

# 笔记

版权：智泊AI

作者：Jeff

## 一、分支【重点掌握】

### 1.代码结构

顺序结构：代码从上往下依次执行

分支结构：根据不同的条件，执行不同的语句

循环结构：根据指定的条件，重复执行某段代码

### 2.分支结构-if语句

#### 2.1简单if语句【单分支】

语法：

if 表达式：

语句

说明; 要么执行，要么不执行，当表达式成立的之后，则执行语句；如果表达式不成立，则直接跳过整个if语句继续执行后面的代码

注意：表达式为真才会运行if中的语句

代码演示：

```
# 单分支
num1 = 50
num2 = 60

# 需求：当num1 != num2, 则给num1重新赋值为100

# 在Python中，通过缩进来区分代码块
if num1 != num2:
    num1 = 100
    print(num1)

# 练习：从控制台输入一个数，判断这个数是否是偶数
num = int(input())
if num % 2 == 0:
    print(num, "是一个偶数")
```

## 2.2 if-else语句【双分支】

语法：

if 表达式：

    语句1

else:

    语句2

说明：如果表达式成立，则执行语句1；如果不成立，则执行语句2

代码演示：

```
# 双分支
# 从控制台输入一个数，判断这个数是否是偶数
num = int(input("请输入一个数："))

if num % 2 == 0:
    print(num, "是一个偶数")
else:
    print(num, "不是一个偶数")

#练习：从控制台输入一个数字，根据数字打印年龄段
age = int(input())
if age >= 18:
    print("成年人")
else:
    print("未成年人")
```

## 2.3 if-elif-else语句【多分支】

语法：

if 表达式1：

    语句1

elif 表达式2：

    语句2

elif 表达式3：

    语句3

...

else:

    语句n

说明：实现了多选一的操作，会根据不同的条件从上往下来进行匹配，如果匹配上了，则执行对应的语句，然后直接结束整个分支，但是，如果前面所有的条件都不成立的话，则执行else中的语句

注意：不管if-elif-else有多少个分支，都只会进入其中的一个分支

代码演示：

```
# 多分支
# 需求：从控制台输入一个整数年龄，根据数字打印年龄段
age = int(input("请输入您的年龄："))
if age < 0:
    print("输入有误")
elif age <= 3:
    print("婴儿")
elif age <= 6:
    print("儿童")
elif age <= 12:
    print("青少年")
elif age < 18:
    print("青年")
else:
    print("恭喜你！成年了！")
```

```
# 练习：根据控制台输入的成绩，输出对应的等级
"""
90以上：优秀
80~90：良好
70~80：还行
70以下：加油吧，少年
"""
score = int(input("请输入学生的成绩："))
if score >= 90:
    print("优秀")
elif score >= 80:
    print("良好")
elif score >= 70:
    print("还行")
else:
    print("")
```

## 2.4 嵌套if语句

语法：

if 表达式1：

    语句1

if 表达式2：

    语句2

说明：if语句的嵌套，可以在单分支，双分支，多分支之间进行任意组合

代码演示：

```
# 输入成绩，打印成绩结果
score = int(input("请输入学生的成绩："))
if score < 0 or score > 100:
    print("输入有误")
else:
    if score >= 90:
        print("优秀")
    elif score >= 80:
        print("良好")
    elif score >= 60:
        print("及格")
    else:
        print("不及格")

# 输入年龄和相貌，判断如果是18岁以上的美女，则要微信，否则略过
age = int(input("请输入年龄："))
looks = input("请输入您的相貌：")

if age >= 18:
    if looks == "美女":
        print("要微信")
    else:
        print("略过")
```

注意：从语法角度来说，嵌套的层数没有任何的限制，但是，为了代码的可读性和可维护性，嵌套层数一般不建议超过3层