


Currying

Interview question-42



Follow on 
@Duvvuru Kishore



simple explanation for Non-coder

Imagine you want to make a sandwich with different ingredients: bread, cheese, and lettuce. Curryng is the process of preparing each ingredient one by one and then combining them to create the final sandwich.



currying for coder

Currying is a transformation of functions that translates a function from callable as **f(a, b, c)** into callable as **f(a)(b)(c)** a sequence of nesting functions.

explanation and example

```
function curriedSum(a) {  
  return function(b) {  
    return a+b;  
  }  
};  
}  
curriedSum(1)(2) //3
```

**you need to understand how function arguments
are taken and performing the calculation**

explanation




- **When `curriedSum(1)` called, the argument is saved in the `Lexical Environment`.**
- **This means we're giving the `curriedSum` function its first argument, and it returns a new function that remembers `a` as 1:**
- **and a new wrapper (which is 2 in our case) will return `function(b)`.**
- **Then this wrapper is called and it passes the call to the sum. finally giving desired result**

Note: few interview questions deal with currying. In normal programming world we can go with normal functions as both are similar

 Follow



Follow on 
@Duvvuru Kishore

