

# 一 课程回顾

1. 编写一个程序，检查任意一个年份是否是闰年

- 如果一个年份可以被4整除不能被100整除，或者可以被400整除，这个年份就是闰年（伪代码）

3. 计算 $1+2+3+\dots+100$ （使用while循环）

4. 打印99乘法表：

```
1*1=1
1*2=2   2*2=4
1*3=3   2*3=6   3*3=9
1*4=4   2*4=8   3*4=12  4*4=16
1*5=5   2*5=10  3*5=15  4*5=20  5*5=25
1*6=6   2*6=12  3*6=18  4*6=24  5*6=30  6*6=36
1*7=7   2*7=14  3*7=21  4*7=28  5*7=35  6*7=42  7*7=49
1*8=8   2*8=16  3*8=24  4*8=32  5*8=40  6*8=48  7*8=56  8*8=64
1*9=9   2*9=18  3*9=27  4*9=36  5*9=45  6*9=54  7*9=63  8*9=72  9*9=81
```

```
for i in range(1,10): # i = 1   i = 2
    for j in range(1,i+1): # j = 1   j = range(1,2+1)   j = 1 , 2   j =
range(1,3+1) = 1,2,3
        print(j, '*', i, '=', i*j, end='\t') # \t 就是四个空格
    print()
```

```
year = int(input('请输入任意一个年份:'))
if year % 4 == 0 and year % 100 != 0 or year % 400 == 0:
    print(year, '是闰年')
else:
    print(year, '不是闰年')
```

```
num = 1 # 计数起始值的变量
sum = 0 # 初识化一个累计的变量   也就是容器
while num < 101:
    sum += num
    num += 1
print('sum=', sum)
```

## 二 数据类型进阶(一)

python的数据类型可以分为三类: 数值类型，序列类型，散列类型

数值类型: int float bool （字符串不属于数值类型 属于序列类型）

### 2.1 序列类型

```
s = 'CSDN博客论坛'
# 取到 C 和 坛
print(s[0], '=', s[-8])
print(s[7], '=', s[-1])
```

可变类型: 列表 集合 字典 (数据发生改变 内存地址不变 所使用的数据是同一个数据)

不可变类型: 数字 字符串 以及 元祖 (数据发生改变时 内存地址也改变, 会产生一个份新的数据)

## 2.2 字符串 str

字符串是一个有序的字符集合, 用于存储和表示基本的文本信息

### 2.2.1 常用方法

操作数据最本质的行为是什么 ==》 增删改查

字符串.方法名()

#### find()

范围性查找子串, 返回索引值, 没有则返回-1

语法:

```
str.find(str, beg=0, end=len(string))
```

- str==>指定检索的字符串
- beg==> 开始的索引, 默认为0
- end ==》 结束的索引, 默认为字符串的长度

常用于检测字符串中是否包含子字符串str

```
s = 'hello world'
print(s.find('h')) # h的索引值就是0
print(s.find('world')) # 找到world则返回首个字符的索引
print(s.find('world')) # find不会报错 没有则返回-1

s = 'python,hello world,java,php'
print(s.find('h')) # 返回的只会是首个h的下标
print(s.find('h', 4)) # 4代表的意思 就是从下标为4的地方开始找h
print(s.rfind('h')) # rfind 从右往左找 返回的值是从左往右的值
```

#### index()

检测字符串中是否包含子字符串str, 如果指定beg和end范围, 则检查是否包含在指定范围内, 和find差不多, 只不过如果子str不在字符串中会报错

```
s = 'python,hello world,java,php'
print(s.index('world'))
print(s.index('world')) # 报错
```

## count()

返回找到的子串个数，找不到就返回0

```
s = 'python,helloworld,java,php'
print(s.count('a'))
```

## strip()

去除字符串两边的空格，换行符，制表符，得到的是一个新的字符串

```
code = input('请输入4位的验证码:')
data = code.strip()
if data == 'ABCD':
    print('验证码正确')
else:
    print('验证码错误')
```

## startswith(),endswith()

- 判断字符串是否以XX开头，得到一个布尔值

```
add = input('请输入住址:')
if add.startswith('湖南省'):
    print('湖南人口') # True
else:
    print('非湖南人口') # False
```

- 判断字符串是否以XX结尾，得到一个布尔值

```
add = input('请输入住址:')
if add.endswith('村'):
    print('农业户口') # True
else:
    print('非农业户口') # False
```

```
s = '北京市朝阳区'
print(s.startswith('北京市'))
```

## isdigit()

判断字符串是否是数字组成，返回的结果为True和False

```
v1 = input('请输入第一个值:')
v2 = input('请输入第二个值:')
if v1.isdigit() and v2.isdigit():
    print('是数字组成')
else:
    print('请输入正确的数字')

v1 = input('请输入第一个值:')
```

```

v2 = input('请输入第二个值:')
if v1.isdigit() and v2.isdigit():
    data = int(v1) + int(v2)
    print(data)
else:
    print('请输入正确的数字')

