

JS call apply bind

“JS call apply bind

this this

```
box.onclick = function(){
  function fn(){
    alert(this);
  }
  fn();
};
```

this box, Window

```
box.onclick = function(){
  var _this = this;
  function fn(){
    alert(_this);
  }
  fn();
};
```

this

call apply bind

this

```
box.onclick = function(){
  function fn(){
    alert(this);
  }
  fn();
};
```

```
box.onclick = function(){
```

```
function fn(){
  console.log(this);
}
fn.call(this);
};
```

call this

```
fn.call(this);
```

this fn this box javascript box fn this

```
box.onclick = function(){
  function fn(){
    console.log(this);
  }
  fn.call(this);
};
```

```
box.onclick = function(){
  var fn = function(){
    console.log(this); //box
  }.call(this);
};
```

```
box.onclick = function(){
  (function(){
    console.log(this);
  }.call(this)); //box
};
```

```
var objName = {name: 'JS2016'};
var obj = {
  name: '0 _ 0',
  sayHello: function(){
    console.log(this.name);
  }.bind(objName)
};
obj.sayHello(); //JS2016
```

call apply bind this

```
function fn(a, b, c, d){
  console.log(a, b, c, d);
}

//call
fn.call(null, 1, 2, 3);

//apply
fn.apply(null, [1, 2, 3]);

//bind
var f = fn.bind(null, 1, 2, 3);
f(4);
```

```
1 2 3 undefined
1 2 3 undefined
1 2 3 4
```

null

call apply bind call apply call apply bind this

```
var objName = {name: 'JS2016'};
var obj = {
  name: '0 _ 0',
  sayHello: function(){
    console.log(this.name);
  }.bind(objName)
};
obj.sayHello(); //JS2016
```

bind call sayHello obj sayHello

Math.max

```
Math.max(10, 6)
```

apply

```
var arr = [1, 2, 30, 4, 5];
console.log(Math.max.apply(null, arr));
```

```
function fn(){
  [].push.call(arguments, 3);
  console.log(arguments); //[1, 2, 3]
}
fn(1, 2);
```

```
var arr = ['aaabc'];
console.log('').indexOf.call(arr, 'b')); //3
```

```
var str = 'aaabc';
console.log(str.indexOf('b'));
```

```
var arr = ['aaabc'];
''.indexOf.call(arr);
```

arr indexOf

100%

```
('').indexOf.call(arr, 'b')
```

arr ['aaabc'] 'aaabc'

```
'aaabc'.indexOf('b');
```

bind

```
Function.prototype.$bind = function(obj){
  // this
  var _this = this;
  //
  var a = [].slice.call(arguments, 1);
  //
  return function(){
    // this obj      concat      f(4)      a argument
    _this.apply(obj, a.concat([].slice.call(arguments)));
  };
}
```

```
};  
};
```

bind

```
var arr = [ ' JSS' ];  
  
var index = ''.indexOf.$bind(arr, ' S');  
console.log(index())  
  
-----  
  
function fff(){  
  [].push.$bind(arguments, 1);  
  console.log(arguments);  
}  
  
fff();
```

bind

Revision #1

Created 17 July 2021 18:04:08 by

Updated 17 July 2021 18:15:57 by