


---

# What is Memoization in JavaScript?




**Memoization** is a programming technique which attempts to increase a function's **performance** by **caching** its previously computed results.



Each time a **memoized** function is called, its parameters are used to **index** the **cache**. If the data is present, then it can be returned, without executing the entire function. Otherwise the function is executed and then the result is added to the cache.

Let's take an example of adding function with memoization



```
1  const memoization = () => {  
2    const cache = {};  
3    return (value) => {  
4      if (value in cache) {  
5        return cache[value];  
6      } else {  
7        let result = value + 20;  
8        cache[value] = result;  
9        return result;  
10     }  
11   };  
12 };
```

Remaining Part >>



```
1 // Returned Function from memoization
2 const returnedFunction = memoization();
3 console.log(returnedFunction(100));
4 console.log(returnedFunction(100));
```

**Did you found it Helpful?**



**Mohit Sehrawat**  
**Full Stack Developer**

**Like** and **comment** if you  
found it **helpful!**

**FOLLOW FOR MORE!**