

中文	英文	示例	说明
如果	if	如果 <code>x > 0</code> : 打印("正数")	条件判断语句
否如	elif	如果 <code>x > 0</code> : 打印("正") 否如 <code>x < 0</code> : 打印("负")	多分支条件判断，否则如果。
否则	else	如果 <code>x % 2 == 0</code> : 打印("偶数") 否则: 打印("奇数")	条件不成立时执行
循环	for	循环 <code>i</code> 在 生成范围(3): 打印(i)	遍历可迭代对象
判断循环	while	判断循环 <code>count < 5</code> : 打印(count) count += 1	条件为真时重复执行
函数	def	函数 <code>greet()</code> : 打印("Hello")	定义函数
类	class	类 <code>Person</code> : 跳过	定义类
返回	return	函数 <code>add(a, b)</code> : 返回 <code>a + b</code>	从函数返回值
导入	import	导入 <code>math</code>	导入模块
从	from	从 <code>math</code> 导入 <code>平方根</code>	从模块导入特定内容
作为	as	导入 <code>numpy</code> 作为 <code>np</code>	为导入的模块/对象起别名
尝试	try	尝试: <code>x = 1 / 0</code> 捕获异常: 打印("错误")	异常处理开始
捕获异常	except	尝试: <code>x = 1 / 0</code> 捕获异常: 打印("错误")	捕获特定异常

中文	英文	示例	说明
最终执行	finally	尝试: file = 打开("a.txt") 最终执行: file.关闭()	无论是否异常都执行
抛出异常	raise	抛出异常 ValueError("无效值")	主动抛出异常
断言	assert	断言 x > 0, # "x 必须为正"	调试时检查条件, 为假会抛出异常
删除	del	删除 my_list[0]	删除对象或元素
全局变量	global	函数 func(): 全局变量 x x = 10	声明使用全局变量
外层变量	nonlocal	函数 outer(): x = 0 函数 inner(): 外层变量 x x = 1	声明使用外层函数变量
是	is	如果 x 是 空值: 打印("空值")	判断对象身份 (是否同一对象)
与	and	如果 a > 0 与 b > 0: 打印("都为正")	逻辑 "与" 操作
或	or	如果 a == 0 或 b == 0: 打印("至少一个为零")	逻辑 "或" 操作
非	not	如果 非 flag: 打印("未启用")	逻辑 "非" 操作
异步	async	异步 函数 fetch(): 等待 asyncio. 延时(1)	定义异步函数
等待	await	result = 等待 fetch()	等待异步操作完成
生成器	yield	函数 gen(): 生成器 1 生成器 2	从生成器返回值并暂停

中文	英文	示例	说明
跳过	pass	如果 真: 跳过	空语句, 占位用
到循环尾	continue	循环 i 在 生成范围(5): 如果 i == 2: 跳出循环 打印(i)	跳过当前循环迭代
跳出循环	break	循环 i 在 生成范围(10): 如果 i == 5: 停止循环 打印(i)	立即退出循环
真	True	flag = 真	布尔值“真”
假	False	flag = 假	布尔值“假”
空	None	result = 空	表示“无”或空值
管理	with	管理 打开("file.txt") 作为 f: data = f. 读取()	自动管理资源 (如文件)
匿名函数	lambda	square = 匿名函数 x: x ** 2 打印(square(3))	定义匿名函数
在	in	如果 "a" 在 "apple": 打印("包含")	成员资格测试