1. TypeScript = JavaScript + Type，给JavaScript添加了类型（类型注解）和接口等内容。
2. TypeScript包含JavaScript，可直接使用JavaScript语法
3. TypeScript代码先经过tsc编译为JavaScript代码再执行
4. declare声明的变量为全局变量
5. TypeScript的变量类型可以是自定义的类型，如Java的类和接口，这一点和面向对象语言类似，c++定义一个结构体作为数据类型，TypeScript定义一个接口作为一个数据类型。**注意**：Java的类声明时用类名声明，JS和TS声明用var声明，后面都是new 类名。
6. TypeScript新增的类型注解类似React的设置属性类型
7. TypeScript类型错误也能正常编译
8. TypeScript没有固定的说一定要用const声明一个常量或块级作用域之类的，都是通过设置编译的配置决定编译为哪个版本的JavaScript
9. TypeScript声明一个变量并赋值时可以省略类型注解，会被自动解析为赋的值的类型（使用基本类型和简单类型），声明一个变量而不赋值需要使用注解，否则该变量为any类型
10. 联合类型（字面量）：var num: Number | String，该变量只能是这几种类型，也可以是具体的值，表示该变量只能是这几个值中的一个
11. TypeScript把一个变量的值赋给另一个变量是根据类型赋值的，类似C++、Java，不同类型变量不能赋值，但是any类型可以赋值给其它变量并被赋值变量变为any类型
12. Any类型和JS一样，可以是任意类型。unKnown也是任意类型。区别：any可以赋值给其它类型，并且被赋值的变量类型变为any，而unKnown不能
13. TypeSc中函数返回值默认是any类型
14. Javascript函数默认返回undefined？？待定
15. TypeScript声明对象类型时，类型注解需要是{name:string，num:number}形式，该变量的实际对象值必须和类型注解的形式一模一样，不多不少。属性名后加问号表示可选。[prop:string]：any表示任意多个属性，且属性类型是any
16. TypeScript声明函数类型时，和声明对象类似，通过箭头函数的形式设置参数个数、类型以及返回值类型等。**注意**：声明函数和声明函数类型不一样，声明函数是使用function关键字声明一个函数，可以定义参数类型和返回值类型等，旨在调用时传递正确参数；而声明函数类型其实是声明一个变量的类型，即规定该变量必须是函数类型，并且通过箭头函数的形式规定其参数类型和返回值类型，旨在定义函数时声明正确参数
17. TypeScript声明数组，var arr: number[] = [1,2,3]
18. TypeScript元组：固定长度的数组，var tem: [number,number] = [1,2]
19. TypeScript枚举就是一些变量或者字符的集合
20. TypeScript类的使用、编译配置等
21. TypeScript类封装属性，可以直接get 属性名设置获取属性的代理，set 属性名设置更新属性值的代理
22. TypeScript类属性的封装三个修饰符：Private：仅该类自己使用，public：可外部访问可被子类继承，protected：不可外部访问，可被子类继承
23. 接口和抽象类相同点：抽象类被子类继承并实例化一个对象，接口可以被实现为一个类并实例化一个对象，这点用法类似。不同点：接口可作为一个变量类型（准确说是对象类型），规范了对象的属性和方法参数，这一点和抽象类不同，也和Java等语言的接口不同
24. TypeScript泛型就是当函数的参数和返回值的类型取决于传递的参数的类型时即函数参数和返回值类型可以是任意类型时，在函数名后加一个<T>表明一个类型，T的具体类型在调用函数时传入。
25. TypeScript具有JavaScript的特性，字符串也有熟悉和方法，可被解析为对象使用
26. 问题：ts能否直接操作对象原型？可以。对象直接声明还是用类实例？都可以
27. 继续思考细节