```

## lower(),upper()

字符串变大写/小写，得到一个新的字符串

lower ==》转小写

upper ==》转大写

```

s = 'C'
print(s)
s1 = s.lower() # 转小写
print(s1) # c
s2 = s1.upper() # 转大写
print(s2)

```

## split()

切分字符串，将字符串类型转成列表，默认以空格切分，也可以指定字符切分

```

s = 'my name is zuoshou'
print(s.split())
s1 = 'python,helloworld,java,php'
print(s1.split(','))

```

## replace()

字符串内容的替换，得到一个新的字符串

```

# 替换敏感词
a = input('请输入评论信息:')
a1 = a.replace('人','*')
print(a1)

```

## join()

用于将序列中的元素以指定的字符串连接生成一个新的字符串

常用于将列表转为字符串

```

arr = ['左手','和','轩轩','是','好朋友']
print(''.join(arr)) # 常用于转化列表为字符串使用
print('+'.join(arr))

```

## 2.2.2 字符串的公共功能

### 切片

顾头不顾尾(左闭右开): 取出索引0-8的所有字符

```
s = 'hello python'
print(s[0:9])
```

步长: 0:9:2 第三个参数是2代表步长, 会从下标为0开始, 每次累加一个2取值, 所以索引值为0/2/4/6/8 字符

```
s = 'hello python'
print(s[0:9:2])
```

### 反向切片

```
s = 'hello python'
print(s[::-1]) # 从右往左依次取值, 步长为负数, 就是从右往左, 首尾不给值就是取所有
```

## 2.3 列表 list

可以存放多个不同类型的元素, 记录多个同种属性的值

列表: 存储同一个类别的数据, 方便操作

```
字符串, 不可变类型: 创建好之后内部无法修改[内置功能都是新创建一份数据]
name = 'zuoshou'
data = name.upper()
print(name)
print(data)
```

```
列表, 可变类型: 创建好之后内部元素可以修改[独有功能基本上都是直接操作列表内部, 不会创建一份新的数据]
list1 = ['车子', '房子']
list1.append('票子')
print(list1)
print(id(list1))
```

### 列表的创建

```
列表名(变量名) = []
li = [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

### 列表的增删改查

## 增，追加，在原列表中尾部追加值

1.append: 添加一个数据，添加到列表的最后一位

语法: 列表名.append('数据')

```
li = [1,2,3,4,5,6,7,8,9]
```

```
li.append('左手')
```

```
print(li)
```

2.insert: 添加一个数据到指定的位置

语法: 列表名.insert(下标位置, 数据)

```
li.insert(0, '轩轩')
```

```
print(li)
```

3.extend: 添加一个序列类型到最后一位，并且把序列类型拆分

语法: 列表名.extend(序列类型)

```
li.extend('浩天')
```

```
li.extend(['浩天']) # 如果用中括号括起来就不会拆分
```

```
print(li)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, '左手']
```

```
['轩轩', 1, 2, 3, 4, 5, 6, 7, 8, 9, '左手']
```

```
['轩轩', 1, 2, 3, 4, 5, 6, 7, 8, 9, '左手', '浩', '天', '浩天']
```

## 作业

```
# 2.将列表["h","e","l","l","o"," "," ","w","o","r","l","d"] 转为字符串 hello,world
```

```
# 4.
```

```
'''
```

输入一个字符串统计每个元音字母（aeiou）在字符串中出现的次数。

对于结果输出5行，格式如下：

a:num1（a的个数）

e:num2（e的个数）

i:num3（i的个数）

o:num4（o的个数）

u:num5（u的个数）

例如输入aeioubbbccc，输出：

```
a:1
```

```
e:1
```

```
i:1
```

```
o:1
```

```
u:1
```

```
'''
```

```
s = "Hello, world!"
```

1.给定一个字符串s，如何获取字符串中第一个字符？

2.给定一个字符串s，如何获取字符串中倒数第二个字符？

3.给定一个字符串s，如何获取字符串中前三个字符的子串？

4.给定一个字符串s，如何获取字符串中除了最后一个字符以外的所有字符？

5.给定一个字符串s，如何获取字符串中从第二个字符开始到倒数第四个字符为止的子串？

6.给定一个字符串s，如何获取字符串中每隔两个字符取一个字符的子串？

7.给定一个字符串s，如何获取字符串中所有奇数索引的字符的子串？

8.给定一个字符串s，如何将字符串倒序输出？

9. 给定一个字符串 `s`，如何获取字符串中最后五个字符的子串？

10. 给定一个字符串 `s`，如何获取字符串中倒数第四个字符到末尾的子串？