Python基础

输入与输出

```
# 输入函数 - 从用户获取数据（返回字符串类型）
name = input("请输入你的姓名：")

# 输出函数 - 打印内容到控制台
print("你好，", name)  # 逗号分隔自动加空格
print(f"希望{name}复习顺利！")  # f-string格式化（推荐）
print("年龄：" + str(20))  # 字符串拼接
```

变量

```
# 变量命名规则：字母/下划线开头，由字母/数字/下划线组成
age = 18  # 整型变量
height = 1.75  # 浮点型变量
is_student = True  # 布尔型变量
message = "老师捞捞！"  # 字符串变量

# 变量重新赋值
counter = 10
counter = counter + 1  # 变量更新
```

数据类型

基本数据类型

```
num_int = 30        # 整数 int
num_float = 3.14    # 浮点数 float
text = "Hello"      # 字符串 str
is_valid = True     # 布尔值 bool
```

容器类型

```
my_list = [1, 2, 3, "a", "b"]          # 列表（有序可变）
my_tuple = (1, "apple", True)           # 元组（有序不可变）
my_dict = {"name": "Alice", "age": 20}  # 字典（无序可变，键值对）
my_set = {1, 2, 3, 3, 2}                # 集合（无序唯一，自动去重）实际存储{1,2,3}
```

列表

```
# 定义
my_list = [1, "apple", True]  # 方括号[]，元素间逗号分隔
# 特点:有序、可变

# 常见操作
# 索引和切片
print(my_list[1])  # 访问第二个元素
print(my_list[0:2])  # 切片获取子列表

# 添加元素
my_list.append("new")  # 末尾添加
my_list.insert(1, "inserted")  # 指定位置插入

# 删除元素
my_list.remove("apple")  # 删除指定值第一个元素
removed_element = my_list.pop()  # 删除最后一个元素并返回

# 修改元素
my_list[0] = 10  # 直接赋新值

# 列表推导式
squared = [x ** 2 for x in range(5)]  # 生成新列表
```

元组

```
# 定义
my_tuple = (1, "apple", True)  # 圆括号()，元素间逗号分隔
# 特点:有序、不可变

# 常见操作
# 索引和切片
print(my_tuple[0])  # 访问第一个元素
print(my_tuple[1:3])  # 切片获取子元组

# 遍历元组
for item in my_tuple:
    print(item)

# 元组拼接
new_tuple = my_tuple + (4, 5)
```

字典

```
# 定义
my_dict = {"name": "Alice", "age": 25}  # 大括号{}，键值对逗号分隔，键和值冒号分隔
```

```
# 特点:无序、可变

# 常见操作
# 访问值
print(my_dict["name"])  # 获取键"name"对应的值

# 添加键值对
my_dict["height"] = 1.75  # 新键赋值

# 删除键值对
del my_dict["age"]  # 删除键"age"及其对应值

# 获取字典视图
keys = my_dict.keys()  # 获取所有键
values = my_dict.values()  # 获取所有值
items = my_dict.items()  # 获取所有键值对
```

集合

```
# 定义
my_set = {1, "apple", True}  # 大括号{}或set()
same_set = set([1, "apple", True])

# 特点:无序、唯一性

# 常见操作
# 添加元素
my_set.add("new")  # 添加单个元素
my_set.update([4, 5])  # 添加多个元素

# 删除元素
my_set.remove("apple")  # 删除指定元素，不存在会报错
my_set.discard("apple")  # 删除指定元素，不存在不会报错
popped_element = my_set.pop()  # 随机删除一个元素并返回

# 集合运算
another_set = {3, 4, 5}
union_set = my_set.union(another_set)  # 并集
intersection_set = my_set.intersection(another_set)  # 交集
difference_set = my_set.difference(another_set)  # 差集
```

运算符

算数运算符

```
print(10 + 3)   # 13
print(10 - 3)   # 7
print(10 * 3)   # 30
print(10 / 3)   # 3.333... (浮点除法)
```

```
print(10 // 3)  # 3（整除）
print(10 % 3)   # 1（取模）
print(2 ** 4)   # 16（幂运算）
```

比较运算符

```
print(5 > 3)    # True
print(5 == 5.0) # True（值相等）
print(5 != "5") # True（类型不同）
```

逻辑运算符

```
print(True and False)  # False
print(True or False)   # True
print(not True)        # False
```

类型转换

显式类型转换

```
num_str = "123"
num_int = int(num_str)    # 字符串→整数
num_float = float("3.14") # 字符串→浮点数
str_num = str(456)        # 整数→字符串
```

隐式转换（自动）

```
result = 5 + 2.5  # int + float → float (7.5)
```

类型检查

```
print(type("hello"))      # <class 'str'>
print(isinstance(10, int)) # True
```

条件分支流程

```
score = 85

# 单分支
```

```
if score >= 60:
    print("及格! ")

# 双分支
if score >= 90:
    print("优秀")
else:
    print("良好" if score >= 70 else "及格")  # 三元表达式

# 多分支
if score >= 90:
    grade = "A"
elif score >= 80:
    grade = "B"
elif score >= 70:
    grade = "C"
else:
    grade = "D"
print(f"等级: {grade}")
```

循环流程

for循环

```
# 遍历序列
fruits = ["apple", "banana", "cherry"]
for fruit in fruits:
    print(f"我喜欢吃{fruit}")
```

range()函数

```
for i in range(3):        # 0,1,2
    print(f"计数:{i}")
```

while循环

```
count = 3
while count > 0:
    print(f"倒计时: {count}")
    count -= 1  # 等同于 count = count - 1
```

循环控制

```
```

```
for num in range(10):
    if num == 3:
        continue  # 跳过当前迭代
    if num == 7:
        break     # 终止循环
    print(num)
```

函数

```
def calculate_area(length, width=1):
    """计算矩形面积，默认宽度为1（正方形）"""
    area = length * width
    return area
# 函数调用
print("\n函数演示:")
print("矩形面积:", calculate_area(5, 3))
print("正方形面积:", calculate_area(4))  # 使用默认参数

# 递归函数示例
def factorial(n):
    """计算阶乘的递归函数"""
    if n == 1:
        return 1
    else:
        return n * factorial(n-1)

print("5的阶乘:", factorial(5))
```