



链滴

什么是函数式编程？

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1549950785958>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

回答

函数式编程是一种编程范式，他使用**纯函数**来构建声明方式以避免状态的共享、数据的易变及一些副作用。函数式编程的核心是：同一个函数中只要输入相同，其返回值就必然相等，且不会产生任何副作用。大多数程序员认为这是软件开发的最佳方法，因为这可以减少 bug 并增加易读性。

加分回答

- 与使用命令式或面向对象编程的代码相比，函数式编程的代码更加简洁、清晰、可预测及易于测试。
- `String.prototype.toUpperCase`、`Array.prototype.map`、`Function.prototype.bind` 都是 JavaScript 中纯函数的例子，其非纯函数的例子有：`Date.now`、`Math.random` 等
- JavaScript 是多范式编程语言（面向对象编程和函数式编程并存）
- 函数式编程的应用
- 高阶函数是一个函数以函数为参数，或以函数为返回值，或者既以函数为参数又以函数为返回值。
- 偏函数是通过指定部分参数来产生一个新定制的函数，如：

```
const isType = function (type) {  
  return function (obj) {  
    return toString.call(obj) === '[object' + type + ']';  
  };  
};  
const isString = isType('string');
```

- 柯里化是将一个多参数函数转换成多个单参数函数，如：

```
function addX(y) {  
  return function (x) {  
    return x + y;  
  };  
}  
addX(2)(1) // 3
```

- 函数复合是结合两个或多个函数，从而产生一个新的函数以简化中间步骤。如：

```
const add = (a, b) => a + b;  
const mult = (a, b) => a * b;  
add(2, mult(3, 5)) // 17
```

返回总目录

每天 30 秒