



TypeScript 泛型

类型验证的好处

如何保证`add`函数是接受的参数都是`number`，

```
function add(x, y) {  
  return x + y;  
}
```

在运行时的时候手动验证

```
function add(x, y) {  
  if(typeof x === 'number' && typeof y === 'number') {  
    return x + y;  
  }  
  throw error;  
}
```

利用第三方工具验证

```
import {object, number} from "superstruct"
function add(x, y){
  const struct = object({
    x: number(),
    y: number()
  })
  assert(struct, {x, y});
  return x + y;
}
```

利用ts在编译时的时候验证

```
function add(x:number, y:number) {
  return x + y;
}

add('1', 1);
```

| 类型验证 | 验证时间 | 优点 | 缺点 |
|-------|------|------------------------------------|---------|
| js | 无 | 项目规模小时配合灵活，开发速度快 | 维护困难 |
| ts | 编译时 | 编译时验证，需要用户书写各种类型,配合编辑器在开发时就能避免一些错误 | 无 |
| 第三方工具 | 运行时 | 对于排查错误(甩锅)，项目代码健壮性有作用 | 引入额外的依赖 |

TS类型检查的限制-类型擦除

TS最终是会编译到JavaScript，TS的类型会被擦除

TypeScript 3.8 引入了 `import type` 的语法，可以显式指明引入类型，从而杜绝值空间潜在的循环引用问题

类型推断

自动类型推断

```
function add<T extends number>(x: T, y: T){  
  return x + y;  
}  
  
const res1 = add<number>(1,2); // number  
const res2 = add('1',2); // type check error  
const res3 = add<string>('1','2'); // typecheckerror
```

前置知识

- 联合类型
- 交叉类型
- 映射类型
- 索引类型
 - ``keyof T`` 获取所有属性名联合
 - ``T[K]`` 获取某个属性的类型
- ``extends ?`` 分支
- 递归 循环
- 依赖类型

类型的基本操作

- 联合类型

```
type Union = string | number;
```


交叉类型是将多个类型合并为一个类型。这让我们可以把现有的多种类型叠加到一起成为一种类型，它包含了所需的所有类型的特性。例如，`Person & Serializable & Loggable`同时是`Person`和`Serializable`和`Loggable`。就是说这个类型的对象同时拥有了这三种类型的成员。

■ 交叉类型

```
type ErrorHandler = {  
  success: boolean  
  error: Error  
}  
  
type PersonVip = {  
  desc: string  
}  
type PersonVisitor = {  
  desc: string  
}  
  
type PersonVipResponse = ErrorHandler & PersonVip  
type PersonVisitorResponse = ErrorHandler & PersonVisitor
```

值空间和类型空间

值

字面量(literal)、变量(variable)、常量(constant)、函数形参、函数对象、``class``、``enum``...

类型

``string``、``number``、``function``、``type`` 关键字定义的类型 ``interface``、``class`` 和 ``enum``...

一些好用的工具类型

- omit (忽略type当中的某个属性)
- pick (选择type当中的某写属性)
- exclude (排除联合类型中某些类型)

reference

- 读懂类型体操：TypeScript 类型元编程基础入门
- typescript handlebook
- TypeScript类型元编程入门指南