**上机一：（考察知识点Swift中的继承、构造函数等）**

**题目：**

在Playground中设计一个学生类Student和它的一个子类UnderGraduate，要求如下：首先定义一个Person类，接下来定义一个Student类继承自Person类。

1．Student类有name（姓名）和age（年龄）属性，一个包含两个参数的构造函数，用于给name和age属性赋值，一个show（）方法打印Student的属性信息。

2．本科生类UnderGraduate增加一个major（专业）属性，有一个包含三个参数的构造函数，前两个参数用于给继承的name和age属性赋值，第三个参数给major专业赋值，一个重写的show方法用于打印UnderGraduate的属性信息。

3．新建一个UnderGraduate类的对象，姓名为“小明”，年龄为19，专业是“信息工程”，并调用该对象的show方法。

class Student {

var name:String

var age:Int

init(name:String,age:Int){

self.name=name

self.age=age

}

func show(){

print("姓名：",name)

print("年龄：",age)

}

}

class UnderGraduate:Student{

var major:String

init(name:String,age:Int,major:String){

self.major=major

super.init(name:name,age:age)

}

override func show(){

print("姓名：",name)

print("年龄：",age)

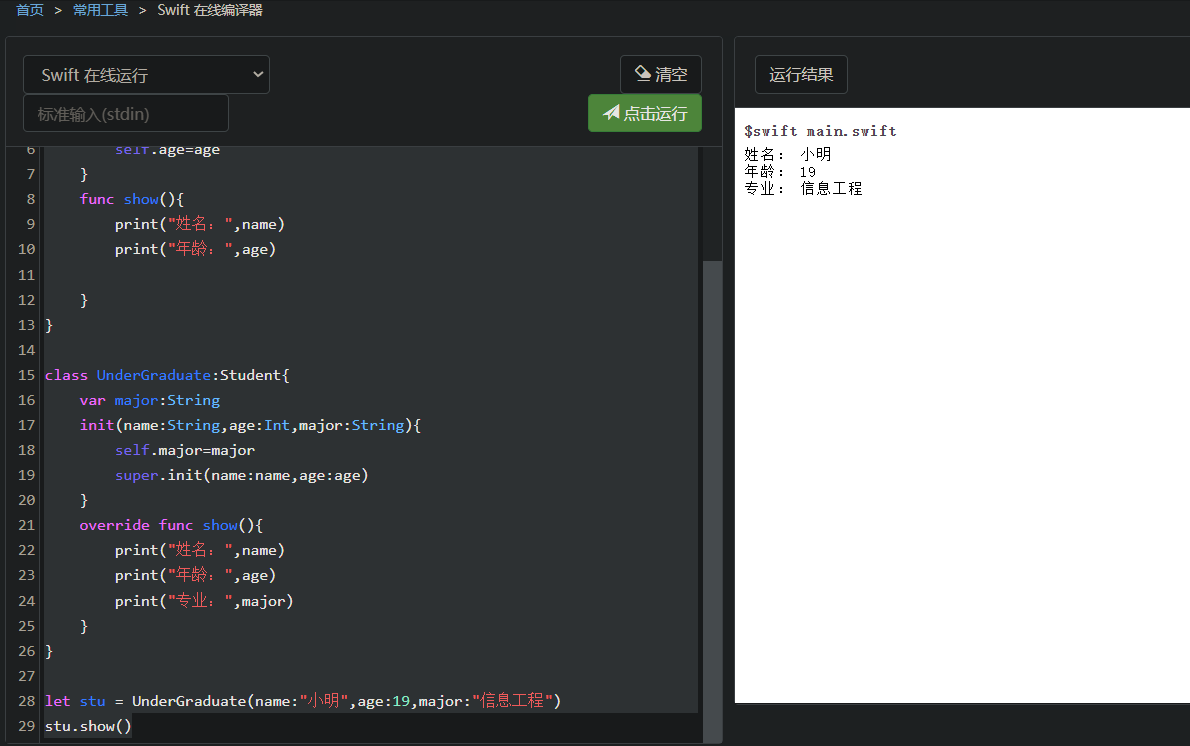
print("专业：",major)

}

}

let stu = UnderGraduate(name:"小明",age:19,major:"信息工程")

stu.show()



**上机二：（考察知识点Swift中的重写）**

**题目：**

在Playground中设计一个图形类（Shape）和它的两个子类，圆形类（Circle）和方形类（Square），要求如下：

1. 图形类（Shape）有一个draw方法，输出“画形状…”。
2. 圆形类（Circle）重写了draw方法，输出“画圆形”。
3. 方形类（Square）重写了draw方法，输出“画方形”。
4. 新建一个方法，叫做drawShape，它有一个Shape类型的参数shape，在drawShape方法体内调用shape参数的draw方法。
5. 分别创建Shape对象，Circle对象和Square对象作为参数传入drawShape方法
6. class Shape {
7. func draw() {
8. print("画形状")
9. }
10. }
11. class Circle:Shape{
12. override func draw() {
13. print("画圆形")
14. }
15. }
16. class Square:Shape{
17. override func draw() {
18. print("画方形")
19. }
20. }
21. func drawShape(*shape*:Shape) {
22. shape.draw()
23. }
24. drawShape(Shape())
25. drawShape(Circle())
26. drawShape(Square())