

* polyfill for bind method (interview question)

→ what is polyfill

polyfill is sort of a browser fallback.

what if suppose, browser doesn't have a bind function and you have to write your own bind function.

This is called as polyfill.

→ what is bind()

The bind() method creates a new function that, when called, has its this keyword set to the provided value, with a given sequence of arguments preceding any provided when the new function is called.

In simple words,

The bind() takes an object as an argument and returns a new function whose "this" refers to the object we passed as an argument.

① example of actual bind method

```
let name = { firstName: "chaitanya" };  
let printName = function() {  
  console.log(this.firstName);  
};
```

```
let printMyName = printName.bind(name);  
printMyName();
```



chaitanya

→ if we keep any method on "Function.prototype" then each and every method what we write has access to those methods.

→ basic implementation of bind (polyfill)

```
let name = { firstName: "chaitanya" };  
let printName = function() {  
  console.log(this.firstName);  
};  
Function.prototype.myBind = function(... args) {  
  let obj = this;  
  return function() {  
    obj.call(args[0]);  
  }  
}  
let printMyName2 = printName.myBind(name);  
printMyName2();
```



chaitanya

We are done with the very basic implementation of bind method.

Here our own myBind() method is exactly behaving like bind() method.

But we are completely not done with it, there is still a problem.

And the problem is with the arguments.

→ example of actual bind method with arguments (not polyfill)

```
let name = { firstName: "Chaitanya" };  
let printName = function(hometown) {  
  console.log(this.firstName + " " + hometown);  
};  
let printMyName = printName.bind(name, "Bhenda");  
printMyName();
```

↓
Chaitanya Bhenda

Now we need to have this functionality in our own myBind() method

Now we need to handle arguments

→ final version of polyfill for bind() method

```
let name = { first: "Chaitanya" };  
let printName = function (hometown, state) {  
  console.log(this.first + " " + hometown + " " + state);  
}
```

```
Function.prototype.myBind = function (...args) {  
  let obj = this,  
      params = args.slice(1);  
  return function (...args2) {  
    obj.apply(args[0], [...params, ...args2]);  
  }  
}
```

```
let printMyName2 = printName.myBind(name, "Bhenda");  
printMyName2("Maharashtra");
```



Chaitanya Bhenda Maharashtra