

useMemo

useMemo allows you to memoize expensive functions so that you can avoid calling them on every render.

```
// Memoize letterCount by ensuring it gets the expensive computeLetterCount step
const computeLetterCount = (word) => {
  let i = 0;
  while (i < 1000000000) i++;
  return word.length;
};

// Memoize computeLetterCount so it uses cached return value if input array ...
// ... values are the same as last time the function was run.
const letterCount = useMemo(() => computeLetterCount(word), [word]);
```

memo() is a higher order function

useMemo() is a hook to memoize expensive functions so they don't run again on re-render

Code Challenge review

1 – Code challenge introduction

Buddy has to present the code challenge.

Boilerplate:

<https://github.com/DX-and-EE-South-Europe/react-for-angular-devs>

2 – Implement the challenge

- TODO list
- Forms (validations)
- API calls example
- Performance example



Complete Availability:

1 week of self coding

Partial Availability:

2 weeks of self coding

3 – Code challenge Resolution

Buddy shows our implementation and resolve the last questions