Coding Question on Currying

Interview questions-43

Follow on in

@Duvvuru Kishore

How do you solve the problem sum(1)(2)(3)(4)(5)()?

Interviewer: The code should work for n arguments

Solution

As of discussion(previous posts) we can solve the above question with currying

solution and explanation 👉

```
function sum(a) {

return function(b){
if(!b){          //to check next function is available or not
return a;
}
return sum(a+b);
}
}

console.log(sum(1)(2)(3)(4)(5)()); //15
```

explanation 👉

Step 1

```
function sum(1) {

return function(2){
if(!b){
return a;
}
return sum(1+2); //3
}
}
```

Step 2

```
function sum(3) {

return function(3){
if(!b){
return a;
}
return sum(3+3); //6
}
}
```

Step 3

```
function sum(6) {

return function(4){
if(!b){
return a;
}
return sum(6+4); //10
}
}
```

Step 4

```
function sum(10) {

return function(5){
if(!b){
return a;
}
return sum(10+5); //15
}
}
```

explanation with words 👉

Explanation of above steps :

Step1 : The first argument Sum(1) is passed and in return there is function which takes another argument (2). As there is next argument it skips if stmt and its goes to return sum(1+2). As its calling sum function again and the process continues which we call recursion

Step2: now sum(3) function continues and in return there is function which takes another argument (3). As there is next argument it skips if stmt and its goes return sum(3+3). As its calling sum again and the process continues

Step 3 and step 4 continues as same as above steps

In the same way we can achieve n arguments

⊞ Follow

M 🔛 in

Follow on in

@Duvvuru Kishore