additional changes!
Last timer created executes!


Initial Component Render
Func provided to useEffect Called
Return a cleanup function
Rerender
Invoke the cleanup function!
Func provided to useEffect called again
Return a cleanup function
Rerender
Invoke the cleanup function!
Func provided to useEffect called again
Return a cleanup function


Initial Component Render
term === 'programming'
results.length === 0
Make request!
Request complete, update results
Rerender!
term === 'programming'
results.length === 10
Results.length changed!
Set timer for another request
Request made!


This video is *optional*
We are going to look at a tough situation around useEffect


Initial Component Render
term === 'programming'
debouncedTerm === 'programming'
useEffect for 'debouncedTerm' runs
Initial data fetch complete
User Types Something
Immediately update 'term'
term === 'programming bo'
Set a timer to update 'debouncedTerm'
User Types Something
Cancel previous timer
Immediately update 'term'
term === 'programming books'
Set a timer to update 'debouncedTerm'
User stops typing for 500ms
'debouncedTerm' updated
debouncedTerm === 'programming books'
State update causes rerender
useEffect watching 'debouncedTerm' runs
Data fetched