

day03 课程大纲

一. 今日内容大纲

1. 基础数据类型总览

2. int

3. bool

4. **str**

- 索引, 切片
- 常用操作方法

5. **for** 循环

二. 昨日内容以及作业讲解

1. pycharm 简单使用

2. while 循环

1. 结构

2. pass

3. 格式化输出: str : 让字符串的某些位置变成动态可变的, 可传入的。

1. % s str d digit i int r

2. %%

4. 编码的初识:

1. 编码: 密码本: 二进制 与 文字的对应关系。

- ASCII: 最早的密码本: 二进制与 英文字母, 数字, 特殊字符的对应关系

01100001 a

01100010 b

'abc': 01100001 01100010 01100011 : 3个字节

- GBK 国标: 英文.... 1个字节 中文 2个字节

- 英文字母, 数字, 特殊字符 ASCII

- 中文自己编写的。

'ab太白': 6个字节

- Unicode: 万国码。(兼容性高, 他与任何的密码本都有映射关系)

01100001 01100001 01100001 01100001 a

01100001 01100001 01100101 01100001 中

- UTF-8:

01100001 a 使用的ASCII

01100001 01100001 欧洲..

01100001 01100101 01100001 中

'ab太白' 8个字节

三. 具体内容:

1. 基础数据类型总览

- 10203 123 3340 **int** +- * / 等等
- '今天吃了没?' **str** 存储少量的数据, + *int 切片, 其他操作方法
- True False **bool** 判断真假
- [12, True, 'alex', [1,2,3]] **list** 存储大量的数据。
- (12, True, 'alex', [1,2,3]) **tuple** 存储大量的数据, 不可改变里面的元素。
- {'name': '太白金星'} **dict** 存储大量的关联型的数据, 查询速度非常快。
- set 交集, 并集差集。。。

2. int

- 十进制二进制转换

```
'''
二进制转换成十进制
0001 1010      -----> ?   26
'''

b = 1 * 2**4 + 1 * 2**3 + 0 * 2**2 + 1 * 2**1 + 0 * 2**0
# print(b)  # 26

'''
42 -----> 0010 1010
'''
```

- bit_length 十进制转化成二进制的有效长度

```
# bit_length 有效的二进制的长度
i = 4
print(i.bit_length())  # 3
i = 5
print(i.bit_length())  # 3
i = 42
print(i.bit_length())  # 4
```

3. bool

- bool str int 三者之间的转换

```
# bool str int
# bool <----> int
'''
```

```

True    1    False    0
非零即True    0 是 False
'''

# str    <--->    int    ***
'''
s1 = 10        int(s1)    : 必须是数字组成
i = 100        str(i)
'''

# str    bool    ***
# 非空即True
s1 = ' '
print(bool(s1))

s1 = '' # 空字符串
print(bool(s1))
# bool    --->    str    无意义
print(str(True))

```

◦ 应用:

```

s = input('输入内容')
if s:
    print('有内容')
else:
    print('没有输入任何内容')

```

4. str

◦ 索引切片步长

```

s1 = 'python全栈22期'
# 对字符串进行索引，切片出来的数据都是字符串类型。
# 按照索引取值
# 从左至右有顺序，下标，索引。
s2 = s1[0]
print(s2,type(s2))
s3 = s1[2]
print(s3)
s4 = s1[-1]
print(s4)

# 按照切片取值。
# 顾头不顾腭
s5 = s1[0:6]
s5 = s1[:6]
print(s5)
s6 = s1[6:]
print(s6)

# 切片步长

```

```

s7 = s1[:5:2]
print(s7)
print(s1[:])
# 倒序:
s8 = s1[-1:-6:-1]
print(s8)

# 按索引: s1[index]
# 按照切片: s1[start_index: end_index+1]
# 按照切片步长: s1[start_index: end_index+1:2]
# 反向按照切片步长: s1[start_index: end_index后延一位:2]
# 思考题: 倒序全部取出来?

```

◦ 练习题

2. 有字符串 `s = "123a4b5c"`

通过对s切片形成新的字符串s1, `s1 = "123"`
 通过对s切片形成新的字符串s2, `s2 = "a4b"`
 通过对s切片形成新的字符串s3, `s3 = "1345"`
 通过对s切片形成字符串s4, `s4 = "2ab"`
 通过对s切片形成字符串s5, `s5 = "c"`
 通过对s切片形成字符串s6, `s6 = "ba2"`

◦ 常用操作方法

```

# upper lower
# s1 = s.upper()
# # s1 = s.lower()
# print(s1,type(s1))

# 应用:
username = input('用户名')
password = input('密码')
code = 'QweA'
print(code)
your_code = input('请输入验证码: 不区分大小写')
if your_code.upper() == code.upper():
    if username == '太白' and password == '123':
        print('登录成功')
    else:
        print('用户名密码错误')
else:
    print('验证码错误')

```

四. 今日总结

1. `bool str int` 三者之间的转换 ***
2. `str`索引切片, 常用操作方法

3. for循环 (大量的练习题)

五. 预习内容

1. 列表

- 初识
- 增删改查
- 嵌套

2. 元组tuple 5分钟

3. range